COPYRIGHT AND CONFIDENTIALITY NOTICE

The work described in this document was performed as part of the xDELIA project ('Boosting Deliberate Practice and Handling Biases through Immersive Cognitive and Emotional Reinforcement Strategies & Tools') which is funded under contract No. 231830 of the European Community. The project is a collaboration between CIMNE (coordinating partner), Forschungszentrum Informatik, Open University, Blekinge Tekniska Högskola (Game and Media Arts Laboratory), Erasmus University Rotterdam (Erasmus Centre for Neuroeconomics), University of Bristol (Personal Finance Research Centre), and Saxo Bank A/S. The opinions, findings and conclusions expressed in this report are those of the authors alone and do not necessarily reflect those of the EC or any other organisation involved in the project.

No part of this publication may be reproduced or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording or otherwise; nor stored in any information retrieval system of any kind; nor used for tendering or manufacturing; nor communicated to any person, without the prior permission in writing of the copyright owner.

The contents of this document must be treated as confidential and protected as defined in the terms and conditions of the xDELIA Consortium Agreement.
Appendices

<table>
<thead>
<tr>
<th>Document Name</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix A. Two index game</td>
<td>September 16, 2010</td>
</tr>
<tr>
<td>Appendix B. Play testing Two Index game, interview questions</td>
<td>February 23, 2011</td>
</tr>
<tr>
<td>Appendix C. Game Experience Questionnaire</td>
<td>February 23, 2011</td>
</tr>
<tr>
<td>Appendix D. System Usability Scale</td>
<td>February 23, 2011</td>
</tr>
<tr>
<td>Appendix E. Consent to participate in Research Study</td>
<td>February 28, 2011</td>
</tr>
</tbody>
</table>
Executive Summary

The objective of Work Package 4 is to design, implement and evaluate game prototypes based on the requirements in the project. This document describes the concepts and prototypes created during the third year of the xDELIA project. The document also describes the evaluation efforts done. The document consists of two parts describing WP2 and WP3 games respectively.

The design and development activities in WP2 started during year two and have resulted in four games (the Aiming game, the Auction game, the Two index game and the Mindfulness game) that will be components in the WP2 learning intervention. The ideas for the games were developed at a workshop with partners from WP2 and WP4. The design and development process has been highly collaborative and iterative where the developer and the product owner have worked closely together.

The Aiming game and the Auction game mainly focus on two aspects, namely
- to make the player aware of his or her emotions and
- to train the player in emotion regulation

The Two index game also has two kinds of usage. The game can be used as
- a diagnostic tool as well as
- a game for learning to manage the disposition effect.

The main purpose of the Mindfulness game is to
- give the player an opportunity to try out some basic mindfulness exercises and experience how mindfulness can be practiced in stressful situations.

A secondary purpose is to
- provide an environment where the player can train mindfulness in a context separate from every day life.

In addition a mindfulness app called xDelia Mind Former has been developed. The xDelia Mind Manager is a toolset with the purpose to facilitate the mindfulness training in the xDelia Mindfulness Trading Training, a four week course for the investors to learn how to practice mindfulness in financial decision making.

The WP2 games have been evaluated and play tested from a game perspective and the results give some positive indications. The three games (with future variations) developed for WP2 will during year three be developed further as well as evaluated with regards how the use of the games creates the desired learning outcome.
## Contents

1 **Introduction**  
1.1 *Document Purpose and Scope*  
1.2 *List of Acronyms*  

2 **WP2 games**  
2.1 *Overview of the design process in WP2*  
2.2 *Aiming game*  
2.2.1 Aiming Game I  
2.2.2 Aiming Game II – Space Investor  
2.3 *Auction game*  
2.3.1 Introduction  
2.3.2 Game description  
2.3.3 Game logic  
2.3.4 Game difficulty  
2.3.5 Input  
2.3.6 Rewards and ranking  
2.3.7 Hardware  
2.3.8 Software  
2.3.9 Use cases and UML  
2.3.10 Sequence diagram  
2.3.11 Class diagram  
2.3.12 Evaluation  
2.3.13 Discussion and conclusion  
2.4 *Two Index Game*  
2.4.1 Introduction  
2.4.2 Description of Prototype  
2.4.3 Prototype specification in summary  
2.4.4 Evaluation  
2.5 *Mindfulness Game*  
2.5.1 Prototype Summary  
2.5.2 Introduction  
2.5.3 Game Description  
2.5.4 Software  
2.5.5 Evaluation  
2.5.6 Summary  
2.6 *xDelia Mind Manager*  
2.6.1 Prototype Summary  
2.6.2 Introduction  
2.6.3 Description of Prototype  
2.6.4 Implementation  
2.6.5 Evaluation  

3 **WP3 Concepts and Games**  
3.1 *Overview of the Design Process in WP3*  
3.2 *Relationship between Y1 and Y2 prototypes*  
3.3 *FinBoard Game*  
3.3.1 Evaluation of the FinBoard game  
3.4 *Banking Game*  
3.4.1 Game play
3.4.2 Game elements 134
3.4.3 Mini-games 134
3.4.4 Evaluation of the Banking Game 137
3.5 Go/nogo prototype series 138
  3.5.1 Description of prototype 139
  3.5.2 Technical Description 143
  3.5.3 Evaluation 151
  3.5.4 Reflections 163
  3.5.5 Conclusions and future work 163
3.6 MINDswap 164
  3.6.1 Game synopsis 164
  3.6.2 Characters 165
  3.6.3 Story (Level 1) 166
  3.6.4 Game world and player interface 167
  3.6.5 Online fincap game (use-centric view) 168
  3.6.6 Non-functional requirements 172
  3.6.7 Online fincap game (game-centric view) 174
  3.6.8 High-level architecture 175
  3.6.9 Summary 176
4 Conclusions 178
5 Appendix A – Two Index Game 179
6 Appendix B – Play testing Two index game, interview questions 188
7 Appendix C – Game Experience Questionnaire, GEQ 190
8 Appendix D – System Usability Scale, SUS 191
9 Appendix E 192
References 193
Tables

Table 2.1 – Initial estimation of required time to finish first prototype cycle ................................................................. 23
Table 2.2 – Breakdown of specification into work items .................................................................................................. 23
Table 2.3 – Example of Arousal Statistics text file ........................................................................................................... 29
Table 2.4 – Example of Shot Statistics text file ................................................................................................................ 29
Table 2.5 – Money earned and Time needed to reach a decision at different arousal levels ............................................. 83
Table 2.6 – Grouping of the GEQ ...................................................................................................................................... 89
Table 2.7 – Overview of levels ......................................................................................................................................... 97
Table 2.8 - Requirements ................................................................................................................................................ 98
Table 2.9 - Overview of what data must be logged ......................................................................................................... 111
Table 2.10 - Final list of requirements ........................................................................................................................... 116
Table 2.11 – Relationship between requirements and use cases .................................................................................... 117
Table 2.12 Summary of the outcome from the user tests ............................................................................................... 127
Table 2.13 – Summary of the different versions of Go/nogo and LineRacer ...................................................................... 140
Table 2.14 – Reaction times (in ms) for each phase (cycle number in brackets) in the different prototypes ..................... 153
Table 2.15 – Refining the high-level use cases ................................................................................................................ 169
Table 2.16 – Non-functional requirements .................................................................................................................. 172

Figures

Figure 2.1 – Screen shot of the Aiming Game .................................................................................................................. 12
Figure 2.2 – Emotions in the valence-arousal space .......................................................................................................... 13
Figure 2.3 – Arousal bar at the bottom of the screen ......................................................................................................... 15
Figure 2.4 – The Emotiv EPOC device ............................................................................................................................. 15
Figure 2.5 – The Aiming Game during Phase 1 .................................................................................................................. 17
Figure 2.6 – The Aiming Game during phase 2 ................................................................................................................... 17
Figure 2.7 – The Aiming Game during phase 3 ................................................................................................................... 17
Figure 2.8 – Airplanes without blur .................................................................................................................................. 18
Figure 2.9 – Simple diagram of the entities and their inputs and outputs in the Aiming Game .............................................. 20
Figure 2.10 – Use Case of a player using the Aiming Game .............................................................................................. 25
Figure 2.11 – Diagram of the Aiming Game in the context of a study .............................................................................. 26
Figure 2.12 – Class diagram of most important parts of the Aiming Game ...................................................................... 27
Figure 2.13 – Emotions in the valence-arousal space ....................................................................................................... 54
Figure 2.14 – Print screen from the Space Investor game ................................................................................................. 56
Figure 2.15 – No sensor. The player’s progress during the game ....................................................................................... 62
Figure 2.16 – Sensor connected. The player’s progress during the game ......................................................................... 62
Figure 2.17 – Emotions in the valence-arousal space ....................................................................................................... 68
Figure 2.18 – Example sketch of a single round ................................................................................................................ 72
Figure 2.19 – Screenshot from the Auction Game ............................................................................................................... 73
Figure 2.20 – Diagram of game rounds ............................................................................................................................. 74
Figure 2.21 – Use Case diagram ....................................................................................................................................... 78
Figure 2.22 – Sequence diagram ...................................................................................................................................... 80
Figure 2.23 – Class diagram ............................................................................................................................................. 81
Figure 2.24 – Screenshot of Tutorial ................................................................................................................................. 86
Figure 2.25 – Screenshot of game play ............................................................................................................................... 87
Figure 2.26 – Results from SUS ....................................................................................................................................... 91
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.27</td>
<td>Game flow</td>
<td>96</td>
</tr>
<tr>
<td>2.24</td>
<td>Use cases</td>
<td>99</td>
</tr>
<tr>
<td>2.25</td>
<td>Start of game</td>
<td>100</td>
</tr>
<tr>
<td>2.26</td>
<td>Quit screen</td>
<td>100</td>
</tr>
<tr>
<td>2.27</td>
<td>Level 1 with end of level</td>
<td>101</td>
</tr>
<tr>
<td>2.28</td>
<td>Level 2 with end of level</td>
<td>102</td>
</tr>
<tr>
<td>2.29</td>
<td>Paced breathing exercise</td>
<td>102</td>
</tr>
<tr>
<td>2.30</td>
<td>Level 3 with end of level</td>
<td>103</td>
</tr>
<tr>
<td>2.31</td>
<td>Level 4 with end of level</td>
<td>105</td>
</tr>
<tr>
<td>2.32</td>
<td>Level 5 with start and end of level</td>
<td>106</td>
</tr>
<tr>
<td>2.33</td>
<td>Level 6 with end of level</td>
<td>107</td>
</tr>
<tr>
<td>2.34</td>
<td>Questionnaire</td>
<td>108</td>
</tr>
<tr>
<td>2.35</td>
<td>End screen</td>
<td>108</td>
</tr>
<tr>
<td>2.36</td>
<td>Flow through the game</td>
<td>109</td>
</tr>
<tr>
<td>2.37</td>
<td>Class diagram</td>
<td>110</td>
</tr>
<tr>
<td>2.38</td>
<td>Use case diagram</td>
<td>116</td>
</tr>
<tr>
<td>2.39</td>
<td>Start screen</td>
<td>117</td>
</tr>
<tr>
<td>2.40</td>
<td>Planner screen</td>
<td>118</td>
</tr>
<tr>
<td>2.41</td>
<td>Main menu</td>
<td>118</td>
</tr>
<tr>
<td>2.42</td>
<td>Screen to choose for how long to meditate</td>
<td>119</td>
</tr>
<tr>
<td>2.43</td>
<td>Meditation timer</td>
<td>119</td>
</tr>
<tr>
<td>2.44</td>
<td>Screen to choose for how long to do the breathing exercise</td>
<td>120</td>
</tr>
<tr>
<td>2.45</td>
<td>Screen with instructions of how to calibrate the paced breathing exercise</td>
<td>121</td>
</tr>
<tr>
<td>2.46</td>
<td>Calibration of paced breathing exercise</td>
<td>121</td>
</tr>
<tr>
<td>2.47</td>
<td>Screen with instructions of how to perform the paced breathing exercise</td>
<td>122</td>
</tr>
<tr>
<td>2.48</td>
<td>Paced breathing exercise. The red arrow shows how the user manually can adjust the breathing pace</td>
<td>122</td>
</tr>
<tr>
<td>2.49</td>
<td>Questionnaire</td>
<td>123</td>
</tr>
<tr>
<td>2.50</td>
<td>Statistics showing how many minutes a user has spent on mindfulness exercises the past week</td>
<td>123</td>
</tr>
<tr>
<td>2.51</td>
<td>Flow chart</td>
<td>124</td>
</tr>
<tr>
<td>2.52</td>
<td>Overview of files in prototype</td>
<td>125</td>
</tr>
<tr>
<td>2.53</td>
<td>Result from SUS questionnaire</td>
<td>128</td>
</tr>
<tr>
<td>3.1</td>
<td>Overview of relationships between prototypes/concepts</td>
<td>131</td>
</tr>
<tr>
<td>3.2</td>
<td>FinBoard game, table top version</td>
<td>132</td>
</tr>
<tr>
<td>3.3</td>
<td>Screen shots of the iPhone version of the FinBoard game</td>
<td>132</td>
</tr>
<tr>
<td>3.4</td>
<td>Picture of LineRacer(S) gameplay</td>
<td>140</td>
</tr>
<tr>
<td>3.5</td>
<td>Picture of Go/nogo(S) gameplay</td>
<td>140</td>
</tr>
<tr>
<td>3.6</td>
<td>Examples of go and nogo conditions</td>
<td>140</td>
</tr>
<tr>
<td>3.7</td>
<td>A turn broken down into its different parts</td>
<td>141</td>
</tr>
<tr>
<td>3.8</td>
<td>A cycle broken down into its different parts</td>
<td>141</td>
</tr>
<tr>
<td>3.9</td>
<td>A description of the use cases</td>
<td>145</td>
</tr>
<tr>
<td>3.10</td>
<td>The sequence in which the game is played</td>
<td>146</td>
</tr>
<tr>
<td>3.11</td>
<td>Class diagram of LineRacer</td>
<td>147</td>
</tr>
<tr>
<td>3.12</td>
<td>Test of Go/nogo task</td>
<td>152</td>
</tr>
<tr>
<td>3.13</td>
<td>Test of LineRacer</td>
<td>153</td>
</tr>
<tr>
<td>3.14</td>
<td>Average response times on go-trials during the different phases for each prototype</td>
<td>154</td>
</tr>
<tr>
<td>3.15</td>
<td>Average number of incorrect responses, over each phase, for each prototype</td>
<td>154</td>
</tr>
<tr>
<td>3.16</td>
<td>Relationship between game score and attentional impulsivity in Go/nogo(S)</td>
<td>155</td>
</tr>
<tr>
<td>3.17</td>
<td>Relationship between game score and attentional impulsivity</td>
<td>155</td>
</tr>
<tr>
<td>3.18</td>
<td>Relationship between motor impulsivity and game score in LineRacer(S)</td>
<td>156</td>
</tr>
<tr>
<td>3.19</td>
<td>Relationship between motor impulsivity and game score in LineRacer(M)</td>
<td>156</td>
</tr>
</tbody>
</table>
Figure 3.20 – Screen shot from the game..................................................................................................... 164
Figure 3.21 – How the player may control the character.............................................................................. 166
Figure 3.22 – Use-centric view of the fincap game system............................................................................. 169
Figure 3.23 – Game-centric view of the online financial capability game system............................................ 174
Figure 3.24 – Online financial capability game system............................................................................. 175
Figure 3.25 – Online financial capability game system............................................................................. 176
1 Introduction

1.1 Document Purpose and Scope

The objective of this document is to describe what prototypes have been created during the past year (2010) and also to briefly explain the processes that led to the prototypes. The document presents concepts and games created in both WP2 and WP3. The prototyping in WP2 started in year two while the game design activities in WP3 have proceeded since year one. The history of WP3 activities is not discussed, but only briefly mentioned to put year three concepts and prototypes in context.

The document is divided into the following sections:

- Section 1 “Introduction”: provides a description of the structure and scope of this document.

- Section 2 “WP2 games”: describes three games designed at a workshop in collaboration with project partners and developed during the past year. The games aim at training emotion regulation and managing the disposition effect, as well as create an awareness of the adherent biases.

- Section 3 “WP3 games”: describes the concepts and prototypes designed and developed during year to within WP3. The section also explains how the different concepts and prototypes have led to the design of the overall learning intervention, MINDswap.

- Section 4 “Conclusions”: summarises the chain of activities that led to creations of the different game concepts and games. It also conclude the present status of the different interventions and point out the direction for the third year.

1.2 List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BTH</td>
<td>Blekinge Tekniska Högskola</td>
</tr>
<tr>
<td>CIMNE</td>
<td>Centre Internacional de Mètodes Numèrics en Enginyeria</td>
</tr>
<tr>
<td>DoW</td>
<td>Description of Work</td>
</tr>
<tr>
<td>EC</td>
<td>European Commission</td>
</tr>
<tr>
<td>EUR</td>
<td>Erasmus University</td>
</tr>
<tr>
<td>FinCap</td>
<td>Financial capability</td>
</tr>
<tr>
<td>FZI</td>
<td>Forschungszentrum Informatik</td>
</tr>
<tr>
<td>GEQ</td>
<td>Game Experience Questionnaire</td>
</tr>
<tr>
<td>HUD</td>
<td>Head-up display</td>
</tr>
<tr>
<td>SUS</td>
<td>System Usability Scale</td>
</tr>
<tr>
<td>OU</td>
<td>Open University</td>
</tr>
<tr>
<td>TIG</td>
<td>Two Index Game</td>
</tr>
<tr>
<td>WP</td>
<td>Work Package</td>
</tr>
</tbody>
</table>
2 WP2 games

2.1 Overview of the design process in WP2

The design and development phase within WP2 started in year two as opposed to the design activities in WP3 that started in year one. In May 2010 frequent video conference meetings were held with the objective to reach a common understanding in WP2, WP4 and WP5 of what could be achieved within WP2 in terms of prototyping. The discussions were followed up by a two day workshop at Saxo Bank, in the beginning of July. During the workshop three different prototypes were specified. The objectives of the games are emotion regulation and reduction of the disposition effect. The three games are named Aiming game, Auction game and Two Index Game.

Three different product owners (partners from WP2) were assigned to the three games respectively. This was done to facilitate the participatory design process and to be able to get feedback from the product owner fast and frequently.

One development cycle lasts for about one calendar month. Each cycle starts with an elucidation of the new requirements and relevant literature is read. Then the development starts and the prototype must be runnable at all time. After about two calendar weeks the prototype is demonstrated to the product owner, and improvements and unsolved questions are discussed. The development process is highly collaborative and iterative and the developer contacts the product owner whenever needed to get quick feedback on functionality and appearance.

The main target group is investors at Saxo Bank. During the development phase the games will be tested by project members, colleagues and/or students to verify the functionality, usability, game play (i.e. game mechanisms), gameplay (i.e. players playing the game (Sennersten, 2009)) and players’ experience when playing the games. The potential impact of the game on the players’ emotion regulation capabilities or ability to handle the disposition effect will be assessed within the evaluation phase at EUR and FZI. The evaluation phase starts in May 2011.
2.2 Aiming game

2.2.1 Aiming Game I

2.2.1.1 Introduction

In the last few years, computer games have started to become valuable tools for different kinds of skill training. These types of games, or serious games, can be designed very differently depending on the type of training they are intended to provide. Simulation games generally try to replicate a real life scenario, such as pilot training and stock trading games, in order to give the player direct training and transferable skills. Some serious games, like the one described in this section, aim to train a specific skill in a game setting, which in turn is hypothesized to be transferable into a real world setting. Using serious games in this manner has the obvious advantage of being both cheap and risk-free in comparison to allowing practice in real life settings.

This section describes a serious game called the Aiming Game (2D), which is used to train players in identifying and controlling their own states of arousal. The Aiming Game is a two-dimensional shooter game where the player tries to aim and shoot down airplanes, as shown in figure 2.1.

Figure 2.1 – Screen shot of the Aiming Game.
2.2.1.1 Purpose

The main concern of the xDELIA project is to develop learning interventions for investors, particularly those using the Saxo Bank trading platform. We focus mainly on investors who meet the following criteria:

1. They trade their portfolio sufficiently often that systemic patterns and biases in their trading are detectible.
2. They trade on a regular basis through a trading platform.

The Aiming Game is a product in an intervention package specifically aimed at assisting investors in becoming aware of their own arousal state as well as training them in regulating their arousal. There is evidence to suggest that effective regulation of emotions can have positive effects on performance in investment and trading settings (Fenton-O’Creevy et al., 2010).

Successfully training investors in arousal regulation in a game environment is hypothesized to have a positive effect on their behavior, in a real trading environment. Since the Aiming Game is meant to be a training tool for investors which uses sensory technology to improve training effects, it would be beneficial to make it available through day trading centers where investors will have easy access to it.

2.2.1.2 Emotion Regulation

Being able to express emotions is one of the key attributes to being human yet we often want to suppress or ignore our emotions, perceiving them as being in the way of our goals. While both positive and negative emotions have shown to have an impact on performance and decision-making, the process of affecting negative emotions requires cognitive resources.

Emotions can generally be classified by the independent components arousal and valence (Russell, 1980), where arousal represents excitement level and valence defines whether the emotional state is positive or negative. This means that emotions can be visualized in a diagram where arousal and valence define each axis, as seen in figure 2.2.

![Figure 2.2 – Emotions in the valence-arousal space](image-url)
When attempting to measure emotions, one is thus actually measuring a combination of valence and arousal. There are methods for extracting and interpreting valence from e.g. Heart rate (Anttonnen & Surakka, 2005; Leng et al., 2007) and Electromyography (EMG) measuring devices (Cacioppo et al., 1986), but since there are several technical difficulties here such as accessibility and extensive setup procedures, as well as game balancing problems, the Aiming Game will not be concerned with valence for now. Instead the focus lies on arousal as the primary attribute of interest. Since arousal is vital in the definition of emotion, emotion regulation is basically the attempt to change state of arousal and valence.

When facing difficult and stressful tasks, people tend to use one of two main strategies to deal with the corresponding emotion (Wallace, 2009). These strategies are:

- Suppression
- Reappraisal

Suppressers generally tend to push down emotions, ignoring the fact that they exist and are continuously affecting them. Reappraisers however tend to positively reevaluate situations.

Both emotion regulation strategies exhaust cognitive resources for the person affected by the emotion (Wallace, 2009). Wallace et al. also points out that suppressing emotions generally takes up more cognitive resources than reappraiser do when reevaluating situations. Generally it is therefore preferred to apply reappraisal strategies when encountering unwanted emotions.

In order to identify emotion regulation strategies used by individuals, Gross et al. (2003) developed the Emotion Regulation Questionnaire (ERQ). The ERQ makes specific statements in regards to the emotion regulatory process intended to be measured such as “I control my emotions by changing the way I think about the situation I’m in”. The result of the ERQ can then be cross-correlated with results from a demanding task, such as the Aiming Game.

2.2.1.1.3 Research Questions provided by xDELIA

The evidence suggests that learning emotion regulation strategies and improving emotion regulation capabilities can increase performance in cognitively demanding tasks (Wallace, 2009). It is therefore desired to implement this type of training in a serious game and study whether it can have effects in first a game setting, but also if newly acquired emotion regulations skill and knowledge is transferable into a real world setting.

- Can the “Aiming Game” help players in learning to identify and experience their emotional state?
- Can the “Aiming Game” be used to improve players’ capability to regulate emotions?
- Can Emotion Regulation be trained in order to increase performance in stressful game tasks?
- Are emotion regulation capabilities, trained with the “Aiming Game”, transferable to real world setting?

2.2.1.1.4 Psychophysiological Data

While playing the game the player wears a wireless head-mounted device called Emotiv EPOC (www.emotiv.com). The EPOC is a combination between Electroencephalography (EEG) and Electromyography (EMG) which can extract the electrical signals produced by the brain and translate this into the instantaneous excitement, or arousal, of the player. The next iteration of the Aiming Game is planned to use Movisens EKG Move, a commercial ECG sensor with specialized software developed by partners within xDELIA.
This information is used as bio-feedback, meaning it is being fed back into the game, and is displayed in a bar on the screen (Figure 2.3), making the player become aware of his or her current arousal state. The arousal is divided into five segments, one being very low arousal (completely calm) and five being very high arousal (chaotic).

In addition to informing the player of his or her arousal state, the registered arousal is also used to distort some of the game elements, making the game harder to play when being in a high arousal state. The challenge is thus to stay calm to be able to aim effectively.

While constantly providing the player with real time bio-feedback and affecting game content depending on arousal, all game data is also logged to file to fully support analysis of both in-game actions and arousal over time afterwards. This allows the Aiming Game to be used as a tool for studies in addition to being a training environment for emotion regulation.
2.2.1.2 Description of the prototype

The Aiming Game is one piece in a learning intervention meant to make people, and particularly investors who trade on a regular basis, aware of their emotional state as well and giving them proper training in how to more efficiently regulate and control their emotions. While this occurs in a game related setting, we hypothesize that the skill learned in the Aiming game can be transferred into a differently setting, such as financial investments. This chapter describes the Aiming Game in regards to game play and its elements and how these are designed to fulfill its purpose.

2.2.1.2.1 Game Play

While game mechanics can be described as any part of the rule system of a game that covers one, and only one, possible kind of interaction that takes place during the game (Lundgren & Björk, 2003), the game play can be described as the process of learning the rules of a specific game.

This chapter describes the game play of the Aiming Game in terms of the flow and progress of the game and also discusses the different game elements which are used to enhance the game experience and fulfill the game requirements.

The Aiming Game is a two dimensional, first person shooter game where the main objective is to score as many points as possible by shooting down targets in the form of black airplanes. This is done by using a regular computer mouse as input device to aim at and shoot targets.
2.2.1.2.1.1 Phases

The core game consists of three levels or phases each lasting 180 seconds. The phases and their respective additional core mechanic are explained in the table below.

<table>
<thead>
<tr>
<th>Phase</th>
<th>Objectives</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 1</td>
<td>Shoot down targets</td>
<td>Targets (Black airplanes)</td>
</tr>
<tr>
<td>Phase 2</td>
<td>Shoot down targets</td>
<td>Targets (Black airplanes)</td>
</tr>
<tr>
<td>Avoiding distractors</td>
<td>Visual distractions (Red airplanes)</td>
<td></td>
</tr>
<tr>
<td>Phase 3</td>
<td>Shoot down target</td>
<td>Targets (Black airplanes)</td>
</tr>
<tr>
<td>Avoiding distractors</td>
<td>Visual distractions (Red airplanes)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Auditory distraction</td>
</tr>
</tbody>
</table>

![Figure 2.5 – The Aiming Game during Phase 1.](image1.jpg)

![Figure 2.6 – The Aiming Game during phase 2.](image2.jpg)

![Figure 2.7 – The Aiming Game during phase 3.](image3.jpg)

The first phase is basically an introduction to the core game mechanics. The player attempts to shoot down airplanes as they appear from outside the screen and rapidly moves across it. Targets are spawned once every 0.8 seconds.
In the second and third phase, visual distractions are added in the form of red airplanes. The goal is still to hit the black planes while then avoiding shooting down the distractions. The purpose of the distractors is solely to disturb the player and pressure him or her into making errors and thus becoming stressed. Distractors are spawned once every 0.4 seconds.

In the first version of the Aiming Game prototype specification the velocity and spawn frequency were exactly the same. According to an early heuristic evaluation (See 4.2 Heuristics) of the prototype, related to game challenge (Isbister & Schaffer, 2008), it was pointed out that the element of distraction had too small of an impact and was not challenging the players in a stimulating way (Gee, 2005). Adjustments to the visual distractor were made accordingly which resulted in the red airplanes moving 30% faster than the black ones and now also spawn 100% more often.

The third phase involves auditory distraction by adding stressful music. North & Hargreaves (2008) argue that music plays a role in task performance and showed that music and concurrent tasks competed for the same cognitive resources. In the Aiming Game the song Surfin’ Bird by the Thrashmen is used because of its stressful nature. The subjectively chosen music will be compared in a study to music generally accepted to induce stress, first listed by (Mayer et al., 1995), to analyze if there is a difference in average stress levels between groups. Hints at trends in this experiment may spark new studies in the future.

There is no limit to how many shots one can fire in an amount of time during the game. To ensure balance regardless of play styles when it comes to fire mechanics, a shot cost was implemented. This means that for every shot fired a score of two points is reduced from the players total score pool. Without this feature, it would become beneficial to shoot frantically without hesitation, consideration or strategy.

2.2.1.2.1.2 Arousal, Bio-feedback and Game Play

The registered EEG-information, provided by the EPOC is used as bio-feedback, i.e. being fed into the game and used as an input device (See 3.5 Measures and Data Logging). The information is used to create distractions in the game dependent on the player’s current arousal level, in two different ways:

- Distort aiming
- Apply blur to targets and distractors

The aiming is distorted by receiving an offset to its original position. This offset is constantly moving within the bounds of a predefined square (Figure 2.8) and the distance between the original position and the offset position is again directly related to the amount of arousal one is experiencing at a specific time.
In the same way, when the player becomes aroused, the targets start to become blurry, according to the same scale as with the aiming offset. The amount of blur affecting the airplanes is balanced so that it, at minimum arousal, exists no blur at all (figure 2.9) while at the maximum level of arousal it is hard to see the airplanes exact position as shown in figure 2.10.

2.2.1.2.2 Motivation
The issue one has to consider when developing serious games is that there may not always exist intrinsic motivation for playing the game or to invest much time in it, because of the motivation not being primary entertainment. In the case of the Aiming Game, and the investor target group, the desired impact is that the sought learning and skill training will be motivation enough to get investors to frequently use the platform or game.

Even though the Aiming Game is a relatively simple game in regards to game mechanics and features, it becomes vital that the elements that do exist both help and support the motivation to play the game. The main design goal of the game is for it to be challenging and thus also to allow players to practice mastery (Schell, 2008). Furthermore, Karat et al. (2000) claims that there can be great satisfaction in the ability to master one’s tools and produce a desired result and so are willing to invest a great deal of time in doing so. Offering challenge and the opportunity to master a skill therefore seems to provide great, and perhaps even sufficient, motivation for people to engage in games.

2.2.1.2.3 Game Logic and Elements
The Aiming Game mainly consists of the elements described in the diagram below (Figure 2.11). All elements described in the diagram are programmed application elements.
The game elements mostly revolve around the Shooter object, where most actions and calculations are performed. This is where bullets are generated when the player shoots by clicking the mouse and also where the mouse object is updated.

Each frame the Shooter object calls the Arousal Controller to request updates on the psychophysiological data collected from the EPOC-wearing player. The response will be a value between one and five which is then translated and sent to the mouse object where the aiming offset is applied accordingly. At maximum arousal level, the crosshair will receive an offset of approximately 15% of the screen width.

There are two object-generating entities working isolated from the rest of the scene called Target Spawner and Visual Distractor Spawner. These entities run on predefined timers and spawn (generate) their respective child objects (Targets and Distractors) according to the timer intervals. Targets are generated with a frequency of 1.25, and Distractors with a frequency of 2.5.

For each frame, the Collision detection object compares Bullet objects to both the Distractor objects and the Target objects to identify collisions, meaning that the bullet actually hit an airplane. The Collision detection object then calls the appropriate actions such as explosion animations, sounds and score adjustments.

Regardless of the rest of the scene, there is a timer object counting down from a predefined time, thus keeping track of when to change levels as well as a Score Counter which, during all scenes, collects all score data. Score is calculated by the following criteria:

- Shot: -2 points
- Target hit: +10 points
- Distractor hit: -10 points
Connected to the Timer is also an Audio Player. This is a simple entity which becomes active in the third phase and controls the background music meant to distract the player.
2.2.1.3 Technical Description

The Technical Description is a constant work-in-progress during the prototype development phase and describes the current state of the Aiming Game prototype. The purpose of this chapter is to provide a more detailed description and in-depth discussions of the specific technical game elements and design solutions of the Aiming Game.

2.2.1.3.1 Prototype Specification in Summary

Each prototype should be possible to explain with a few lines of text. This summary table is frequently used with game prototypes developed by xDELIA partners at BTH. The intention of the table is to, in a standardized and familiar way, be able to quickly acquire a basic understanding of the purpose of the game.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Aiming game (2D)</td>
</tr>
<tr>
<td>Version</td>
<td>1.2</td>
</tr>
<tr>
<td>Status</td>
<td>Ongoing</td>
</tr>
<tr>
<td>Target User Group</td>
<td>Investors</td>
</tr>
<tr>
<td>Prototype User Group</td>
<td>Internal + students</td>
</tr>
<tr>
<td>Purpose</td>
<td>To train emotion regulation, but also to teach players to become aware of the arising emotions.</td>
</tr>
<tr>
<td>Description</td>
<td>The player aims at a target and tries to hit the bull’s-eye. The player wears an EEG-sensor that feeds the players emotional state into the game in real time. Dependent of the level of arousal the target gets unstable and blurred.</td>
</tr>
<tr>
<td>Training Principle</td>
<td>Emotion Regulation</td>
</tr>
<tr>
<td>Transfer</td>
<td>Unknown for now.</td>
</tr>
<tr>
<td>Context of Use</td>
<td>Part of a training platform available on Saxo Bank</td>
</tr>
<tr>
<td>Appearance</td>
<td>The first level contains only targets. The second level adds distractions in form of visual effects and the third level add even more distractions like sounds and noises.</td>
</tr>
<tr>
<td>Inputs (inc. sensors)</td>
<td>Mouse and Emotiv EPOC and in future versions Movisens EKG Move.</td>
</tr>
<tr>
<td>Feedback</td>
<td>The player’s emotional state will be shown by an indicator of one of five levels (very cool, cool, average, excited and panic).</td>
</tr>
<tr>
<td>Guidance</td>
<td>Before the player starts to play he or she has to read some instructions of how to play and what the purpose of the game is. It is possible to skip the instructions to facilitate for players that play the game more than once.</td>
</tr>
<tr>
<td>Gameplay/ Challenges</td>
<td>To keep calm to minimize the distortion and thereby have a better chance to hit the target.</td>
</tr>
<tr>
<td>Emotional reporting</td>
<td>Level of arousal</td>
</tr>
<tr>
<td>Human instructor</td>
<td>None</td>
</tr>
</tbody>
</table>
Social network | None
---|---
Logging devices | Cumulative score
 | Logging of emotional state.
Development environment | Unity 3D Pro
Platform | PC
Infrastructure | Stand-alone application.
Testing | Functionality and usability verification of gameplay, scoring and feedback, heuristic evaluation of gameplay.

2.2.1.3.2 Implementation

Development of the Aiming Game was scheduled to span over one iteration period, i.e. one month, starting on September 2nd 2010. The initial estimation of time to finish the game prototype were in total 7 ½ days (60 hours), provided that the integration of the sensor equipment went smoothly. At this stage, time estimations were done per level according to table 2.1:

<table>
<thead>
<tr>
<th>Level</th>
<th>Estimated Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>3 days</td>
</tr>
<tr>
<td>Level 2</td>
<td>4½ days</td>
</tr>
<tr>
<td>Level 3</td>
<td>5½</td>
</tr>
</tbody>
</table>

*Table 2.1 – Initial estimation of required time to finish first prototype cycle.*

Initially a Work Breakdown Structure (WBS) (Evans, 1994) was created by detecting all involved work items and tasks from the original specification. Each work item was estimated in regards to required time for completion. Below in table 2.2, a more detailed list over the specific work items is provided.

<table>
<thead>
<tr>
<th>Work Item #</th>
<th>Description/Name</th>
<th>Est. Time Needed (h)</th>
<th>Time Spent (h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cross Hair Controller</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Target Controller</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Input Controller</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>Bullet Controller</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Biofeedback/Cross Hair Noise Controller</td>
<td>8</td>
<td>24</td>
</tr>
<tr>
<td>6</td>
<td>Arousal Level Presenter</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>Score Controller</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>Visual Distraction / Distraction Controller</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>Audio Distraction / Audio Controller</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>Moving targets</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>11</td>
<td>Several targets/ Target Spawn Controller</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>Blurred targets</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>13</td>
<td>Connecting features and tweaking</td>
<td>Remaining time</td>
<td>-</td>
</tr>
</tbody>
</table>

*Table 2.2 – Breakdown of specification into work items.*
The first iteration was expected to last for approximately 50 hours for the combined time of the work items. Due to several unexpected circumstances, particularly regarding bio-feedback and collaboration between the EPOC device and the game engine, this time frame had to be extended to approximately 65 hours.

2.2.1.3.3 Technology
The Aiming Game is developed in the game development environment Unity 3D Pro. The choice of using Unity 3D was taken by xDELIA partners at BTH after agreement that all prototypes should be developed in the same environment. It was also agreed upon that all prototypes should be developed in a generic way allowing for easy customizations according to studies and changes in specifications.

The Aiming Game is actually similar to a regular 2D shooter game where all targets and other visual content is represented in 2D, via 2D textures. Developing a 2D game in a 3D environment has its downsides, becoming unnecessarily complex for such a simple game, but at the same time having the strength of being easily upgraded to a more complex game.

The Aiming Game is neither graphically nor computationally demanding, rendering most computer systems able to run it. Exact hardware requirements has not yet been established but will be accounted for when the prototype is closer to the final product.

2.2.1.3.4 System Description
This chapter presents the Aiming Game prototype from a technical perspective, showing its system structure and interaction structure with the use of Use cases, Class Diagrams, Sequence Diagrams and descriptions of all major components.

2.2.1.3.4.1 Use Cases
Use Cases can be very informative when describing game prototypes. This chapter presents two illustrations whereas the first one is a traditional Use Case describing the different actions the play can take (Figure 2.12) while the second is a none-traditional illustration describing the system as different interactions involving the player, the system and an experimenter (2.13).
While playing the Aiming Game the player is very limited in the amount of different action available. Once the game is initiated there is basically only one direct action available which is shooting bullet. The consequences of shooting can however be sub-categorized into shooting targets and shooting distractors which creates different outcomes. Meanwhile, the passive actions which affect the game is game state changes depending on the EPOC data readings. There are two main actions from the EPOC which change the state of the game: Target blurring and Arousal Bar changing.

Figure 2.10 – Use Case of a player using the Aiming Game
The diagram above in Figure 2.13 represents the Aiming Game in the context of a simple straight forward study. The player plays the game while wearing the EPOC. The game progresses through the three phases while being constantly fed with information from the EPOC-wearing player. The game in its turn writes two files to the disk for each phase which afterwards can be analyzed by the experimenter.

2.2.1.3.4.2 Class Diagram and Component Description

This chapter describes the game system classes and the different objects needed. In the class diagram, in order to receive a general overview, all elementary attributes and functions has been excluded to visualize the more important parts of the system.
**Shooter**

The Shooter object involves all aiming and shooting mechanics. It contains a mouse object, which can apply the offset algorithm to the crosshair, making it move around during non-zero arousal levels. The Shooter also calls the Music Player to play sound effects when a shot is fired.

**Bullet**

The Bullet is a simple component containing only a three-dimensional position and a velocity. It is instantiated at the position of the cursor at the time of a fired shot with z-position=0. Its position is then updated each frame, moving it further into the screen. If it collides with another object or has lived for more than one second, it is destroyed.
Target Spawner
This entity exists throughout all phases. The Target Spawner takes care of generating new Target objects according to specific criteria. A new Target is generated once every 0.8 seconds at any position just outside the visual screen space. The Target Spawner then defines, for each new Target, its velocity and lifespan.

Target
Targets are collision-detectable objects, the form of black airplanes, generated by the Target Spawner. Targets are given a lifespan from the Target Spawner. Once the lifespan reaches zero, the Target is destroyed. The lifespan is defined in such a way that it is able to travel across the entire screen before being destroyed.

Visual Distraction Spawner
This entity only exists through phase two and three. Its purpose is to generate new visual distractions at specific intervals. The current version of the Aiming Game defines the Visual Distraction Spawner in such a way that it generates one new distraction once every 0.4 seconds. It follows the exact same logic as the Target Spawner regarding spawn points and lifespan to allow visual distractions to behave in the same manner as targets.

Visual Distraction
Visual Distractions (Distractors) are collision-detectable objects, the form of red airplanes, generated by the Visual Distraction Spawner. Distractors are given a lifespan from the Visual Distraction Spawner in the same way that Targets are from the Target Spawner. Once the lifespan of a Distractor reaches zero, it is destroyed. The lifespan is defined in such a way that it is able to travel across the entire screen before being destroyed.

Timer
The Timer is a simple object which keeps track of the time since the phase started and also switches between phases according to a predefined time value (180 seconds for each phase). This object also calls the File Writer at the end of each phase as well as the Music Player in the beginning of the last phase to get background music.

File Writer
This object is called by the Timer at the end of each phase. It then fetches all collected information regarding arousal over time from the Arousal Controller object and Shot Statistics over time from the Shooter object. Examples of file formats can be found in the following chapter “Measurements and Data Logging”.

Music Player
The Music Player contains references to both music and sound effects, which are instantiated when called from other objects.

Arousal Controller
The Arousal Controller is directly listening to the port connected to the Emotiv EPOC device. For each frame during the entire game it registers the current arousal level and rescales it from a value from zero to one, to a value between one and five. This value is then presented in a graphical arousal bar with the same scaling and also fed to the Shooter object to affect the aiming offset accordingly.

Collision Detection
The Collision Detection is handled very much by the development environment itself, meaning collisions are not programmed by the prototype developer. In Unity 3D, there are predefined algorithms for handling collision so the only thing that the Aiming Game Collision Detection object does is delivering the correct object the collision test. For each frame it takes all active bullets and compares these with first Target objects and then also Visual Distraction objects. If collisions are detected, the Score counter is informed which in its turn, translates the outcome into score addition or reduction. Important to notice is that the game actually interacts in a three-dimensional space meaning that the collision are also handled and calculated in 3D. This design choice was made to allow a generic design which easily could be translated into a full 3D game.
2.2.1.3.5 Measures and Data Logging

This chapter discusses the handling of data collected in the Aiming Game. It is divided into two sections; Psychophysiological Measurements which discusses the handling of arousal data, and In-Game Data Logging which involves the collection of data gathered from player actions while playing the game.

2.2.1.3.5.1 Psychophysiological Measurements

For the first iterations of the development of this prototype, the EPOC will be used. The EPOC is able to register arousal, and summarizing these signals into a value called instantaneous excitement, which can be translated into arousal. There already exists an Application Programming Interface (API) of the EPOC device, freeing prototype developers from having to create such a library. This makes the EPOC a suitable and convenient tool to use.

The EPOC will be substituted with a mobile ECG sensor developed by Movisens, which will result in improved measurement accuracy compared to the EPOC. Moreover, Forschungszentrum Informatik will provide a software framework which connects the sensor with the game and computes arousal information from the data obtained from the sensor.

For each phase, a .txt log file is created which contains the psychophysiological data collected from the EPOC at each frame during the game (Table 2.3). This data is stored in relation to time, meaning that each value collected by the EPOC is linked with a time variable which represents the exact time when the value was collected. Therefore graphs can be created easily, showing arousal in relation to time. It is also possible to compare arousal over different phases using this method. Due to convenience in the current development environment (Unity 3D), statistical data is not stored in an external text file until the end of each phase. Instead, the data is temporarily stored in a large array in-game and then exported to a text file when the phase time limit has been reached. Because of the large quantity of data being exported, this creates a small delay, depending on the capacity of the computer, at the shift between phases. In addition, if a phase should be interrupted for any reason, data for that phase will be irretrievable.

<table>
<thead>
<tr>
<th>Time stamp (seconds from start)</th>
<th>Arousal (1 = very low, 5 = very high)</th>
</tr>
</thead>
<tbody>
<tr>
<td>54.44</td>
<td>3</td>
</tr>
<tr>
<td>54.67</td>
<td>2</td>
</tr>
<tr>
<td>54.88</td>
<td>3</td>
</tr>
<tr>
<td>55.10</td>
<td>4</td>
</tr>
<tr>
<td>55.38</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 2.3 – Example of Arousal Statistics text file.

2.2.1.3.5.2 In-Game Data Logging

Each time a shot is fired this is registered in a separate .txt log file. For each shot fired a time stamp is also collected and linked to the shot. The application then investigates what the outcome of the shot was, i.e. hit, miss, distractor hit, and then links the outcome with the shot and time stamp. Shot statistics can thus be generated regarding both time and arousal. In Table 2.4 below, an example is presented which shows what such a file can look like.

<table>
<thead>
<tr>
<th>Time of fired shot (seconds from start)</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.33</td>
<td>Hit</td>
</tr>
<tr>
<td>7.24</td>
<td>Miss</td>
</tr>
<tr>
<td>11.54</td>
<td>Miss</td>
</tr>
<tr>
<td>19.99</td>
<td>Distractor hit</td>
</tr>
<tr>
<td>25.48</td>
<td>Hit</td>
</tr>
</tbody>
</table>

Table 2.4 – Example of Shot Statistics text file

In-game data is what ultimately represents performance in the Aiming Game. Performance can however be calculated in several different ways in the case of this game. Since shot statistics is gathered for different
arousal levels it becomes difficult to fairly calculate game performance without also including emotion regulation performance. When being in a relaxed state (or stressed state in the fourth phase) the game becomes easier which allows for better performance in theory. It is therefore also valuable to analyze performance in individual arousal levels by cross reference the shot statistics with the arousal statistics.

2.2.1.3.6 Prototype Design

The complexity of design and functionality requirements in software development as well as game development calls for a systematic approach in the initial phase to ensure that all requirements have been accounted for (Stellman & Greene, 2005). In the development of the Aiming Game all requirements regarding both graphical design and functionality should have an explicit design solution. In this chapter the requirements are presented as they were interpreted by developers and agreed upon among involved partners. Each requirement is then rewritten as one or several design solutions which are required in order to fulfill that specific requirement.

2.2.1.3.6.1 Requirements, Design Solutions and Issues

The purpose of the requirements is to cover all the desired design- and functional features in an application. A requirement should be stated in such a way that it can be easily explained by a set of design solutions. Requirements are listed with unique ID numbers as an abbreviation of R_ID, Category and Sub-category, such as R_ID12.3. Following each requirement, the design solutions for that specific requirement is listed as well as issues associated with the designs. The design items in are described by unique design ID, reference to the specific requirement in the requirement list, as well as a description of how the requirement was ultimately implemented.

The requirements are listed with all follow-up requirements which are dependent on some other requirement.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:1.1</td>
<td>Game support for a mobile ECG-device</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:1</td>
<td>This design solution is not implemented. See Issue I_ID:1 for more information.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ID:1</td>
</tr>
<tr>
<td>The original plan was to use the mobile ECG-device developed by Movisens. In order not having to wait for this product to be completely finalized before testing out the prototype series, the device was temporarily substituted for the Emotiv EPOC device.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:1.2</td>
<td>Game supports the Emotiv EPOC</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:2</td>
<td>The use of the EPOC was a solution to not having to wait for the mobile ECG-device but rather be able to start developing immediately. The support for the EPOC was implemented in the Arousal Controller object which constantly listens to the same port as the EPOC communicates to.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue ID</th>
</tr>
</thead>
</table>
The Aiming Game does not register when there is an error with the EPOC which limits or blocks signals to enter the game. It is currently being looked into how this issue can be solved in a generic way so that it is applicable to all game prototypes.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:1.3</td>
<td>Mouse should be used as input device</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:3</td>
<td>The Aiming Game is developed in the Unity 3D environment. This platform already supports mouse input so the only thing that had to be done was to map mouse buttons to certain actions.</td>
</tr>
</tbody>
</table>

**Requirement ID**

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:2.1</td>
<td>Support for two-dimensional aiming mechanics</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:4</td>
<td>Since the development environment is Unity 3D, which obviously is a 3D environment, the camera will by default have a three-dimensional “field-of-view”. The camera object therefore has to be set to Orthographical perspective, removing the 3d visual representation. However, all objects still exist in a three-dimensional environment which must be considered when performing calculations and developing game logic. The use of only GUI items (2D) was proposed and tried out but did not effectively solve the solution.</td>
</tr>
</tbody>
</table>

**Requirement ID**

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.1</td>
<td>Support for Targets</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:5</td>
<td>The Target object must consist of texture, lifespan and spawn position.</td>
</tr>
</tbody>
</table>

**Requirement ID**

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.2</td>
<td>Support for moving targets</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:6</td>
<td>Variables “current position” and “velocity” have to be added to Target object.</td>
</tr>
</tbody>
</table>

**Requirement ID**

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.3</td>
<td>Support for several Targets</td>
</tr>
<tr>
<td>Design Solution ID</td>
<td>Design Solution Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>D_ID:7</td>
<td>Implement a Target Spawner-object which can generate Target object in accordance with specific predefined criteria.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.4</td>
<td>Support for blur effects on targets</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:8</td>
<td>Unity 3D has a plug-in which creates blur on the camera screen. This does not blur the specific targets but rather everything that the camera perceives.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue ID</th>
<th>Issue</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ID:3</td>
<td>This might become an issue in the game, should it be decided to expand it in some unexpected direction. This because of the way the blurring is implemented as of right now. Instead of blurring specific objects, blur is currently applied to everything the camera perceives. This will have to be investigated since it is possible that blur could be applied to specific objects as well.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.5</td>
<td>Support for shooting down targets</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:9</td>
<td>There has to be an object called Bullet, which is instantiated by a Shooter objects.</td>
</tr>
<tr>
<td>D_ID:10</td>
<td>Collision Detection is needed to test bullets with targets to determine hits and misses. This is supported in Unity 3D so this object only need to define the bullets and the targets as input to the collision handler.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.6</td>
<td>Visual feedback of target hit</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:11</td>
<td>There is a built in explosion animation which can be used when the collision detection detects a bullet hitting a target.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:4.1</td>
<td>The game should consist of three levels</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:12</td>
<td>Three scenes have to be created in the game environment, Unity 3D.</td>
</tr>
<tr>
<td>Requirement ID</td>
<td>Requirement Description</td>
</tr>
<tr>
<td>----------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>R_ID:4.2</td>
<td>Each phase should increase stressful elements</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:13</td>
<td>Each scene in Unity 3D can easily contain different game elements.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:5.1</td>
<td>The player’s arousal should be visualized</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:14</td>
<td>An arousal bar should be implemented and displayed in the Graphical User Interface.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue ID</th>
<th>Issue</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ID:4</td>
<td>Heuristics and play testing showed that some players did not pay enough attention to the feedback. The Arousal bar will therefore be increased in size as well as in an alternative version be presented in a different manner. Further play testing will have to evaluate if the changes had a positive effect on the feedback perception.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:5.2</td>
<td>Arousal should consist of 5 levels</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:15</td>
<td>The arousal received from the Arousal Controller should already be normalized between 1 and 5 and converted into an integer by using the “floor”-function.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:6.1</td>
<td>Keeping calm will stabilize the targets</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:16</td>
<td>Arousal collected by the Arousal Controller should be connected to the camera object so that the blur depends on the arousal.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:7.1</td>
<td>Data logging of arousal over time</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:17</td>
<td>A file writer should be implemented which in real time, or after each phase, writes the arousal value collected from the EPOC in regards to time.</td>
</tr>
<tr>
<td>Requirement ID</td>
<td>Requirement Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>R_ID:8.1</td>
<td>Visual distractions</td>
</tr>
</tbody>
</table>

**Design Solution ID**  
**Design Solution Description**  
D_ID:18  
Visual Distractions could inherit most of the features of the Target object and modified so that hitting these affects score counting and sound effects.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:9.1</td>
<td>Support for sound and music</td>
</tr>
</tbody>
</table>

**Design Solution ID**  
**Design Solution Description**  
D_ID:19  
A Music Player need to be implemented which contains paths to specific sound sources. This needs to be connected to the Collision Detection object as well as the Shooter object to allow these objects to play their respective sound effects.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:9.2</td>
<td>Auditory distractions</td>
</tr>
</tbody>
</table>

**Design Solution ID**  
**Design Solution Description**  
D_ID:20  
The Music Player needs to be available in a specific phase to allow for audio distraction in the form of music.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:10.1</td>
<td>Time limits</td>
</tr>
</tbody>
</table>

**Design Solution ID**  
**Design Solution Description**  
D_ID:21  
Each phase must contain a Timer object which keeps track of time and gives necessary commands to other objects.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:11.1</td>
<td>Support for score counting and presentation</td>
</tr>
</tbody>
</table>

**Design Solution ID**  
**Design Solution Description**  
D_ID:22  
There must exist a Score Counter object which is fed with information regarding shots and collisions. This object must then be defined in such a way that in-game actions and effects has a numerical meaning in regards to score counting.

D_ID:23  
There must exist a graphical representation of the score in the GUI. Unity 3D
supports 2D labels which fit this requirement perfectly. The score will be presented in the bottom right corner.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:12.1</td>
<td>Support for two-player mode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:24</td>
<td>This design was not implemented. See Issue I_ID:4 for more information.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue ID</th>
<th>Issue</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ID:5</td>
<td>The Emotiv software does not support multiple EPOCs connected to one computer at the same time. The desire to have a two-player mode was therefore abandoned.</td>
</tr>
</tbody>
</table>
2.2.1.4 Evaluation

The evaluation of the Aiming Game consists of playability and usability testing using Heuristic evaluation as well as play testing by students. Usability refers to user interfaces and how helpful the game is in providing the player with necessary information and guidance while playability analyzes the actual game play and how well it can be used in terms of game flow.

First the Heuristic evaluation design is presented, describing the process of the heuristics, followed by results and improvements.

2.2.1.4.1 Data gathering and analysis

The incentive for data collection in the Aiming Game prototype is twofold. The first and most important reason is being able to present relevant data to the player in real time. Malone (1982) stresses the importance of that players always should be able to identify their score or progress in the game. At the same time, the game interface should be as non-intrusive as possible not to interfere with the player’s attention. The components which are necessary to visually represent are:

- Real time arousal value
- Score

These components are placed in the bottom of the screen; arousal value to the left and score presenter to the right. In the Auction Game prototype (see Auction Game Specification), the arousal bar was placed in the top of the screen. Developers wanted to identify which solution was the most accessible to players, i.e. which one was being watched the most.

An issue which unfolded during the heuristic evaluation of the prototype (see 4.2 Heuristics) was that both components were too small and insignificant, resulting in some players not paying enough attention to the feedback. To resolve this, measures are taken to improve the visual feedback of the consequences of the in-game actions. This issue however goes against Play Testing results (See 4.3 Play Testing) where players suggested that the arousal bar was actually sufficient in size and could be kept unchanged. In the next iteration, hitting targets, missing and hitting distractors will give direct graphical and numerical feedback at the specific position of the event. The score presenter will also be adjusted in size by a factor of two.

The second incentive to gather data is to be able to perform analysis of the participant performance regarding both score and success rate in emotion regulation. Data is therefore stored in two separate files, namely arousal statistics and shot statistics, for each phase and participant. The data that needs to be collected for each phase is:

- Participant ID
- Phase
- Play time this phase (in seconds)
- Number of samples
- Sampling frequency
- Arousal value in regards to time
- Shots and consequences in regards to time
- Total shots fired
- Total hits

The two data files are uniquely identified by Participant ID to ensure complete anonymity. The samples related to time are collected each frame to ensure that there is not a lack of data. Each sample contains a time stamp and an arousal value. The arousal value in the text file ranges from one to five, one being very low arousal and five being the maximal arousal the EPOC device is able to register. In order to analyze changes in arousal over time in regards to in-game actions we also store shot statistics which for each action, stores the time and outcome of that specific action. This can show how temporary failures (misses
and distractor hits) are portrayed by changes in arousal, by correlating information from arousal statistics and shot statistics.

2.2.1.4.2 Heuristic evaluation

The game development iteration was followed by a heuristic evaluation which aims at qualitatively identify design errors, and suggest improvements to them (Desurvire, 2004). The Heuristic Evaluation used on the Aiming Game is part of a generic evaluation tool kit which is being developed by xDELIA partners at Blekinge Tekniska Högskola and is used on all prototypes produced by the xDELIA partners at BTH. The Heuristics are divided into a set of categories inspecting different aspects of the game prototype. There are several more Heuristics to the evaluator’s arsenal but this chapter only discusses the ones that actually had an impact on the evaluation of the Aiming Game. The framework and evaluation results are presented below.

2.2.1.4.2.1 Evaluation process

The heuristic evaluation requires three evaluators. It is desired that the three evaluators have competence both in games and in usability. The process of evaluating a game prototype using heuristics can be described as:

**Step 1:** The evaluators evaluate the game separately. No collaboration is allowed. The list of the heuristics is distributed to the evaluators. The evaluators describe the issues violating each heuristic in the list. The task is to just find the problems. How to deal with the problem is left to the development team.

**Step 2:** When the first step is completed the evaluators meet and put together their list of issues. If two evaluators have the same issue on their lists the problem stays on the final list. All the issues are discussed and only if one has some issue and the others can agree that the issue is a fact then that problem stays on the list well.

**Step 3:** A report is prepared describing the issues in more detail that it can be done in the list. Screen shots should be used to clarify the issues.

**Step 4:** The evaluators together with the developer discusses possible solutions and the suggestions are compiled into recommendations and added to the documentation.

**Step 5:** After the heuristic evaluation is conducted and documented, the result is presented to the product owner and in collaboration decisions regarding what to do with the recommendations and if a new iteration should start or if play testing should be conducted.

2.2.1.4.2.2 Heuristics Description, Outcome and Improvements

Heuristic evaluations of prototypes, in rather early stages, is most definitely going to identify a lot of issues both related to game play and graphical design choices. In this section the different findings from the Heuristic Evaluation are discussed briefly and where it is due, solutions are suggested. Heuristics in the table below have not been edited and are presented as they were originally written. They are uniquely identified by a Heuristic ID according to:

$H_{ID}\#_{CATEGORY}$
<table>
<thead>
<tr>
<th>H_ID1_CONSISTENCY</th>
<th>Arousal bar should only exist during game play. It should therefore be removed during questionnaires, in-game questions, tutorials and other similar parts.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Arousal level bar is shown during menus throughout the game.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID2_CONSISTENCY</th>
<th>The crosshair should follow the outlines of the screen at all times. If the offset algorithm instructs the crosshair object to move to a position which lies outside of the screen borders, the position should be unchanged.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Mouse pointer gets capped away when aiming at the edges of the screen. The reason for this is that when players get highly aroused, mouse automatically moves away from the actual place we are aiming. This bug puts the player in a disadvantage since he or she cannot shoot the whole screen.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID3_CONSISTENCY</th>
<th>Bullets are shot from the camera into the screen and should not move in x- or y-position. Revision of the Shooter-object is needed to identify the issue causing this bug.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sometimes bullets slide across the screen when you shoot them, away from the point you shoot.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID4_CONSISTENCY</th>
<th>Menu buttons are supposed to be in normal state when passive and become highlighted when the cursor enters their target area.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Menu buttons are lighted in a non-standard way when sliding across them; grey when mouse is over them and lighted when it is not.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID5_CONSISTENCY</th>
<th>Mapping the Enter-key to any input field will solve this inconvenience.</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the text input boxes in menu screens, Enter key should work so that when user inputs the arousal level he can press enter and continue. Right now the player has to manually press the continue button.</td>
<td></td>
</tr>
</tbody>
</table>
**H_ID6_FEEDBACK**

<table>
<thead>
<tr>
<th>On the text input boxes in menu screens, Enter key should work so that when user inputs the arousal level he can press enter and continue. Right now the player has to manually press the continue button.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping the Enter-key to any input field will solve this inconvenience.</td>
</tr>
</tbody>
</table>

**H_ID7_FEEDBACK**

<table>
<thead>
<tr>
<th>Arousal level bar should be continuous.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Letting the arousal presenter bar have a higher sampling rate is expected to give the player more fine-grained, and thus better, feedback on the changes in their arousal state. This might help the face validity of the presented feedback.</td>
</tr>
</tbody>
</table>

**H_ID8_FEEDBACK**

<table>
<thead>
<tr>
<th>Arousal level bar has to provide help pointers on what level the players are on; either by specifying numbers next to it, color or some other way so that user can exactly see on what level of arousal he is.</th>
</tr>
</thead>
<tbody>
<tr>
<td>In the state the arousal bar is right now, there is limited indication of values. The maximum size of the bar in relation to the current arousal state however hints at an estimation of that value. By mapping the bar with numbers, the player will have an easier time keeping track of their current arousal state.</td>
</tr>
</tbody>
</table>

**H_ID9_AVOID_ERRORS**

<table>
<thead>
<tr>
<th>User input in the menu text boxes should be restricted to numbers in specified range. Maybe even exchange textbox for a dropdown list. Right now user can input any faulty string.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measures should be taken to eliminate the risk of errors due to confusion. By forcing the player to choose from a set of alternatives instead of having free choice, the input will always be in the correct format.</td>
</tr>
</tbody>
</table>

**H_ID10_PROVIDE_HELP**

<table>
<thead>
<tr>
<th>Provide the players with the button to exit the game during play.</th>
</tr>
</thead>
<tbody>
<tr>
<td>The version of the Aiming Game which is meant to be a training platform will also provide a pause option from which the player is able to exit the game.</td>
</tr>
<tr>
<td>H_ID11_PROVIDE_HELP</td>
</tr>
<tr>
<td>---------------------</td>
</tr>
<tr>
<td>When a plane gets shot, provide the player with a help tip how many points he scored or lost. It can be easy as a small number on the explosion.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID12_PROVIDE_HELP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Players should have a help tip at introduction time on shooting lag. Reason is so that player knows he has to shoot in front of the plane.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID13_SCREEN_LAYOUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text on introduction menu isn’t fitted on screen. It gets cut off.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID14_SCREEN_LAYOUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score display is too small.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H_ID15_SCREEN_LAYOUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Help text in the introduction menu is redundant and unnecessary since you get it later in the game before the certain stage starts. It makes more sense on this place since the player will deal with this issue immediately after proceeding.</td>
</tr>
</tbody>
</table>
### H_ID16_SCREEN_LAYOUT

**Aiming is wrong at most resolutions but one; bullets end up next to the cursor instead of in the middle of it.**

There seem to be an issue where bullets are not always generated in the right position at certain resolutions. This has not affected studies since the resolution has been controlled but will be resolved in the next prototype iteration.

### H_ID17_SCREEN_LAYOUT

**Intro menu gets cut off on most resolutions.**

The prototype will be tested more thoroughly at different resolutions in the future.

### H_ID18_SCREEN_LAYOUT

**Screen background during play is cut off in high resolution (1900x1080) and most others.**

The method for representing backgrounds in the game will be different in the next iteration. Real sky-mapping will be used.

### H_ID19_SCREEN_LAYOUT

**Some planes appear in the middle of the screen (noticed on red planes stage). They should really fly from outside of screen, through the screen and back out again.**

This issue appears at certain high resolutions and will be resolved by using screen percentage instead of pixels.

### H_ID20_GAME_CONTROLS

**Bullet lag is something bothersome in the orthogonal shooter; because of orthogonal perspective player has no feedback in how far is the bullet from the plane or into the screen. It feels more like dropping bombs in front of the planes.**

As long as the game is represented in a 2D environment, the lag from shoot to hit will be removed. Should the game be converted into 3D, it might make sense to have a certain delay.

### H_ID21_AUDIO_VISUAL_REPRESENTATION

**Sound has to be adjusted outside of the game, so the player has to exit the game to do this. Maybe provide the**

This is a sensitive subject since sound is a vital part of the training process. The...
Volume control inside of the game separate for music and effects. | Heuristic will be processes and discussed more thoroughly before decisions will be taken.

**H_ID22_AUDIO_VISUAL_REPRESENTATION**

Music is so silent that it can't be heard from the shooting and explosions. | Adjustments to the music in relation to ambient sound and other sound effects will be made to ensure a more balance sound difference.

**H_ID23_CLEAR_GOAL**

The goal in this game is not motivating enough for a 10 – 12 min game; it becomes very dull after a while. | There should be other means to progress in the game except for time. Performance has been discussed as a viable variable for this.

**H_ID24_CLEAR_GOAL**

Game doesn’t provide player created goals (like a high-score list). | A high score system will be implemented which creates a possible incentive for some players to train and perform better.

**H_ID25_PROGRESS**

There is no comparing of results from various players. | See **H_ID24_CLEAR_GOAL**.

**H_ID26_PROGRESS**

Players have no idea on how much time is left until the end of a level or how far the level has progressed. Level progress indicator is missing. | No matter if the time countdown system persists or if it will be replaced by a performance-based level up system, the progress of the game will be visualized graphically so that the progress is clear to the player at all times.

**H_ID27_REWARDS**

Players are not rewarded at all. Provide achievements related to the score to motivate the player. | A high score system or an individual score comparison system should be implemented to...
<p>| give the player feedback on their performance. |</p>
<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>H_ID28_CHALLENGE_STRATEGY_PACE</td>
<td>Level progression in between levels is too slow. Make levels progress dependent on player’s performance. See H_ID23_CLEAR_GOAL.</td>
</tr>
<tr>
<td>H_ID29_CHALLENGE_STRATEGY_PACE</td>
<td>Aiming can be manipulated even at highest arousal; in between steps of making the aim off. If a player learns this he can still have a perfect score and shooting. Different technique in making the aim off is needed that a discrete one. The aiming mechanics will be changed to more represent a sniper rifle in any regular shooter game, such as Counter Strike. This makes it almost impossible to predict the offset and build up strategies to counteract this.</td>
</tr>
<tr>
<td>H_ID30_CHALLENGE_STRATEGY_PACE</td>
<td>Support for different playing strategies. For example, players who control emotions very well, increase the sensitivity of the emotion detection measurements and for the good shooters increase the speed of a plane. It is easy to implement a difficulty system which is based on the players themselves. The targetSpawnRate and distractorSpawnRate variables can be dependent on the score- or accuracy variables.</td>
</tr>
<tr>
<td>H_ID31_GAME_STORY</td>
<td>Make up a story to support the lag of shooting and connect it with the level progression (weapons, add-ons, etc.) This might be relevant if the game becomes transformed into a 3D game in the future. For now this Heuristic is referred to H_ID20_GAME_CONTROLS.</td>
</tr>
<tr>
<td>H_ID32_REPETITIVE_TASKS</td>
<td>Tasks in the game are repetitive. Additional object could be implemented to create a more varied game play. Objects which have some special bonus linked to it could for example be harder to hit.</td>
</tr>
</tbody>
</table>
H_ID33_REPETITIVE_TASKS
Introduce random event to break the repetition and introduce disturbance.
See H_ID32_REPETITIVE_TASKS.

H_ID34_DIFFERENT_PLAYING_STYLES
Game does not support different play styles. For example, accurate shooting against button mashing.
There is actually support for this in a sense, but since “button mashing” costs points it will most likely not be as effective as accurate shooting.

H_ID35_GAME_STAGNATION
Player has no feeling of progression. No feedback on how the game is developing.
See H_ID26_PROGRESS.

H_ID36_GAME_CONSISTANCY
Other game elements (audio for example) have to support change in affective state; so that when player gets aroused, the music and audio change as well.
This would have to be investigated further. If the literature discusses these types of effects, they could definitely (and easily) be implemented.

2.2.1.4.3 Play Testing
Play Testing is valuable since it allows different subjective players to analyze the game. This may produce novel aspects which has not previously been discussed or evaluated by the developer team.

2.2.1.4.3.1 Procedure
For the Play Testing Evaluation six arbitrarily chosen people will play the game. Before playing the game, demographic data considering gender, age and experience with similar digital games will be collected before the test starts. In order to objectively determine which game elements the players are paying attention to, the game will be played through the eye tracker. The purpose is to be able to tell how important different objects are, and in interviews afterwards, receive indications on how to improve the visual representations of the game.

Time Estimation
The total time estimation for the Play Testing is predicted as:
- Preparation 16 hours
- Play Testing 6 players á 1.5 hours = 8h
Step-by-Step Process

The process of the Play Testing of the Aiming Game can be described in 11 specific steps in accordance with the list below:

1. Recruit students
2. Set up
   Use six players for the play test (Pernice & Nielsen, 2009). The game and the sensor software should run on the eye tracker computer.
3. Introduction (Purpose + in-game-description)
4. Questionnaire (Appendix C). Check so that the questionnaire is properly filled in.
5. Play - During the play session the test leader observes the player and makes notes of the player’s behaviour. (Pernice & Nielsen, 2009)
6. GEQ (Appendix B) and SUS (Appendix C) (Nacke, 2009) (Nacke, 2010)
7. Closing questions. The interview section is recorded and the test leader also takes notes.
8. Analysis
   a. The results from the GEQ and SUS are arranged and analyzed.
   b. The gaze replay is analyzed and summarized. Gaze replay provide the best understanding of what goes on in the players head compared to other eye tracking analysis methods with 6 players ((Pernice & Nielsen, 2009). Areas of Interest may also be a good tool if there are things, like a bio feedback bar, of interest to explore.
   c. The closing questions are summarized.
   d. A final summary draws conclusions from the results from A-C to answer the question “What does this mean for the game design?”
   e. Analysis of the logged data. “Does the players’ performance improve from play session to play session?” and “How well did the players manage their emotions/biases from play session to play session?”
9. After the play test is conducted and documented the result is presented to the product owner and the pedagogue, and in collaboration decisions regarding what to do with the results are discussed.
10. The discussion results in a prioritized product backlog.

2.2.1.4.3.2 Results and Discussion

The Play Testing evaluation generated much valuable data which can be used in order to improve the Aiming Game both as an entertainment platform and as a learning tool. In addition to answering a generic questionnaire regarding Game Experience, participants were also interviewed regarding specific topics which the experimenter/developer needed answers for. The interview questions usually became open discussions regarding the specific topics which in turn generated additional useful data. The topics discusses with participants were:

- Arousal bar and Bio-feedback
- Aiming Mechanics
- Flow and Progression
- Difficulty
- Music
- Other (open discussion to let participant speak freely regarding the game)

An observation regarding the Arousal Bar was that five out of six participants claimed to have had awareness of their own arousal with the help of the arousal bar. Eye Tracking results however indicate that none of the participant paid any (or very little) direct attention at all to the arousal bar during the game. This finding might indicate that players are able to perceive the arousal bar in their peripheral view while playing the game that direct attention focused on the bar itself is unnecessary. Another possible explanation
for the participant claiming that they at most times had full awareness of the level of the arousal bar might be that they received this information in other forms. This hypothesis is based on the fact that players receive several indications of their arousal in the form of airplane blurring and crosshair offsets as well their own gut feeling. This phenomenon will have to be investigated further.

In general participants answered uniformly in the discussing of the Aiming Mechanics. All participants experienced the aiming as being somewhat rough and several participants drew the analogy to an old mouse with wheel-mechanics. It turned out that there was a bug in the aiming class update section preventing smooth movement in certain arousal levels. This issue has been completely capsulated and will be resolved in the next iteration.

Most participants (five out of six) also had a problem with the delay between shot fired and actually hitting the targets. Because of this game element, players had to learn how it worked before being able to hit targets correctly. Since the game does not explicitly explain the phenomenon, this caused much confusion among participants. In a new version of the Aiming Game this effect would be removed and replaced by an instant shoot/hit mechanic.

Several participants described the development of tactics throughout the game and how these changed in accordance with the changes between phases. The most common tactic seemed to be to focus the attention towards the middle where evidently the most planes eventually appeared. This tactic was shattered however when the second phase was initiated and red distraction airplanes appeared. Since these come in greater number with the same generation procedure as the targets, the red airplanes swarm in the middle of the screen making it very hard for the player to separate the targets from the distractors in this region. This led to players completely switching tactics and focusing their attention to the borders of the screen instead, something that can be seen relatively clearly in the Eye Tracker data of some participants during the last phases.

When asked what their suggestion to make the game more interesting to play would be, all participants talked about the repetitiveness which eventually leads to boredom in the game and suggested different ways to vary the game content. Suggestions to decrease repetitiveness were:

- Additional airplanes with different features such as blue planes which rewarded with more points.
- Reaching new levels by performance instead of time to increase the incentive to achieve better.
- Several graphically different levels
- A variation of many games with the same mechanics but different game manifestations, e.g. click-and-drag items to different places.

From the Game Experience Questionnaire (Nacke, 2009) (Appendix B) answers, it was possible to extract uniform trends regarding several components. In order to gain statistical significance from the GEQ one would preferably need approximately 20-30 participants. In the case of the Play Testing of the Aiming Game, which only used six participants, this will however not be possible. In this section several components will be discussed which stood out from the crowd and should be taken into consideration.

Participants generally answered that they felt tense during the game (Average: 4, Standard deviation: 0.89) as well as having to constantly focus on the game (Average: 4.17, Standard deviation: 0.41). They also congruently answered that they were not allowed to explore things (Average: 1.5, Standard deviation: 0.84) and did not feel very imaginative (Average: 2, Standard deviation: 1.09).

Participants answered that they did not feel happy during the game (Average: 2, Standard deviation: 0.63) at the same time as stating that they were not particularly bored (Average: 1.83, Standard deviation: 1.17).
2.2.1.5 Reflections

The Aiming Game was developed to assist investors in learning to identify and regulate their emotional state, more specifically their arousal level. In a first iteration of the development of such a learning platform, the game had to support designs which can be applied to a general audience, before specifically targeting the investor group. In this respect, our development process has been rather successful in its structure. The breakdown of the initial requirements into work items which were also quite blindly implemented was sure to lead to design faults. Being aware of this fact and having specific methods to tackle it was therefore crucial. In the case of the Aiming Game prototype development, we applied the Heuristic Evaluation, executed by colleagues, followed by a Play Testing session involving six arbitrarily chosen subjects in order to identify these design flaws and systematically structure and prioritize them. These methods generated a large quantity of faults, opinions and suggestions on how to improve the prototype.

Interestingly there was not always a congruent philosophy between evaluators and play testers. Certain topics split the groups into two camps expressing completely different opinions and also suggesting very different solutions, while in other cases, expressed issues by one group was rejected as an issue at all by the other. Both the Heuristic evaluation as well as the Play Testing have been quite successful in its purpose to identify design flaws and will therefore be applied in future to similar development scenarios.

During the Play Testing, and also during interviews with said subjects, it became clear that it is important to bear in mind the potential difference between Play Testing subjects and the actual ultimate target group. In the case of the Aiming Game this difference was quite large since our Play Testers mostly consisted of BTH students and colleagues while the target group in fact is investors with very different background and experience. However, this difference does not exclude the possibility of receiving much helpful data from such a study. Regardless of gaming experience, all players should immediately understand the purpose, goal and mechanics of the Aiming Game. When game rules or other elements were unclear to subjects, this was immediately registered as a fault in the game design. In this way, the Aiming Game strives to become an intuitive tool suitable for anyone, regardless of experience.

One of the greater challenges with developing games which should suit all types of players with very different levels of experience is the balance. The Aiming Game should be tuned in such a way that it can be stimulating and meaningful to experienced players as well as people with limited knowledge and experience with computer games. The current design, having time as the only factor for switching between phases, does not fully support this requirement. Instead, in coming development, the game should be progressed through other means, such as performance.

From the Play Testing it was rather obvious that for the Aiming Game to work as a long-term learning platform, it has to become more interesting for players. According to the results of the Play Testing, players found the game to become quite boring even before having played an entire session (4x 3min). Since the idea is that the game should, at least to some extent, entertain players while also learning, the complexity of the game and the variety of game elements must be increased. Both Heuristics and Play Testing suggested several ways to do this.
2.2.1.6 Conclusions and Future Work

The first iteration of the Aiming Game has generated much data, feedback and comments which will be organized into a backlog on which the second iteration will be built on. Play Testing with students and colleagues, Heuristic Evaluation and studies carried out with the help of the Aiming Game, testing its potential as a learning- and training tool, have all led to a more profound understanding of how the development should proceed.

When working with design and collaborating internationally, it is important to have a clear structure which allows all partners to have frequent access to the development process and progression. In the case of the development of this game prototype, this issue was solved by setting follow-up meetings with the product owner, where initial requirements were matched against design choices and implementations. Meetings were held after two weeks of implementation to ensure that the development was heading in the right direction.

The Aiming Game will be used in several xDELIA studies regarding Emotion Regulation, in the summer of 2011. It will also be featured in a study and a paper regarding the training of emotion regulation in relation to performance in games. For future studies the game will be split into three different versions, namely down-regulation, up-regulation and mid-regulation. Before the coming studies, the Emotiv EPOC will be replaced by a wireless ECG device (Movisens EKG Move) and a software framework developed at Forschungszentrum Informatik, which may give more precise readings in terms of arousal which hopefully also will provide improved face validity of the represented bio-feedback.

In the next iteration, there will be several adjustments to the core mechanics of the Aiming Game. Hitting targets, missing and hitting distractors will give direct graphical and numerical feedback at the specific position of the event. The score presenter will be changed to become more visually obvious. There will also be a calibration phase at the beginning of the game session to more accurately account for each individual player.

The development process will continue as before without any major changes. There might be requests for spontaneous Heuristic Evaluation in order to evaluate design changes.

Because if a new version of Unity 3D being available at this time, there will be better support for game elements as graphical user interface and debugging options.

2.2.2 Aiming Game II – Space Investor

2.2.2.1 Prototype Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Space Investor previously: The Aiming Game</td>
</tr>
<tr>
<td>Version</td>
<td>2.0</td>
</tr>
<tr>
<td>Status</td>
<td>Prototypes</td>
</tr>
<tr>
<td>Target User Group</td>
<td>Investors</td>
</tr>
<tr>
<td>Prototype User Group</td>
<td>Internal + students</td>
</tr>
</tbody>
</table>
### Purpose
Train regulation of arousal and determine effects on arousal regulation (as indexed by heart rate)

### Description
The player freighters goods by defending a spaceship from incoming asteroids while travelling through space toward Earth with the purpose of earning resources. Successfully regulating emotions (arousal) makes the game easier to play.

### Training Principle
Becoming aware of one’s emotional state in a stressful situation and adjusting accordingly.

### Transfer
To be studied.

### Context of Use
Part of training intervention package.

### Appearance
Outer space setting.

### Inputs
Mouse + heart rate sensor

### Feedback
Arousal level feedback through arousal indicator, but also through game behaviour such as distortion of the screen and aiming offset. Cumulative score (i.e. resources) logging are available at all times.

### Guidance
Before the player starts to play he or she has to read the instructions of how to play the game. Levels will gradually contain more and more game features so the player will ease into the gameplay. At the beginning of each level there is instructions indicating the new features of that level.

### Simulation
n/a

### Gameplay/Challenges
Keep a given arousal level while performing a shooting task, which difficulty is dependent on the arousal level.

### Emotional reporting
Biofeedback as well as support for in-game questions.

### Human instructor
Experiment instructor

### Social network
None in this version.

### Logging devices
Electrocardiography (ECG) and in-game event handler.

### Development environment
Unity3D 3.3 Pro

### Platform
PC

### Infrastructure
Stand-alone application

### Testing
Functionality and heuristic verification of gameplay, scoring and feedback.

#### 2.2.2.2 Introduction
Space Investor is a game that builds on the concepts from the Aiming Game, which is to perform a shooting task while controlling the arousal level. In contrast to the Aiming Game, Space Investor takes place in a 3D
environment in space. The player is lost in space and wants to find the way back to Earth. In order to do this, the coordinates for Earth has to be found, and the ship needs to be upgraded in order to sustain the journey. The player will travel through asteroid fields to get to different planets in order to gather resources that can be used for upgrades. During the flight between the planets, asteroids have to be shot down to protect the ship and cargo the player is carrying, but also new resources can be collected along the way.

Space Investor is a training platform to train the player’s emotion regulation strategies by punishing the player if an undesired arousal level is present. Both up- and down-regulation is practiced by setting the desired arousal level during various stages in the game.

Since Space Investors is a follow-up from the original Aiming Game, all references in this document to the Space Investor implicitly also refer to the Aiming Game, exceptions are made explicit. This document is meant to be read by itself, but in order to see historical information and discussions about the Aiming Game, the reader is directed to the Aiming Game specification.

2.2.2.2.1 Purpose
The main concern of the xDELIA project is to develop learning interventions for investors, particularly those using the Saxo Bank trading platform. We focus mainly on investors who meet the following criteria:

1. They trade their portfolio sufficiently often that systemic patterns and biases in their trading are detectable.
2. They trade on a regular basis through a trading platform.

Space Investor is a product in an intervention package specifically aimed at assisting investors in becoming aware of their own arousal state as well as training them in regulating their arousal. This is important, as emotions and arousal are strongly linked phenomena. There is evidence to suggest that effective regulation of emotions can have positive effects on performance in investment and trading settings (Fenton-O’Creevy et al., 2010).

Successfully training investors in arousal regulation in a game environment is hypothesized to have a positive effect on their behaviour, through improved arousal regulation, in a real trading environment. Since Space Investor is meant to be a training tool for investors, and at the same time requires sensory technology, it would be beneficial to make it available through day trading centres where investors will have easy access to it.

Pilot studies, Heuristic Evaluations as well as Play Testing on the Aiming Game have pointed out many flaws in the game and its design but have also indicated the potential of the game in its original purpose to train emotion regulation. The second iteration of the Aiming Game development (Space Investor) aims at summarizing the most crucial design flaws and redesign the game prototype accordingly. This document is the result of such an evaluation and presents a solution to correct inadequacies of the first prototype version.

2.2.2.3 Emotions and Emotion Regulation
Being able to express emotions is one of the key attributes to being human yet we often want to suppress or ignore our emotions, perceiving them as being in the way of our goals. While both positive and negative emotions have shown to have an impact on performance and decision-making, the process of affecting negative emotions requires cognitive resources.

Emotions can generally be classified by the independent components arousal and valence (Russell, 1980), where arousal represents excitement level and valence defines whether the arousal is positive or negative. This means that emotions can be visualized in a diagram where arousal and valence defines each axis, as seen in Figure 2.13.
When attempting to measure emotions, one is thus actually measuring a combination of valence and arousal. There are methods for extracting and interpreting valence from e.g. electromyography (EMG) measuring devices (Cacioppo et al., 1986), but since there are several technical difficulties here such as accessibility and extensive setup procedures, the Space Investor will not be concerned with valence for now. Instead the focus rests on arousal as the primary attribute of interest. Since arousal is vital in the definition of emotion, emotion regulation is basically the attempt to change state of arousal and valence.

When facing difficult and stressful tasks, people tend to use one of two main, broadly defined, strategies to deal with the corresponding emotion (Wallace, 2009). These strategies are:

- Suppression
- Reappraisal

Suppressers generally tend to push down emotions, ignoring the fact that they exist and are continuously affecting them. Reappraisers however tend to positively re-evaluate situations.

Both emotion regulation strategies exhaust cognitive resources for the person affected by the emotion (Wallace, 2009). Wallace et al. also points out that suppressing emotions generally takes up more cognitive resources compared to reappraisal when re-evaluating situations. Generally it is therefore preferred to apply reappraisal strategies when encountering unwanted emotions.

In order to identify emotion regulation strategies used by individuals, Gross et al. (2003) developed the Emotion Regulation Questionnaire (ERQ). The ERQ makes specific statements in regards to the emotion regulatory process intended to be measured such as “I control my emotions by changing the way I think about the situation I’m in”. The result of the ERQ can then be cross-correlated with results from a demanding task, such as the Space Investor.

2.2.2.3.1 Research Questions provided by xDELIA
The evidence suggests that learning emotion regulation strategies and improving emotion regulation capabilities can increase performance in cognitively demanding tasks (Wallace, 2009). It is therefore desired to implement this type of training in a serious game and study whether it can have effects in first a game setting, but also if newly acquired emotion regulations skill and knowledge is transferable into a real world setting.

- Can Space Investor help players in learning to identify and experience their arousal state?
- Can Space Investor be used to improve players’ capability to regulate arousal?
- Can Emotion Regulation be trained in order to increase performance in stressful game tasks?
- Are emotion regulation capabilities, trained with Space Investor, transferable to real world settings?

2.2.3.2 Psychophysiological Data

The Aiming Game (v.1) used the Emotiv EPOC in order to measure the arousal level. However, during a study performed by EUR, self-report measurements and the EPOC’s arousal data did not correlate to desired levels. The Space Investor game will instead use ECG-sensors (electrocardiography) developed by Movisens, and arousal-interpreting software developed at FZI. This sensor is predicted to give a better measurement of arousal, and also a more stable wireless signal than the EPOC did. The ECG-sensor will need to establish a baseline for five minutes before interpretation of arousal can be made. Thereafter data from the sensor will give an arousal value in five discrete steps.

2.2.4 Game Description

Throughout this chapter, the design decisions and details of the game are described. However, specific software implementations and description of these are not shown in this chapter.

2.2.4.1 Purpose

The game’s purpose is to be a tool embedded in xDelia’s learning intervention where participants train emotion regulation across a longer period of time (e.g., weeks). Therefore, the final version of the Space Investor can be played for several hours.

2.2.4.2 Goal of the Game

The player’s goal in the game is to return to Earth, but the player has no idea of where that is and do not have a good enough ship to complete the travel. In order to find information about where Earth is and to upgrade the ship, resources are collected throughout the game to enhance the chances of getting home. This is done through shooting down asteroids and avoiding having an undesired arousal state (thus the player is not actually navigating the ship). There is also the incentive to become a better player by showing the players statistics of their performance.

2.2.4.3 Story & Theme

The story takes place in outer space where the player steers a freighter spaceship between planets. The purpose is to transport valuable goods between space kingdoms in order to receive payment and other rewards. The player takes the role of a ship captain using a biometric interface to navigate the ship. The steering mechanics, shields, scanning devices (i.e., the vision) of the ship is affected directly by the navigator’s emotional state. Should the navigator divert from the target aroused, the mechanics of the ship will become impaired.

2.2.4.4 Gameplay

The aim of the game is to navigate a freighter spaceship from one planet to another in space, collecting resources and destroying obstacles on the way. The freighter ship is carrying goods, which are to be delivered on the destination planet, together with the resources the player managed to pick up on the way there.
The game is structured in different levels, where each level consists of a travel route between two planets. In between levels, the player will receive resources, as well as upgrade the spaceship (See section Upgrades). The game is presented in 3D and the camera is set to the front of a freighter spaceship, which means that the player will not actually see the ship (First person view, see Figure 2.14).

![Figure 2.14 – Print screen from the Space Investor game](image)

The spaceship is constantly moving forward through space and is frequently approached by asteroids that must be shot down in order for the ship not to get hit. At its disposal the spaceship has a set of guns, which can be used to shoot down asteroids and other obstacles. Occasionally when traveling through space, the player will encounter resources. Some resources will be collected automatically by passing through them, but should they accidentally get shot they will be destroyed, others are needed to first destroy the surrounding shell in order to be collected.

### 2.2.2.4.5 Tutorial

There is a dedicated tutorial in the game, presenting all the different elements that will occur, letting the player test out the different game elements while being briefed about them. This tutorial should be played first, but can of course be skipped by an experienced player. In this tutorial the player will learn how to play the game gradually, and also what effects the arousal levels will have.
2.2.2.4.6 Missions
During three levels the player has the mission to arrive at the destination with a certain level of resources (i.e. successfully avoided to be hit by too many asteroids) in order to upgrade the ship to be able to proceed to the next level. Should the player not reach the desired goal so the new ship parts cannot be bought, the level has to be played again. The player keeps the resources from the previous attempt, so it will be easier to complete the consecutive attempts. There will also be achievements that the player can complete, but these are level-independent and are up to the player if they are to be strived for.

2.2.2.4.7 Game Mechanics
The game mechanics are the rule system of a game, which are to support the entertainment and, in the case with learning games, primarily the learning.

2.2.2.4.7.1 Game Logic
Space Investor is a one-player game in which the player cannot die. Instead, resources are lost every time the ship gets hit by an asteroid, which happens with each asteroid that is not shot down. But there is also have the chance of getting resources by collecting resource asteroids by not firing at them and let them hit you.

The arousal value will be set for each level with an desired value, any deviations from this value will render different sanctions for the player, among other increase of speed of the spaceship, blurry vision, and aiming offset (see further description under Difficulty level).

2.2.2.4.7.2 Difficulty level
When developing a serious game, such as the Space Investor game, it must be taken into consideration the fact that the target group may not be experienced game players or have the incentive to go through a steep learning curve. Therefore the game must be playable by all types of people, ranging from hard core gamers to completely inexperienced players. In the case of the Space Investor game there is a delicate line where players must feel pressured and perceive the game as hard no matter how experienced they may be, at the same time as not perceiving the game to be too difficult for them to even try.

To summarize, there are two dimensions of variety of difficulty in Space Investors:
- Game elements not affected by arousal
- Game elements affected by arousal

In Space Investors, different game elements are affected by the player’s arousal level, which will make the game harder. In table 1 below, the five different game elements affected by arousal level and the three game elements that are not affected by arousal are presented.
### Arousal effects

In order to train emotion regulation during the game, it is important that the game is sufficiently challenging in the aspect of emotional control, in order to elicit an emotional response. These items described below are the different ways arousal influence gameplay, and are meant to make the game harder in different aspects. The further away from a wanted level of arousal, the bigger each of the effects will be. In game these effects are explained with the bio-physiological interface towards the ship and when the physical state of the player is undesired, the ship’s functions are working at a suboptimal level.

#### 2.2.2.4.7.2.1.1 Speed

The ship’s speed is increasing if an unwanted arousal level is detected. This makes the game harder since asteroids will hit the ship faster and also spawn with higher frequency (in time). The normal speed of the ship is 100 space units per second, with up to another 100 space units depending on arousal level. This means that the speed will double if the player has an arousal level on a maximum distance from the wanted value.

#### 2.2.2.4.7.2.1.2 Aiming offset

As in The Aiming Game, a randomized offset from the aiming point will be present when unwanted levels of arousal occur. This is meant to make the game harder since the targets will be harder to hit in general. With the help of a double cosine function a fractal movement pattern is created. The arousal value determines the amount the fractal function is allowed to affect the aiming.

#### 2.2.2.4.7.2.1.3 Blur

The blur effect is also a remaining feature from The Aiming Game where it was implemented in order to make it hard to hit the targets. In Space Investor this is also the case, but it is primarily a means to make it hard for the player to distinguish between resources and asteroids, rendering the player to shoot down things that would have been better to save.

### Table 1 Different difficulty alterations of game elements

<table>
<thead>
<tr>
<th>Arousal effects</th>
<th>Non-arousal effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aiming</td>
<td>Asteroid spawn rates</td>
</tr>
<tr>
<td>Blurry vision</td>
<td>Various types of asteroids</td>
</tr>
<tr>
<td>Spaceship speed</td>
<td></td>
</tr>
<tr>
<td>Weakened shields</td>
<td></td>
</tr>
</tbody>
</table>

© xDELLA Consortium – All Rights Reserved
2.2.2.4.7.2.1.4 Shields
The spaceship has shields in order to protect it from asteroids, however, these are weakened as a result of arousal levels. This means that more resources will be lost when hit in this state, as well as the camera will shake for a longer period of time. Resources lost are doubled when the player is on the wrong side of the arousal spectrum, compared to the right side. It is a linear increase.

2.2.2.4.7.2.2 Non-arousal effects
These effects will increase when progressing to higher levels. Asteroid spawn rate will be higher, making it harder to shoot them down in time. Various types will appear, making it harder to distinguish the one’s that will give resources and the one’s that need to be shot down.

2.2.2.4.7.2.3 Input
When playing, the ship is automatically moving forward in the space environment without the need for player input. Just as its predecessor, the Aiming Game, the main goal is to aim and shoot at different object. This is done with the mouse buttons, left and right for different weapons. The indirect (meaning: not under direct conscious control) input from the player is the heart rate that will distort the game and make it harder.

2.2.2.4.7.3 Elements
Throughout the game there are two main categories of objects that the player encounter, resources and asteroids. These vary (randomly set) in size (and thus damage/resource gain). Various levels exists and are characterized by difference the spawn rate of asteroids and resources, the skybox (i.e. background), and music.

2.2.2.4.7.3.1 Weapons
To enhance the possibility of immersion and the drive for enhancing oneself in the game, various weapons exist. This is also so players’ can chose their own strategy when playing. During gameplay, the player is equipped with two weapons (one for left mouse button, and one for the right one) at all times. This makes it a strategic choice about which guns the player should choose and use. Each weapon has the attributes damage and cooldown time, which is the amount of time before the weapon can be fired again.

2.2.2.4.7.3.1.1 Main Cannon
The main cannon is the default weapon that all players receive from the very beginning of the Space Investor game. It is designed to have a rather standardized functionality, which should be recognizable by most semi-experienced players.
Weapon damage: 20
Cool down time: 0.2s

2.2.2.4.7.3.1.2 Bomber
The Bomber is a slow but powerful weapon that acts as an auxiliary weapon. It can destroy any obstacle in one hit but has a long cool down period before it can be used again making it rather powerless again multiple targets.
Weapon damage: 300
Cool down time: 3s

2.2.2.4.7.3.1.3 Shotgunner
The Shotgunner is powerful weapon again large or multiple targets. It simultaneously fires several Main Cannon bullets in a small spread forward, allowing it to hit several targets at once.
Weapon damage: 100
Cool down time: 5s

2.2.2.4.7.3.1.4 Laser
The laser is the ultimate weapon in this game, fast with great damage.
Weapon damage: 200/s
Cool down time: None

2.2.2.4.7.4 Upgrades

While completing missions, the player will eventually have enough resources to buy upgrades. Upgrades are divided into two categories:

- Ship upgrades
- Weapon upgrades

Ship upgrades are upgrades that need to be purchased in order to continue to certain levels, and will not have any effect on the gameplay. Weapon upgrades are concrete weapons; there are the four weapons that can be unlocked throughout the game.
Credit system
When playing the Space Investor game, in addition to destroying asteroids, the player should also avoid destroying various resources which sporadically appear on the screen. By avoiding shooting these resources down, the player is able to collect them by flying into them. The resources are then automatically added the ship’s resource load.

The resources used in the game are:

- Emerald - green
- Ruby - red
- Sapphire - blue
- Topaz – yellow

Resources are used to purchase weapons and upgrades at the end of each level, when entering the Upgrade screen. All weapons and upgrades require a specific amount of at least two different resources, which is why it becomes imperative for the player to accumulate an even amount of all available resources in order to purchase the most important upgrades that the game has to offer.

2.2.2.5 Software
Here specific software solutions as well as what are being used in order to implement the Space Investor game are discussed.

2.2.2.5.1 Bio-feedback & Logging
The player is connected to the game via a customized heart-rate sensor, developed at Movisens(hardware) and FZI(software), that transmits information to the game regarding the player’s arousal state. The arousal logging is done by the xAffect software, which connects directly to the sensor. Space Investor is writing to the same file for convenience, but also to a separate text file for backup. Data is logged continuously while the game is being played, but at each level a summary of that specific level is saved as well. This summary contains information of numbers of asteroids and resources spawned; hits and misses for each weapon; start and end time for the level; resources collected.

There is support for questions to be answered between each level, put a file named “Questions.txt” in the same folder as the .exe-file and write one question per line in that file and they will appear after each level (excluding the tutorial). These answers will be included in the log-files containing the statistics from the game.

2.2.2.5.2 Implementation
The programming environment used is Unity 3D Pro 3.3, a game development platform (unity3d.com). The game is constantly saved on the xDELIA code SVN server (https://svn.fzi.de/svn/ipesvn/code/trunk/xdelia/SpaceInvestor), thus version controlled, as well as local copies, thus protected from computer malfunctions. The game project follows an incremental development style, always having a product that can be demonstrated at all times.

2.2.2.5.3 Graphics
There are five different screens that the player will visit during the game. These are: the Sensor Connection screen, the Main Menu, the Baseline Reading, the Game screen, and the Upgrade screen (see Figure 2.15).
The Main menu is a simple menu with a space background where the player can make appropriate choices like running the tutorial, playing the game, continue the last saved game. Here, there are some restrictions in choice depending on whether or not the current game version is a study version or a training version.

The Level is where the player has to shoot down asteroids and consists of a view of the space, a HUD (heads-up display) containing information about cooldown periods, arousal value, resources left on ship, and distance travelled.

There will also be on-screen messages from the ship-AI, which encourages the player to avoid shooting at resources and to destroy asteroids.

The game is designed to run at resolution 1280x1024, if that is not possible, a higher resolution should be chosen. Failing to do so might distort graphical elements of the game.

2.2.2.5.3.1 Feedback
From the evaluations of The Aiming Game, several limitations with the feedback was displayed. The players’ only glanced at the arousal level during gameplay, in this version there is a horizontal bar placed down at the centre of the screen, which also is color coded, hopefully enhancing the players’ awareness of their arousal state. Another problem was that people did not really understand their current points and when they got them, this has been enhanced in Space Investor by moving the points to the HUD and have on-screen feedback when resources are gained or lost. These messages are also color coded.

2.2.2.5.4 Sound
Typical game sounds for explosions, gunfire, and such exists and are retrieved from freesounds.org. Music in the game is not yet a completed design decision but support exists to have a music file for each different level.

2.2.2.5.5 Versioning
The Space Investor game is made as a generic platform for game studies. More specifically, in the case of xDELIA and emotion regulation studies and training, there are two different directions to take while playing the game. These are “Sensor connected” and “Sensor disconnected”, which simply means that the game is playable with and without a Movisens HR sensor. The main advantages of this approach are:

- Easily accessible platform for control group studies
- Almost complete game access, even without a sensor
- Access to a test environment where game difficulty can be evaluated separately

The “no-arousal” version is accessed by entering sensor number “00000” instead of a real sensor number or MAC address into the sensor number field. The game engine then disregards the connection to the external sensor and instead sets up the game environment with a set of predetermined variables to substitute the absence of a sensor device.

2.1 The sensor connection screen
2.2.2.6 Evaluation

Space Investor was tested out in three ways before the actual delivery (31st May) to EUR: functionality testing, heuristics and play testing. For details about evaluation methods used, see D17-4.1.

2.2.2.6.1 Functionality Evaluation

The functionality evaluation consists of three categories: validation, verification, and future support. The validation was done together with the product owner (EUR) at a face-to-face meeting where the product was demonstrated. By letting the product owner test it before it was finalized much of the rather abstract discussions become very concrete and much feedback could be given, also misconceptions due to possible miscommunication was solved at this point.

The verification was done by the development team throughout the development process. By working in an incremental development style the product was constantly tested for errors when new things are added, thus minimizing latent errors.

Measurement of arousal was done by a separate component, thus allowing future support for other sensors to be used within the same game. Many variables were set in the Unity 3D environment, making them easy to adjust for future studies.

After delivery, there were unforeseen sensor related issues, thus allowing more time to develop the game further. The project owner then decided about new implementations. These implementations were communicated, implemented and then validated by the product owner in short cycles.

2.2.2.6.2 Gameplay and Usability Evaluation

The play testing evaluation of the Space Investor game was done with five participants who all played the version used in the longitudinal study performed by xDELIA-partners at Erasmus University. All participants used the Movisens EKG sensor with a 300 second baseline. Five levels, excluding tutorial and intermissions, with a length of up to 180 seconds were played. All players filled in the game experience questionnaire (GEQ) and system usability scale (SUS). For detailed results see Appendix I (results in Swedish).

The results showed a spread in difficulty (GEQ 13), enjoyment (GEQ 1, 41), immersion (GEQ 5, 10, 15, and 40), stress (GEQ 7), perceived skill (GEQ 17), boredom (GEQ 18), irritation (GEQ 24), challenge (GEQ 29), and time pressure (GEQ 37). These results suggest that the Space Investor game might not be a game for everybody. While several participants showed positive results in the above mentioned categories, others did not, suggesting that another game might be more suitable for them.

Most participants did not appreciate the story of the game. According to the play testing, very few players felt that it contributed to the game experience at all (GEQ 3, 35). While most of the play testers did not feel completely stimulated (GEQ 26) by the game, nor felt like they learned a lot (GEQ 8), the majority felt happy during the session (GEQ 6, 38). It is likely that these results are due to play testers’ expectations and gaming experience and might not transfer to the investor setting. A positive note is that even the experienced gamers in this play testing sample found the game to be rather aesthetically appealing (GEQ 14). Participants experienced the game as limited in terms of exploration possibilities (GEQ 21). The game is, however, intentionally very linear in order to control content. More work in making the game being perceived as less linear could be done, if deemed necessary.

One of the most important aspects from the Aiming Game was the intuitive interface and gameplay mechanics. According to the Space Investor play testing, players congruently felt that this aspect had been sustained, if not improved (SUS 1-10). All items of the SUS questionnaire consistently gave good usability results pointing towards usability being preserved between the Aiming Game and the Space Investor game.
2.2.2.7 Discussion

Previous iterations lacked a clear goal if the game, this is now implemented. Also it did not focus to the same extent of arousal being as central as it is now. Many ideas of how to make the game harder by arousal have been scratched, stripping it down to the most appropriate ones.

Since Space Investor is module based, the sensor can be exchanged fairly easy. Difficulty can be adjusted by the use of Unity3D by shortening level lengths, speed of the spaceship, spawning rate of asteroids, etc. Also, adding new levels, theme music for each level, environment pictures, arbitrary objects that can collide with the spaceship, and up- or mid-regulation of emotions. Thus Space Investor is highly configurable, making it good in experimental use.

- EUR input on what the game means for emotion regulation
2.3 Auction game

2.3.1 Introduction

The game prototype described in this section is an Auction game; a serious game simulating stock exchange. Player is set in the position of an auctioneer where he/she buys or sells goods/stocks. Every round stock has a different price and player’s goal in the game is to buy and sell stocks to the right true price of that round. Helping him in his decision are his trusted consultants; thus every round player receives three stock price estimations from them. The true price of a stock can be calculated by the mean of those three estimations. Dependent of the emotional arousal of a player, given estimations are spread out with smaller/larger variance making it easier/harder to pick out the correct true price of a round; thus making it easier/harder to buy and sell to the right price. The variance of the estimations will get larger the higher the emotional arousal. To make it harder for a player, engaging his cognitive abilities to the limit, next true price is further away from the previous one as the emotional arousal gets higher.

The objective of the Auction game is to train investors in emotion regulation; by showing the level of arousal the player can gain an awareness of his her emotional state and the influence of emotion regulation on decision making. At the same time the arousal level will influence the game play, specifically price estimations. To be able to play successfully the player has to regulate his or her arousal.

The main target group is traders at Saxo Bank. During the development phase, the game was tested by project members, colleagues and/or students to verify the functionality, usability and gameplay. The potential impact of the game on the players’ emotion regulation capabilities was dealt with within the evaluation phase and it will be further explored in the future studies.

2.3.1.1 Purpose

The field of modern financial economics assumes that people behave with extreme rationality, but they do not. Traditional finance theory is usually based on efficient market hypothesis, assuming that investors are rational and the prices of stocks fully reflect available information while price changes should be random. Many empirical researches did not support this way of thinking (i.e. (Shiller, Fischer, & Friedman, 1984), (De Bondt & Thaler, 1985)). A study of verbal expressions and emotions performed by Shimanoff (1984) found that regret was the most frequently named negative emotion. Similar statement has been confirmed by Lankman (1993, p.110) where regret was found to be common, if not universal, experience. Emotions have a powerful impact on our lives: They shape our behavior, and their influence is so pervasive that no decision theory could be complete without taking their role into account. When markets are as volatile as they currently are, then there is no doubt that emotions play even more of a role. This has been confirmed in a research study where it was found that investors’ psychology tend to feel sorrow and grief after having made an error in judgment (Yahyazadehfar, Ghayekhloo, & Sadeghi, 1985). They found that the investors deciding whether to sell the stock are typically emotionally affected by whether the stock was bought for more of less than the current price. Therefore, the investor must recognize this fact and try to practice some mechanism to control his (her) irrational behavior.

In economical investing emotional factors play an important role. A study in the domain of losses has been conducted measuring investors experienced regret and (dis)satisfaction during incurring various sizes of gains or losses over multiple periods, and link these emotions to their subsequent hold/sell decisions (Lee, Kraeussl, & Paas, 2009). It has been found that in the loss domain, only regret has significant impact on investors’ subsequent hold/sell decisions. Furthermore, they report that high experienced regret and pride are linked to a larger probability to sell; as for anticipated pride leads to a larger probability to sell, while anticipated regret links to a smaller probability to sell. Their findings demonstrate that investors’ subsequent hold/sell decisions are simultaneously affected by their experienced and anticipated emotions. In a similar study it has been found that alternative outcomes may affect evaluation of decisions (Fogel & Berry, 2006). Anticipation of regret may lead investors into the trap of holding losing stocks too long. All of respondents in their study reported regret for investment decisions, either for not selling a losing stock soon enough, or for selling a winning stock too soon.
The psychology of emotions, however, offers a further set of influences to consider, and Elster (1998) advocates that economic theory should take account of their impact too. He concludes “we need a better understanding of how emotions actually influence behavior” and how they “affect the ability to make rational choices”. Elster's (1998) call has been heard and there is a move in the economics literature to recognize those emotions (i.e. (Loewenstein, 2000) and feeling (Romer, 2000) have important roles to play in understanding economic behavior. As Summers & Duxbury (2007) found that emotional response is clearly important in explaining behavior in disposition effect and that manipulating emotions changes behavior. This provides evidence of the importance of emotion in economic behavior.

Regret is a prominent example for the impact of emotions in financial decision making (Engelbrecht-Wiggans & Katok, 2008). Moreover regret is believed to have a major impact for several biases in financial decision making e.g. the disposition effect. We measure skin conductance and heart rate; with these measures we seek to improve the understanding of the emotional processing in economic decision making. Additionally, questionnaires examine qualitative differences in emotion and emotion regulation between the subjects. These insights in turn can help understanding traders’ behavior in the field, where regret also is believed to play a major role.

The laboratory experiment was conducted in spring and summer 2010 at the Institute of Information Systems and Management (IISM) at the Karlsruhe Institute of Technology (KIT). Each session lasted approximately 1.5 hours. All together 72 students participated, most of them undergraduates with background in economics. In its structure the experiment was based on an auction experiment with varying regret information of (Engelbrecht-Wiggans & Katok, 2008). Each human subject played 50 first-price sealed-bid auctions. Each subject bids against 2 computerized subjects to exclude social preferences or other effects of interpersonal activities. The computerized agents received equally distributed valuations between 0 and 100 and bid always 2/3 of this valuation which corresponds to the Nash equilibrium (NE). The incentive to win for each participant was cash and the value for each auctioned good was an induced private value with values in random order of 50, 60, 70, 80 and 90. Each subject received 10 times the same value consecutively to control for the effect of learning. Three differing treatments with 24 subjects each were arranged. The separate regret treatments varied in the form of regret information the bidders received after they had placed their bids, called the ‘winner regret (WR)’, ‘loser regret (LR)’ and ‘no regret (NR)’ treatment. In the WR treatment bidders received the information ‘money left on the table (MLOTT)’, i.e. own bid minus highest opponent bid, in case they won an auction. In the LR treatment bidders received their ‘missed opportunity (MO)’, i.e. own valuation minus highest bid. In the NR treatment bidders didn’t receive any regret information. This gave the opportunity to explicitly test for the influence of loser regret and winner regret compared to the treatment with no regret feedback information. After the experiment bidders received a questionnaire that accounted for risk aversion, two types of emotion regulation, and perceived emotions. Additionally, for all participants the heart rate (HR), skin conductance response (SCR) was measured via psycho-physiological appliances during the complete auction task. Economic results were consistent with the results of (Engelbrecht-Wiggans & Katok, 2008): Subjects in the ‘loser regret (LR)’ treatment placed significantly higher bids than in the ‘no regret (NR)’ treatment and bidders in the ‘winner regret (WR)’ treatment placed lower bids than in the ‘no regret (NR)’ treatment. In accordance to other auction experiments participants placed significantly higher bids than the risk neutral Nash equilibrium (NE). Summarizing: LR > NR > WR > NE.

Subjects’ SCR to the experience of winning an auction was significantly higher compared to the experience of losing an auction for all treatments. This can be interpreted in a way that the “happiness” of winning an auction is higher than the “negative emotion” of losing an auction.

Subjects’ SCR to the feedback information of LR was higher compared to the feedback information to WR. That means loser regret might be higher than winner regret.

More strikingly, subjects’ SCR was not related to the height of the “missed opportunity”. This leads to the idea that, opposed to the assumption of (Engelbrecht-Wiggans & Katok, 2008), the experienced regret is not depended on the height of the “missed opportunity”.

Most strikingly, subjects’ SCR was highly correlated to the amount of “money left on the table”, but opposed than expected. That means, a small value of “money left on the table” (i.e. values of 0 or 1) results the highest SCR. Opposed to that, SCR is smaller for higher values of “money left on the table”. This high
response can be interpreted as “relief” or even “thrill of winning” instead of winner regret, since the bidder won by a very small amount. This has not been considered before in this case. Significant drop in heart rate variability, when participants lose an auction compared to the drop when bidders win an auction. This high drop in heart rate as response to negative affect is coherent with former considerations of Bradley and Lang (2008).

As these studies explore and suggest, emotional factors have an impact and important role in economical investing. They shape our behavior and decisions. With our game prototype we set out to better understand how emotions actually influence behavior, specifically investors’ buy/sell decision. The Auction Game’s aim is to train emotion regulation, but also to get the player aware of the arising emotions, as Yahyazadehfar, Ghayekhloo, & Sadeghi (1985) suggest. In other words, guide the player towards mindfulness of emotions. By displaying player’s emotional state as an indicator one of five arousal levels (very cool, cool, average, exited and panic), player has to keep calm to minimize the deviation of the prices and thereby have a better chance to accomplish higher profit.

### 2.3.1.2 Emotions and emotion regulation

Expressing emotions is one of the key attributes to being human, but we often suppress or ignore our emotions as we perceive them being in the way of our goals. It has been shown that both positive and negative emotions have an impact on performance and decision making; moreover, the process of affecting negative emotions requires cognitive resources. Russell (1980) generally classified emotions by their independent components arousal and valence, where arousal represents excitement level and valence defines whether the arousal is positive or negative. Figure 2.17 seen below visualizes this separation by placing arousal and valence on separate axis.

![Emotions in the valence-arousal space](image)

Thus when measuring emotions one is actually measuring a combination of valence and arousal. There are methods for extracting and interpreting valence from e.g. electromyography (EMG) measuring devices (Cacioppo et al., 1986), but for the Auction Game there are several technical difficulties implementing it, such as accessibility and extensive setup procedures, thus we will not be concerned with valence for now.
Arousal remains as the primary attribute of interest. Emotion regulation is generally the attempt to change state of arousal and valence, since arousal is vital in the definition of emotion. People tend to use one of two main, broadly defined, strategies to deal with corresponding emotions emerging when facing difficult and stressful tasks (Wallace, 2009). These strategies are:

- Suppression
- Reappraisal

Suppressers generally tend to constantly push down emotions, ignoring the fact that they exist and are continuously affecting them and using their cognitive ability. On the other hand, reappraisers tend to positively re-evaluate situations.

For the person affected by the emotion, both emotion regulation strategies mentioned take up cognitive resources (Wallace, 2009). (Wallace, 2009) also states that suppressing emotions generally takes up more cognitive resources in comparison to the reappraisal strategy when encountering unwanted emotions.

Emotion Regulation Questionnaire (ERQ) was developed by Gross et al. (2003) in order to identify emotion regulation strategies used by individuals. It makes specific statements in regards to the emotion regulatory process intended to be measured, such as “I control my emotions by changing the way I think about the situation I’m in.” Results from a demanding task presented in the Auction Game can be cross-correlated with the results of the ERQ questionnaire.

The objective of the Auction game is to train investors in emotion regulation; by showing the level of arousal the player can gain an awareness of his/her emotional state and its influence of emotion regulation on decision making. This is done by:

- Improvement of interoception and self-monitoring of physiological arousal and hence emotional state with indication of personal emotional state.
- Improvement of skills in emotion regulation by elements that reward good emotion regulation and punish poor emotion regulation strategies.

So the core motivation for the Auction Game is that there is a strong link between maladaptive financial behavior and poor emotion regulation. Therefore the Auction Game can be considered as an emotion regulation training game in the context of financial decisions.

2.3.1.3 Psychophysiological data

The Auction game prototype aims to train emotion regulation, but also to get the player aware of the arising emotions. It makes sense to gather data on player’s arousal level during the game and the outcome of his/her choices. Heart Rate Variance (HRV) calculated from player’s electrocardiogram (ECG) signal provides a good measure of his/her emotional arousal. Movisens wireless device is used to acquire the ECG signal; this device provides us with an arousal level value through an API with, calculated internally within the xAffect software.

2.3.1.4 Research questions provided by xDelia

The prototype was proposed by the FZI and they have a special goal with the game. From a game perspective, for BTH within xDelia, it is interesting to know:

1. What elements in the games does the player focus on?
2. How is the arousal indicator used by the player?
3. What strategies do the players use to be able to play successfully?

Wallace (2009) states that learning emotion regulation strategies and improving emotion regulation capabilities can increase performance in cognitively demanding tasks. It is therefore desired to implement this type of training in a serious game and study whether it can have effects in first a game setting, but also if newly acquired emotion regulations skill and knowledge is transferable into a real world setting.
The stated research questions are agreed on inside xDelia:

1. Do players become better emotion regulators in life and in finance by playing more often?
2. Do players playing this game get more aware of their emotions while making financial decisions?
2.3.2 Game description

The objective of the Auction Game is to train the players in emotion regulation. Better emotion regulation leads to a more predictive game play; thereby, it is easier to make higher profit in the game. This is supported with estimations given to the player from his/hers advisors. Player is presented with three price estimations before he/she has to reach a buy or sell decision. Price estimations are directly linked to player’s emotional arousal level. Consultants’ price estimations are generated by a computer, but they deviate from the true price with higher variance the more aroused the player is. In the Auction game the task is to calculate a mean value of a stock, from the given estimations, to be able to sell or buy the stocks at the correct best price; thus a lower emotional arousal will make the stock price estimations closer to the true price where buy or sell decision can easily be made. Moreover, goal is to reach the highest level possible. Hence, the ending time of the single levels and the game can vary, depending on player’s skills. Player’s skills to earn money in the game are strongly related to player’s skills to regulate his/hers overall emotional arousal state. Good emotion regulation will help the player being successful in the game. From level to level the tasks to regulate your emotion will become more and more difficult as piecewise new elements are included. Among those are additional information, time, auditory, and visual constraints (will be described later in detail). A certain randomness of these elements also strengthens the immersion and excitement into the game.

This game prototype aims to train emotion regulation, but also to get the player aware of the arising emotions. In other words, guide the player towards mindfulness of emotions. By displaying player’s emotional state as an indicator one of five levels (very cool, cool, average, exited and panic), player has to keep calm to minimize the deviation of the estimations and thereby have a better chance to accomplish a higher profit. Indicator levels are dependent of player’s level of arousal.

To advance from one level to the next, the player has to gain a certain amount of money to reach that level’s profit goal. As soon as he hits this amount, a button pops up and he can proceed to the next level. While in the first levels the constraint for physiological arousal is to simply down-regulate your emotion, at later levels player had to aim at optional level of arousal.

Player has a limited amount of time during which he/she has to make enough profit to reach the next level or the game is finished. As the game gets harder, eventually player gets kicked out of the game and is presented with the level he/she has reached. An additional high-score of money earned gives the player an opportunity to compare his result to his previous ones.

Throughout this chapter, the design decisions and details or the game are described.

2.3.2.1 Purpose

The game’s purpose is to be a tool embedded in xDelia’s final learning intervention where participants train emotion regulation across a longer period of time (e.g., weeks), inevitably they also get aware of the arising emotions. Therefore, the final version of the Auction Game can be played for several hours on multiple occasions.

2.3.2.2 Goal of the game

The player’s goal in the Auction Game is to earn as much money as possible and thus reach highest level possible. In order to gain money player has to take the right buy/sell decisions. By calculating the mean of the three prices in the clouds, player knows the true price of the good/stock. After the indication of the three prices player has to reach buy or sell decision for the offered price.

To reach the next level player has to earn enough money before the round time ends. As the game progresses, as well as your arousal gets higher, game difficulty will change making it harder to reach profitable decision.
2.3.2.3 Story and theme

The story in the Auction Game sets the player as an auctioneer where he or she buys or sells goods/stocks. As the player makes profitable decisions, he/she earns money and progresses through the game levels. At his/hers disposal are three trusted consultants, thus every round player receives three stock price estimations from them. Price estimations are directly simulating effects of player’s arousal so that the more the player is aroused, the more price estimations are spread away from each other making it harder to calculate the true price of the stock; moreover, next true trading price in the next round is farther away from the previous one.

The theme of the game is abstract one depicting sky and clouds. This supports the goal of the game to relax the player and motivate mindfulness.

2.3.2.4 Gameplay

The goal of the Auction Game is to reach the highest level possible and thus mark your score on the highscore list. To reach the next level player has to make a certain amount of profit. Every trading decision brings profit or loss for the player. In every round player receives three price estimations from his trusted consultants. From those estimations player has to calculate the mean true price with which he trades this round, as can be seen on the Figure 2.18.

As can be seen on the Figure 2.19 the Auction Game is presented in 2D environment where clouds represent price estimations. Player can see his arousal level on the meter in the right top corner, as well as by the color of the clouds (green, yellow and red). The profit goal and total money earned are presented on the meter on the right side of the screen.
Game levels get tougher as the player progresses. This has been achieved by random distracting effect affecting the clouds representing the price estimations. The color of the sky background depicts progress through the game as it turns dimmer and grey. Game is conceptualized such that it has no predetermined ending, but after 10th level, game becomes quite hard entering what is called “death mode”.

2.3.2.5 Tutorial
The Auction Game features full tutorial guiding the player through every aspect of gameplay. Player is slowly guided through the price estimations and decision process. Tutorial should be played the first time player gets in contact with the game, but it can be skipped if the player has knowledge about the Auction Game.

2.3.2.6 Game mechanics
The game mechanics are the rule system of a game, which is there to support the fun and run the game.

2.3.3 Game logic

The game consists of the introductory screen where the player is presented with the instructions on how to play the game. Here is also possible to choose to take the baseline reading, or to display or not certain elements in the game; i.e. emotion arousal indicator.

The game logic consists of estimation clouds which are individually presented to a player for one second, this way the player has to concentrate on the game. To make the player even more attentive clouds appear at random places on the screen. Arousal level indicator has been placed in the top right part of the screen;
this position was chosen so that the player can easily see his or her own emotional state indicated but still not get cluttered screen. The more aroused the player is the price estimation presented in the clouds deviate more and more from the mean true price of the stock. If the player is not aroused at all it is easy to pick out the correct price, on the other hand, the more aroused the player is the harder it gets to calculate the true price; therefore the player benefits from awareness and control of his or her emotional state. After all of the clouds are presented, player has two seconds to make a buy, sell or no trade decision. To make a decision player has to click on a buy, sell or no trade button. Dependent of his decision he/she is presented with the audio and video feedback on the outcome of his decision, profit or loss. The total gain and loss values are updated respectively and decide if the player reached the new level. After which a new round begins where player is again presented with clouds estimations and so on. After a certain profit gain player reaches a new level. The diagram of game rounds is presented on the next Figure 2.20.

Figure 2.20 – Diagram of game rounds

2.3.4 Game difficulty

When developing a serious game, such as the Auction Game, it must be taken into consideration the fact that the target group may not be experienced game players or have the incentive to go through a steep learning curve. Therefore the game must be playable by all types of people, ranging from hard core gamers to completely inexperienced players. In the case of the Auction Game there is a delicate line where players must feel pressured and perceive the game as hard no matter how experienced they may be, at the same time as not perceiving the game to be too difficult for them to even try. Thus the game starts slowly introducing distracting elements throughout the levels with a crescendo of extremely hard gameplay.

To summarize, there are two dimensions of variety of difficulty in Auction Game:

- Game elements not affected by arousal
- Game elements affected by arousal

In the Auction Game, different game elements are affected by the player’s arousal level, which will make the game harder.

2.3.4.1 Arousal effects

In order to train emotion regulation during the game, it is important that the game is sufficiently challenging in the aspect of emotional arousal control. These item described below are the different ways arousal influence gameplay, and are meant to make the game harder in different aspects. The further away from a wanted level of arousal, the bigger each of the effects will be.

- **Distribution of price signals** – In each round player is presented with three price estimations from trusted consultants. Dependent on the arousal level of the player detected by the consultants, they
will spread their estimations with larger variance. This will make harder to calculate the mean true price. The more players away from the desired arousal, the larger the effect i.e. more spread out price estimations.

- **Distribution of true price in the next period** – Every round stock shifts its true price on the market. Dependent on the player’s arousal deviation of the next true price is larger confusing the player. The further away from a wanted level of arousal, the bigger each of the effects will be i.e. true price will shift more unexpectedly.

- **Speed of cloud appearance increases** – As the game progresses clouds start moving and their speed is directly linked to the player arousal. The further away from a wanted level of arousal, the bigger each of the effects will be i.e. clouds will start moving faster which makes it harder to see the price estimation.

2.3.4.2 **Non arousal effects**

These effects will increase when progressing to higher levels.

- **Background sky** – Picture of the sky in the background is reflecting level the player has reached. On subsequent levels sky turns dimmer and grey.

- **Speed of cloud appearance increases. Appearance/Decision time decreases** – As the game progresses clouds will start moving and their speed will increase as the player progresses through the levels.

While the player is progressing through a level towards its goal, some elements get in the way to distract him:

- **Music becomes faster** – As the player reaches half of the level goal music slightly increases its tempo; moreover one quarter away from the goal music noticeably speeds up to distract the player in making quick decisions in the attempt to make him commit error.

- **Speed of cloud appearance increases. Appearance/Decision time decreases** - As the player progresses through the level cloud estimations slightly increase their movement speed; moreover appearance and decision time slightly decreases to distract the player in making quick decisions in the attempt to make him commit error.

All levels have in common that the cloud estimations vary in some random pattern. This helps keeping the game interesting.

- **Clouds can move simulating the wind**

- **Change their positions**

- **Can become bigger and smaller in sizes (so single numbers seem more important, in fact they aren’t)**

- **Speed and appearance time of the clouds can become random**

- **Fake clouds start to appear**
Sometime pictures from IAPS should appear as a background screen, when a new cloud is coming this is intended to distract participants.

### Input

The Auction Game is simulating the real stock exchange where player is expected to reach a decision in certain amount of time. Decision can be made clicking on the Buy/Sell/No Trade button using the mouse.

### Rewards and ranking

Every profitable buy/sell decision will reward a player with a certain amount of money. Collecting the money to reach level goals and thus progress through the levels, will earn a player with a place on the highscore list. Goal is to reach the highest place on the highscore list. Non-profitable buy/sell decisions will reduce player’s earned money and distance him from the level goal. No Trade gives the player a possibility to skip the decision this turn and start the next round, player makes no profit or loss. Too many money losses and the game ends. Not taking a decision is the most expensive action taking a large sum of profits away from the player.

Players who can achieve targeted emotional arousal level will be rewarded with larger profit/lesser loss money values. On the other hand, unwanted emotional arousal values will yield lesser profit and larger loss.

### Hardware

The Auction game communicates with the biosensor device to acquire player’s emotional arousal level. Dependent of the signal, the game changes the price estimation value deviation from the true price, as well as their movement speed. Movisens wireless device is used to acquire the ECG signal. Acquired signal is processed and HRV arousal level is extracted from it using Java xAffect framework. Mentioned device provides us with an arousal level value through a Java API communicated over TCP/IP network.

### Software

The Auction game runs on a Unity 3d pro game engine. It is an integrated game development platform featuring full blown game engine and editor enabling rapid game development. Unity supports JavaScript, Boo and C# scripting languages, from which C# is used in the prototype. Integration with the third party APIs is supported which makes it the perfect choice for this kind of a prototype. On a second note, Unity automatically compiles the games for different platforms (Windows, Mac, iPhone, Web) so at any time the development targeting platform can be changed.

#### Logging

The actual ECG & HRV signals are logged by the xAffect software in the Unisens format. Thus it makes sense to log every game event in Unisense format as well. This will unify the data and make it easier to analyze it afterwards. So every data point mentioned below has been logged in Unisens data format using provided DLL file from the FZI. Logging has been separated on data concerning whole gaming session and repetitious data from round to round.

Data points concerning whole gaming session are as follows:

- Time stamp of baseline start
- Time stamp of baseline stop
- Time stamp of Game start
- Time stamp of level 0 start
- Time stamp of level 0 stop
- Time stamp of level 1 start
• Time stamp of level 1 stop
• …
• Time Stamp of Game stop
• Overall money earned
• Level reached
• Decisions taken correctly
• Decisions taken incorrectly

Data points concerning repetitious data from round to round

• Subject ID identifier
• Unique Overall Decision Number
• Decision Number in This Level
• Arousal Level in the time of current decision
• Price Estimation 1
• Price Estimation 2
• Price Estimation 3
• Profit/Loss Outcome of This Decision (correct/incorrect)
• Profit Money Earned This Round
• Time stamp of decision
• Time needed for decision-making
• Time interval showing the clouds

2.3.8.2 Implementation
This chapter will describe how the game was implemented internally, depicting mechanisms and architecture elements of the Auction Game.

2.3.9 Use cases and UML

This chapter will introduce: game actors, game actions; sequence through one play through of a game; game architecture elements and their bindings. Diagrams are presented for easier understanding and comprehension.

The Auction game prototype has two actor roles: player and administrator. Administrator sets up the game prototype prior to playing, configuring different elements in the game and preparing it for the player. Player can choose which mode of the game is he or she comfortable playing, simple or advanced one. Player has the option of recording the baseline reading of the biosensor if he or she feels that the game doesn’t respond well to the biosensor data. This is presented on the next Figure 2.21.
Figure 2.21 – Use Case diagram
2.3.10 Sequence diagram

Sequence diagram presented on the next Figure 2.22 depicts a one play through of a game. In the top part, game objects are presented; following the arrows, bindings and communication between those elements is presented. With/without baseline signal logging sections regarding options that a player can choose are presented in the boxes. Auction game loops between multiple rounds and stages presented in the boxes bellow. Actions performed between presented game elements are stated on the arrows.
Figure 2.22 – Sequence diagram
2.3.11 Class diagram

Class diagram presented on the next Figure 2.23 depicts all of the game objects in detail. Their bindings between themselves are presented with the arrows.

![Class diagram](image)

Figure 2.23 – Class diagram

2.3.12 Evaluation

The Auction Game has been thoroughly test using functionality, heuristics and play testing. For details about the evaluation, please see D17-4.1.

2.3.12.1 Functionality evaluation

The functionality evaluation consists of three categories: validation, verification, and future support. The validation was done together with the product owner (FZI) at a face-to-face meeting where the product was demonstrated. By letting the product owner test it before it was finalized much of the rather abstract discussions become very concrete and much feedback could be given, also misconceptions due to possible miscommunication was solved at this point.

The verification was done by the development team throughout the development process. By working in an incremental development style the product was constantly tested for errors when new things are added, thus minimizing latent errors.

Measurement of arousal is done by a separate component, thus allowing future support for other sensors to be used within the same game. Many variables are set in the Unity 3D environment, making them easy to adjust for future studies.
2.3.12.2 Heuristic evaluation

Development of the Auction Game was followed by a heuristic evaluation. This evaluation aims at qualitatively identify design errors and suggest improvements. The heuristic evaluation used on the Auction Game is part of a generic evaluation tool kit which is being developed and used on all prototypes produced at BTH developed within xDelia (Isbister & Schaffer, 2008). The heuristics are divided into a set of categories inspecting different aspects of the game prototype. Framework and evaluation results are presented below. If two or more reviewers agree on an issue, it’s immediately noted.

Heuristic evaluation pinpointed several important design issues and reinforced the robustness of the game. For more detail about heuristic evaluation please refer to the DOC-AUC-5 document (Results of heuristic evaluation).

2.3.12.3 Playtesting evaluation

A total of six students volunteered to participate in the Playtesting Evaluation. They were all students of BTH Institute aged between 20 and 32 years old with four of them being male and two female. They reported varying gaming experience.

Before the game, the students were fitted with the Movisens ecgMove HR sensor and given a tutorial session. In order to objectively determine which game elements the players are paying attention to, the game was played through the Tobii T60 eyetracker logging data on different Areas of Interest (AOI) and recording the whole gaming session on video. The purpose is to be able to tell how important different visual objects (AOIs) on screen are to a player. At the end of each game, each participant was given: an Emotion Regulation Questionnaire (Gross et al., 2003) in order to identify suppression and reappraisal tendencies of individuals; a modified System Usability Scale (Nacke, 2010; Brooke, 1996) questionnaire measuring game usability. The questionnaire contained 10 questions whose score was summed up in a single number representing a composite measure of the overall usability of the game being studied; an interview session where participants could openly discuss perceived game speed and difficulty, as well as visual cue elements and any issue they want to note.

2.3.12.4 Playtesting Evaluation Results

The Auction Game scored a mean value of 67.92 in a range from 0 to 100 on modified SUS questionnaire. Thus according to Tullis (2008) where a score of 60 presents a border between poor and average usability, we can conclude that the game fulfills the average game usability.

The game was successfully played up to a hard 8th level by two participants, both of which are high reappraisers while one had low and the other normal suppression tendencies. They both evaluated the game as manageable and in the interview session reported that they were practicing emotion regulation techniques themselves without being instructed at all. This provides evidence for a good game design of the Auction Game. Rest of the participants got stuck on the first level struggling with the game mechanics and speed. This provides evidence for making the beginning levels a bit easier so every participant can experience motivation of progress.

Eyetracker gaze data analysis clearly identified areas of player interest which perfectly correlated with visual cue elements presented on the screen. Identified elements are: estimation clouds in the center of the screen; buy/sell buttons in the bottom part of the screen; arousal meter indicator in the top right corner; money pile in the bottom right corner of the screen.

Because of the fast pace of the game, five out of six participants reported that they were not paying attention at all to the arousal meter indicator present at the top-right of the screen. We evaluated this claim on how informed were the participants about their arousal level by keeping track of it on the arousal meter indicator during rounds in the whole gaming session. A paired-samples t-test was concluded on the eyetracker data to evaluate the difference in number of gaze
observations on marked indicator arousal meter AOI compared to number of rounds taken for each participant. There was a statistical significant difference found with number of rounds (M=110.17, SD=95.26) to number of arousal meter observations (M=16.33, SD=24.5, t(5)=2.94, p=.032). Thus we can say that participants paid little or no attention on the arousal meter indicator during the whole playing session. To rule out the fact that two advanced participants used the information on the arousal meter indicator to perform better in the game, a similar test was concluded comparing focus time on marked indicator arousal meter AOI for each participant where no significant difference was found. Participants reported that the reason for paying little or no attention on the arousal meter indicator was lack of time during fast paced decisions. Most of the participants reported that they were paying attention to the arousal indicated by the color of the cloud estimations, especially when it turned red. This gives evidence to hide or remove arousal meter indicator and concentrate on making the color coded cloud estimations more distinct.

Table 5: Money earned and Time needed to reach a decision at different arousal levels

<table>
<thead>
<tr>
<th>Arousal level while making a decision (Groups)</th>
<th>Money earned</th>
<th>Time needed for decision making</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. dev.</td>
</tr>
<tr>
<td>1</td>
<td>0.632776</td>
<td>2.94760</td>
</tr>
<tr>
<td>2</td>
<td>.4638</td>
<td>3.24442</td>
</tr>
<tr>
<td>3</td>
<td>-0.4660</td>
<td>2.81549</td>
</tr>
<tr>
<td>4</td>
<td>0.6650</td>
<td>3.29969</td>
</tr>
<tr>
<td>5</td>
<td>-1.3569</td>
<td>3.02566</td>
</tr>
</tbody>
</table>

A one-way between-groups analysis of variances was conducted to explore the impact of arousal level on profit in each round (see Table 5). Total number rounds played, 661 rounds, were divided into 5 groups according to their arousal level while making a decision (Group 1: 1[relaxed], Group 2: 2 ... Group 5: 5[highly aroused]). There was a statistically significant difference at the p<.05 level in profit made each round for the five arousal groups [F(4, 656)=3.566, p=.007]. The effect size, calculated using eta squared was .02. Post-hoc comparison using Turkey HSD test indicated that the mean score for Group 1 (M=.6328, SD=2.95) was significantly different from Group 5 (M=-1.369, SD=3.3). Other groups did not differ significantly. Same has been conducted for the time needed to reach a decision in seconds and there was a statistically significant difference the p<.05 level [F(4, 656)=5.753, p=.000] between Group 5 (M=1.55, SD=.45) and rest of the groups. The effect size, calculated using eta squared was .03. This gives strong evidence supporting a good design of the Auction Game to reward a player achieving a target arousal level increasing his earned profit, while at the same time presenting a hard challenge and punishment to a player in an undesirable high arousal emotional state.

2.3.13 Discussion and conclusion

Evidence shows that emotions have an in impact on decision making, especially in the field of finance. Thus it makes sense to develop a tool to get people aware of this implication as well as to help them regulate their emotions to reach better financial decisions. A serious game emerged as a perfect tool for the job where players get instant emotional arousal biofeedback information on screen which they can interact with, that is regulate it. The Auction Game is a serious game where a player buys or sells stocks with the objective to train emotion regulation; but also to get them aware of the arising emotions. To support this, achieving a target arousal level will reward the player accordingly, increasing his earned profit.
During fast paced decisions, players concentrate on the information presented and do not have enough time to shift their attention away from the task at hand. As participants reported and data showed, they paid little or no attention on the arousal meter indicator, while the arousal indicated by the color of the cloud estimations was perceived, especially when it turned red on high arousal values. This is why we concentrated on making the color of the cloud estimations more distinct, since players are focusing their concentration on them. Further studies should identify how to optimally present the arousal information to the cognitively engaged player during fast paced decisions.

In the interview session novice participants perceived the game as too fast/just bearable, while on the other hand, two successful participants perceived it as manageable as they were practicing emotion regulation (relaxation) techniques. Those successful participants report that the game was quite overwhelming at some points where they needed to let a decision pass, thus losing profit, in order to relax and win in the next set of decision. This gives evidence that the Auction Game is indeed overwhelming and putting players at highly aroused state where they need to practice emotion regulation techniques to succeed in the game. But in further studies we need to explore finding an optimal speed to accommodate novice players and thus leave enough time and space for them to practice emotion regulation, while at the same time keep the game overwhelming and highly arousing. One way to realize this would be to use only integral numbers for calculation in the beginning levels, as some of the participants stated that they were only calculating with integral parts of the real numbers, not the fractional decimal part. On the other hand, it has been identified in the study that the game gets harder when the offer price and true price of the round are close together. This could potentially be used to make the game harder at higher arousal/levels.

We have demonstrated in the Auction Game, how one can reward a player achieving a target arousal level increasing, while at the same time presenting a hard challenge and punishment to a player in an undesirable high arousal emotional state. Through this experience emotion regulation can be learned and practiced. We still need to balance the game in regards to arousal levels to make the gameplay for arousal levels significantly different from each other, not just highest and lower arousal level as it is now.

We have demonstrated that the Auction Game was successful at reaching its goals as a study tool, as well as a usable game. If we can systematically succeed in this, we can make learning in behavioral economics more fun and effective.

2.4 Two Index Game

2.4.1 Introduction

The Two Index game is a game where the player has to buy or sell stock shares. Two indices are provided, one with current prices and one predictor index, which partially determine the tradable stock value. The player’s assignment is to try and predict the price based on what he or she can read in the predictor index chart; furthermore, buy and sell the assets at appropriate times in order to make a maximum profit. The idea is that the indices give indications of when it may be a good time to sell, but dependent on the present market value the player may deviate from their personal trading rules. The game logs the player data in order to determine if, and how severely the player is suffering from the disposition effect.

The objective of the game is to make traders aware of the disposition effect and to train them to regulate such biases. Hence it is both a diagnostic and a didactic game.

2.4.1.1 Purpose

A study of verbal expressions and emotions performed by Shimanoff (1984) found that regret was the most frequently named negative emotion. Similar statement has been confirmed by Lankman (1993, p.110) where regret was found to be common, if not universal, experience. It was found that investors’ psychology tend to feel sorrow and grief after having made an error in judgment (Yahyazadehfar, Gheyekhloo, &
Sadeghi, 1985). They found that the investors deciding whether to sell the stock are typically emotionally affected by whether the stock was bought for more of less than the current price. Therefore, the investor must recognize this fact and try to practice some mechanism to control his (her) irrational behavior.

In economical investing emotional factors play an important role. A study in the domain of losses has been conducted measuring investors experienced regret and (dis)satisfaction during incurring various sizes of gains or losses over multiple periods, and link these emotions to their subsequent hold/sell decisions (Lee, Kraeussl, & Paas, 2009). It has been found that in the loss domain, only regret has significant impact on invertors’ subsequent hold/sell decisions. Furthermore, they report that high experienced regret and pride are linked to a larger probability to sell; as for anticipated pride leads to a larger probability to sell, while anticipated regret links to a smaller probability to sell. Their findings demonstrate that investors’ subsequent hold/sell decisions are simultaneously affected by their experienced and anticipated emotions. In a similar study it has been found that alternative outcomes may affect evaluation of decisions (Fogel & Berry, 2006). Anticipation of regret may lead investors into the trap of holding losing stocks too long. All of respondents in their study reported regret for investment decisions, either for not selling a losing stock soon enough, or for selling a winning stock too soon.

Summers & Duxbury (2007) found that emotional response is clearly important in explaining behavior in disposition effect and that manipulating emotions changes behavior. This provides evidence of the importance of emotion in economic behavior.

Regret is a prominent example for the impact of emotions in financial decision making (Engelbrecht-Wiggans & Katok, 2008). Moreover regret is believed to have a major impact for several biases in financial decision making e.g. the disposition effect. We measure skin conductance and heart rate; with these measures we seek to improve the understanding of the emotional processing in economic decision making. Additionally, questionnaires examine qualitative differences in emotion and emotion regulation between the subjects. These insights in turn can help understanding traders’ behavior in the field, where regret also is believed to play a major role.
2.4.2 Description of Prototype

The index game starts with a tutorial where the player gets instructions of how the game works, which buttons to use and how to play the game (Figure 2.24).

As mentioned the main parts of the game are the two indices. The underlying principle of the game is that the future value of the tradeable stock is based upon the share price of an index that it follows according to a partially predictable function. On the top half of the screen the player sees a chart showing a relationship between time and price for the tradeable shares; this index shifts regularly, at a rather fast pace.

On the bottom half of the screen a similar index is shown for the stock that partially determines the tradeable stock value (Figure 2.25) The player’s assignment is to try to predict the price based on what he or she can see in the lower chart and buy and sell their assets at appropriate times in order to make a maximum profit. The player gets time slots when he or she can sell or buy, indicated by an audio and/or visual cue. A more elaborate description of the logic behind the game can be found in Appendix A.

Figure 2.24 – Screenshot of Tutorial
The score generated is how well the player manages the assets, profits and losses, but also how well the player handles biases such as the disposition effect. In addition to this, motivational quests are present in the game. Distance from maximum show the player how far from playing optimally the player is, money is the current wins and losses summarized, both these have no importance to the game more than act as motivation.

### 2.4.3 Prototype specification in summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Two Index game</td>
</tr>
<tr>
<td>Version</td>
<td>2</td>
</tr>
<tr>
<td>Status</td>
<td>Developed</td>
</tr>
<tr>
<td>Target User Group</td>
<td>Investors</td>
</tr>
<tr>
<td>Prototype User Group</td>
<td>Internal + students</td>
</tr>
<tr>
<td>Purpose</td>
<td>To train managing bias (disposition effect), but also to get aware of the arising emotions.</td>
</tr>
<tr>
<td>Description</td>
<td>The player has to future prices on a visually represented price index based upon the value of a related index that is also displayed. On the top of the screen the player sees a chart showing the index as a relationship between time and price and the index increments regularly, at a rather fast pace. The evolution of this index has a partially stochastic relationship to the value of the other index. The</td>
</tr>
</tbody>
</table>
players’ assignment is to try to maximize their profit based on what they can see in the related index chart.

<table>
<thead>
<tr>
<th>Transfer</th>
<th>Yet unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context of Use</td>
<td>In training situations at day trading centers or on-line</td>
</tr>
<tr>
<td>Appearance</td>
<td>Two indices giving two different perspectives of the price</td>
</tr>
<tr>
<td>Inputs (inc. sensors)</td>
<td>Mouse and keyboard.</td>
</tr>
<tr>
<td>Guidance</td>
<td>Before the player starts to play he or she has to go through the tutorial to learn how to play and what the purpose of the game is. It is possible to skip the instructions to facilitate for players that play the game more than once.</td>
</tr>
<tr>
<td>Gameplay/ Challenges</td>
<td>To predict when to sell and when to buy, and to keep to personal trading rules.</td>
</tr>
<tr>
<td>Logging devices</td>
<td>Cumulative score logging. Logging of disposition effect.</td>
</tr>
<tr>
<td>Platform</td>
<td>PC</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Web based application, single player</td>
</tr>
<tr>
<td>Testing</td>
<td>Heuristic evaluation and playtesting by project members, Saxo Bank employees and students.</td>
</tr>
</tbody>
</table>

2.4.4 Evaluation

2.4.4.1 Play Testing

The process to evaluate the Two Index game follows the same procedure as for the Aiming game and the Auction game (see section 2.2 and 2.3). The analysis of the results of the Two Index game is not finished why only some initial results are reported. During the past week project members as well as employees at Saxo Bank have play tested the game and the data from those tests are not incorporated in the results below. The data presented are from six students and colleagues play testing the game at BTH. They started to learn how to play the game by playing the tutorial and thereafter playing one session of the Two Index game (released 23 February 2011). The players had to answer the Game Experience Questionnaire (GEQ) as well as the System Usability Scale (SUS) (see appendix C and D). It is the results from those questionnaires that are reported below. In addition the participants were interviewed after they have filled in the questionnaire and a brief summary of the outcome of the interviews is reported in Appendix B.

2.4.4.1.1 Results from the GEQ

The questions in the GEQ can be grouped to get a measure on how immersed the player felt when playing, if the player came into flow, how competent the player experienced him- or herself, how tens the player felt and how the player experienced the challenges in the game (Nacke & Schild, 2010). The grouping can be found in Table 2.6.
<table>
<thead>
<tr>
<th>Factor</th>
<th>No</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Immersion</td>
<td>3</td>
<td>I was interested in the game’s story</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>It was aesthetically pleasing</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>I felt imaginative</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>I felt that I could explore things</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>I found it impressive</td>
</tr>
<tr>
<td></td>
<td>33</td>
<td>It felt like a rich experience</td>
</tr>
<tr>
<td>Flow</td>
<td>5</td>
<td>I felt completely absorbed</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>I forgot everything around me</td>
</tr>
<tr>
<td></td>
<td>28</td>
<td>I lost track of time</td>
</tr>
<tr>
<td></td>
<td>31</td>
<td>I was deeply concentrated in the game</td>
</tr>
<tr>
<td></td>
<td>34</td>
<td>I lost connection with the outside world</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>I was fully occupied with the game</td>
</tr>
<tr>
<td>Competence</td>
<td>2</td>
<td>I felt skillful</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>I felt strong</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>I was good at it</td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>I felt successful</td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>I was fast at reaching the game’s targets</td>
</tr>
<tr>
<td></td>
<td>42</td>
<td>I felt competent</td>
</tr>
<tr>
<td>Tension</td>
<td>7</td>
<td>I felt tense</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>I felt restless</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>I felt annoyed</td>
</tr>
<tr>
<td></td>
<td>27</td>
<td>I felt irritable</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td>I felt frustrated</td>
</tr>
<tr>
<td></td>
<td>39</td>
<td>I felt pressured</td>
</tr>
<tr>
<td>Challenge</td>
<td>8</td>
<td>I felt that I was learning</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>I thought it was hard</td>
</tr>
<tr>
<td></td>
<td>26</td>
<td>I felt stimulated</td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>I felt challenged</td>
</tr>
<tr>
<td></td>
<td>36</td>
<td>I had to put a lot of effort into it</td>
</tr>
<tr>
<td></td>
<td>37</td>
<td>I felt time pressure</td>
</tr>
<tr>
<td>Positive affect</td>
<td>1</td>
<td>I felt content</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>I could laugh about it</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>I felt happy</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>I felt good</td>
</tr>
<tr>
<td></td>
<td>22</td>
<td>I enjoyed it</td>
</tr>
<tr>
<td></td>
<td>41</td>
<td>I thought it was fun</td>
</tr>
<tr>
<td>Negative affect</td>
<td>10</td>
<td>I thought about other things</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>I found it tiresome</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>I felt bored</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>I was distracted</td>
</tr>
<tr>
<td></td>
<td>35</td>
<td>I was bored by the story</td>
</tr>
<tr>
<td></td>
<td>38</td>
<td>It gave me a bad mood</td>
</tr>
</tbody>
</table>

*Table 2.6 – Grouping of the GEQ*
The diagrams below show how the six participants experience the game.

2.4.4.1.2 Discussion of Results
When looking at the results, two dividing camps can be seen. Player 1, 2 and 5 seemed to feel more competent, less tensed and much more positive towards the game. Player 3, 4, and 6 felt on the other hand less competent and much less positive. However, it is interesting that these do not translate into tension, flow or immersion in the game play. From the discussion with the players afterwards it seems like it mostly had to do with usability issues, such as that the affordances of the different buttons were not clear enough. Additional instructions needs to be added, and also this data need to be compared against game performance in order to see if being competent in the game gave an effect in the positive affect.
2.4.4.1.3 Results from SUS

The System Usability Scale consists of questions about how the user experiences the usability of the system. The Scale is designed for ordinary software and to adapt it for games, the Word system is replaced by game system (Nacke & Schild, 2010). The questions can be found in Appendix D.

In Figure 2.26 the results are summarized, the score is from “1 - strongly disagree” to “5 – strongly agree”. As we can see the score on the different questions varies a lot, this is probably based in the preconditions of the players since the game looks very much alike an investor environment and do not have the usual characteristics of a game. What the results indicate is that there is a need to study the usability more in detail to determine what the problem is, and how to address it for those player who did not understand the system. Many of these results are also influenced that the game is perceived as non-deterministic and would thus result in less confidence when using the system.

<table>
<thead>
<tr>
<th>I think that I would like to use this system frequently.</th>
<th>I found the system unnecessarily complex.</th>
<th>I thought the system was easy to use.</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Bar Chart" /></td>
<td><img src="image2.png" alt="Bar Chart" /></td>
<td><img src="image3.png" alt="Bar Chart" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>I think that I would need the support ...</th>
<th>I thought there was too much inconsistency ...</th>
<th>…...learn to use this system very quickly.</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image4.png" alt="Bar Chart" /></td>
<td><img src="image5.png" alt="Bar Chart" /></td>
<td><img src="image6.png" alt="Bar Chart" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>I found the system very cumbersome to use.</th>
<th>I felt very confident using the system.</th>
<th>I needed to learn a lot ...</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image7.png" alt="Bar Chart" /></td>
<td><img src="image8.png" alt="Bar Chart" /></td>
<td><img src="image9.png" alt="Bar Chart" /></td>
</tr>
</tbody>
</table>

Figure 2.26 – Results from SUS

2.4.4.2 Summary

The data is as mentioned not analysed fully and therefore it is hard to propose design strategies based on the data. What can be said is that the variations in the answers can depend on the nature of the game in
combination with the different backgrounds and experience of the participants. What can be said is that the participants experienced flow when playing, and they felt challenged and tense. Flow and challenge is important factors for games to be successful (Kiili, 2005) which is a positive result.

The usability questionnaire did not give any guidance and the usability issues must be explored further through more detailed questions regarding the difference in skill and usability in the game. The data from the play testing will during the upcoming weeks be analysed in detail and summarized in terms of what the data implicate when it comes to further development.

### 2.5 Mindfulness Game

#### 2.5.1 Prototype Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mindformer</td>
</tr>
<tr>
<td>Version</td>
<td>1</td>
</tr>
<tr>
<td>Status</td>
<td>Developed</td>
</tr>
<tr>
<td>Target User Group</td>
<td>Investors</td>
</tr>
<tr>
<td>Prototype User Group</td>
<td>Internal + students</td>
</tr>
<tr>
<td>Purpose</td>
<td>The purpose is to give the player an opportunity to try out some basic mindfulness exercises and experience how mindfulness can be practiced in stressful situations. The purpose is also to provide an environment where the player can train mindfulness for a couple of minutes in a context separate from everyday life.</td>
</tr>
<tr>
<td>Description</td>
<td>The game is a platform game where the player has to guide a character from platform to platform while collecting xDelia coins. The game consists of six levels with different difficulty and mindfulness tasks to perform. In between the levels the player can practice paced breathing to get more mindful about the situation.</td>
</tr>
<tr>
<td>Transfer</td>
<td>Yet unknown</td>
</tr>
<tr>
<td>Context of Use</td>
<td>On-line</td>
</tr>
<tr>
<td>Appearance</td>
<td>2D world with a naïve appearance.</td>
</tr>
<tr>
<td>Inputs (inc. sensors)</td>
<td>Mouse and keyboard.</td>
</tr>
<tr>
<td>Guidance</td>
<td>The player gets guidance in-between the levels</td>
</tr>
<tr>
<td>Gameplay/ Challenges</td>
<td>Practice mindfulness like awareness, attention and paced breathing.</td>
</tr>
<tr>
<td>Logging devices</td>
<td>Logging of all user input</td>
</tr>
<tr>
<td>Platform</td>
<td>PC</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Web based application, single player</td>
</tr>
<tr>
<td>Testing</td>
<td>Playtesting by project members and students.</td>
</tr>
</tbody>
</table>
2.5.2 Introduction

Mindfulness is associated with general improvements in well-being, emotions regulation and coping with stress. In this game, simple mindfulness techniques will be implemented in a game setting. This will allow the project to determine whether such simple, short-lasting mindfulness techniques may already have specific behavioural improvements associated with them (e.g., improved performance on a stressful game).

As part of the xDelia learning journey (D18-2.4.1 Intervention package – specification and development), participants can engage in a mindfulness training. In this training, the participants will conduct a number of exercises over a long period of time (i.e. 4 weeks). The game described in this section is part of the learning journey involving mindfulness. The game will above all be an introduction to the training and a pragmatic yet experiential introduction to the mindfulness course for novice users.

The game is a “platform” game (see: http://en.wikipedia.org/wiki/Platform_game ) where the player has to guide an on-screen character to jump from platform to platform to continue playing. Bonuses will be accrued by collecting items on screen, and penalties will occur if the player fails to guide their character from one platform to the next, falling to the ground.

The game consists of six levels. Each level will be more stressful than the previous, operationalized by higher speed movement of the platforms, more complex obstacles, different tasks to accomplish or more difficult placing of bonuses.

Between the levels the player gets instructions of what to do in the next level and after level two the player can practice paced breathing to experience what mindfulness exercises can add to the performance.

The choice of the game genre is based on the following criteria:

- The game must be playable from the start even for users without game experience.
- No tutorial should be needed to be able to play the game successfully.
- The player can get involved and engaged in the game immediately.
- The game should be easily extended with new mindfulness exercises even within the game itself.
- It must be easy to make the game stressful.

A platform game fulfills the criteria above. It is a well know, and in its basic form, self-explaining genre and the player may understand what to do just by looking at the screen. It is very easy to make a platform game stressful just by, for example, making the platforms scroll faster and faster. In addition it is easy to incorporate mindfulness exercises in the game by for example make some platforms retreats or quests that the player can choose to jump on whenever the player feels there is a need for it.

2.5.2.1 Purpose and Research Question

The project goal is to design a learning intervention where people experience that learning emotion regulation is beneficial to decision-making processes. Hereto, xDelia has developed a mindfulness training for day traders. In this mindfulness training, participants will learn to pay attention to their emotions and be aware of emotional fluctuations (e.g., in arousal). Consequently, this heightened awareness and increased attention level is hypothesized to influence participants’ decision-making.

The purpose to create Mindformer is for it to be an interesting teaser for people interested in the mindfulness training to try out and learn very quickly and experientially about some of the principles in the full training course.

The game provides an environment which requires the player to play a multi-levelled game, lasting (10-30 minutes), with mindfulness exercises undertaken between each game level. A relevant question is then if
simple, short-lasting mindfulness techniques such as within Mindformer have specific behavioral improvements associated with them (e.g., improved performance on a stressful game).

2.5.3 Game Description

The game is a platform game with six levels where each level is different from the other. In between the levels there are some explanations of mindfulness and a possibility to do a paced breathing exercise. Some of the levels contain mindfulness tasks aiming for training attention and awareness.

The player gets 5000 points at the beginning of each level and as soon as the game has started the time spent will be withdrawn from the score. To get some extra time (i.e. score) the player must collect xDelia coins, perform the paced breathing exercise or perform well on the mindfulness tasks. After the last level the player must relate to three statements. This is something that is important from a research point of view.

2.5.3.1 Goal of the game

The goal of the game is to score as high as possible and to improve when playing the game repeatedly. The player must master the levels quickly and collect as many xDelia coins as possible. As the player get 5000 points at the start of each level and the time it takes to finish a level will be withdrawn from this amount the player also has to collect the coins to add to the sum which will result in a higher end score.

The goal of playing the game is to learn about and get curious about mindfulness.

2.5.3.2 Story and theme

There is no story in the game. The game is more of a metaphor for life. Life has a tendency to run faster and faster in the hunt to perform better and better. Now and then we need to slow down and be mindful within the situation and pay attention to what is important just like in the game.

2.5.3.3 Gameplay

Player controls an onscreen character that must jump from platform to platform. The player has run and jump controls, via keyboard.

During gameplay, extra bonuses (xDelia coins) appear on screen. The player has to collect as many coins as possible.

If the character fails to reach the next platform, a penalty occurs, the number of deaths increases and the player starts all over again by falling down from the sky. Treats/bonuses collected remains.

The player gets a bag of points from the beginning and for each life he or she loses this amount decreases. The player has infinite lives, but less life lost is an advantage.

The mindfulness exercises are short and concise forms of the training exercises that are described in detail in the full training course. So, instead of a full hour-long meditation, a paced-breathing exercise such as implemented in the xDelia Mind Manager (section 2.6) for the learning intervention.

In addition to the mindfulness exercises, there will be hidden psychological tasks during the game that have been associated with mindfulness. For example pay attention to objects and sound.

2.5.3.4 Game Mechanics

The game mechanics are the rules of the game.

2.5.3.4.1 Game logic
Mindformer is a single player game where the player can play the game as long as his or her score is above zero. Every time the player dies he loses points. The core flow in the game is to jump – collect coins – land on a platform – jump again and so on (Figure 2.27). As long as there are points left the loop will continue to the end of the level. If the character dies he will start with a new jump from the sky and the player must see to that the character lands on a platform. For each second passed the score will decrease with one point and when the character dies 1 000 points will be withdrawn from the total score. For each collected coin the score is added with 5 points. The player can also gain point from doing the paced breathing exercise (1500 points) and from performing well on some mindfulness tasks. The questionnaire does not generate any bonuses.

![Figure 2.27 – Game flow](image)

### 2.5.3.4.2 Game difficulties

The first level act as a tutorial where the player can get used to making the character jump from platform to platform. The player has infinite lives (except for level 2) and can guide the character to the end of the level. The only limitation is that the character must reach the end of the level before the scores have gone down to zero. In level two the platforms will scroll faster and faster, and the player only has three lives. The speed of the platform when the character dies the third time will be set as the maximum speed of level 3 and 6. In this way the player’s skills will set the difficulty of the levels. The game should be experienced as challenging, but not impossible to master.

### 2.5.3.4.3 Levels

Table 2.7 summarizes how the different levels look like.

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
<th>Comment</th>
<th>Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>The platforms move when the player moves, the player jumps on the platforms and collects coins.</td>
<td>An ordinary platform game tutorial type level.</td>
<td>1-2</td>
</tr>
<tr>
<td>Level 2</td>
<td>The game gets slowly faster and faster, but at some point, the player falls off.</td>
<td>This level functions as a baseline, whereas level 1 is more of a tutorial function.</td>
<td>2</td>
</tr>
<tr>
<td>Retreat A</td>
<td>Breathing exercise</td>
<td>The pace must first be calibrated by the user holding down the B-key while inhaling. A circle will expand and shrink according to the calibrated pace. The player follows the</td>
<td>2-3</td>
</tr>
<tr>
<td>Level 3</td>
<td>Same as level 2, but the speed when the player falls off in level 2 will be the maximum speed in this level.</td>
<td>Here, due to breathing exercise, the idea is that players a) achieve a greater distance, b) longer time period, c) collect more coins etc. When the player has done the breathing exercise and calibrated his or her breathing the breathing pattern will be visualized in the game (darkening clouds).</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Level 4</td>
<td>Before the game starts the player is told to count the number of platforms he or she jumps on. The platforms behave as in level 1, but objects, not part of the gameplay, moves across the screen. After the level the answers to the tasks are presented, that is the number of platform the player did jump on. There is also a question if the player saw the objects.</td>
<td>The player is likely to see the big object (dinosaur), but maybe not the small ones (flying ducks). The player doesn’t know that he is supposed to keep track of the objects. Additional objects (different rabbits) are also added, but not important for the gameplay. Visualization of breathing pace. Possibility to do the breathing exercise before starting.</td>
<td></td>
</tr>
<tr>
<td>Level 5</td>
<td>Before the level starts the player is told to listen for a specific sound and push the key ‘B’ when he or she hears it to get a bonus. The platforms behave as in level 1 and in addition the player has to react on sounds.</td>
<td>There is a button to click if the user wants to hear the sounds before the level starts. The sound is significant different from the music and sound effects in the game. But the sound should be subtle and associated with the game environment (e.g. bird tweeting) Visualization of breathing pace. Possibility to do the breathing exercise before starting.</td>
<td></td>
</tr>
<tr>
<td>Level 6</td>
<td>A combination of level 3, 4 and 5</td>
<td>Possibility to do the breathing exercise before starting.</td>
<td></td>
</tr>
<tr>
<td>End</td>
<td>The score of the last level is presented. Tip to use breathing exercise in daily work or take the mindfulness course.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.7 – Overview of levels

2.5.4 Software

The game is developed in an iterative process with frequent meetings with developers and other partners. The requirements have been established during the development process (Table 2.8). The purpose of the requirements was to support the communication between developers and non-developers in the team. During the meetings the game was demonstrated, tried out and discussed, and changes were made in line with the discussions. The technical solutions are derived from the user requirements and decided on during the meetings.

<table>
<thead>
<tr>
<th>No</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>There must be mindfulness exercises and tasks between levels.</td>
<td>Functionality</td>
</tr>
<tr>
<td>R2</td>
<td>The first level must have a constant speed.</td>
<td>Functionality</td>
</tr>
<tr>
<td>R3</td>
<td>The second level must go faster and faster and have a finite number of tries (lives). How far the character has moved will determine what the max speed is for the rest of the levels.</td>
<td>Functionality</td>
</tr>
</tbody>
</table>
R4  The platform game is played by using keyboard.  
Functionality

R5  Game data is to be saved to a database. Level number, life left, treats collected, time to complete must be collected for each level. The number of times and for how long the breathing exercise is done and the results from the other exercises are logged.  
Functionality

R6  The platform game must be kept very simple and just implement jumping between platforms and collecting items and faster scrolling platforms, and simple attention objects.  
Functionality

R7  Score is calculated based on time to complete, points/coins collected, lives left and how well the mindfulness exercises are performed.  
Functionality

R8  The game must give a hint in the end to try out the mindfulness course.  
Functionality

R9  There must be a welcome screen with options to choose to learn about the game, how to play and play the game.  
Functionality

R10  A local high score list is kept.  
Functionality

R11  Short questionnaire at the end of the game on mindfulness state.  
Functionality

R12  The game must provide the score to the player at the completion of play and between every level.  
Functionality

R13  In Level 3-6 the breathing pace from the breathing exercise must be visual while playing.  
Functionality

R14  The game must be stressful  
General

R15  The game must be completed in 10-30 minutes  
General

R16  Music and sound effects is necessary. The player will need a computer with a sound card; however it will be possible to play the game without sound. Except the mindfulness task in level 5 cannot be done.  
General

R17  There must be an in-game description of the game in terms of intentions and goals.  
General

R18  There must be instructions of how to play the game.  
General

R19  Simple visual aesthetic.  
General

R20  Single player game  
General

R21  Six levels of gameplay.  
General

R22  The algorithm for calculating the scores looks like this: The player starts with score: 5000 (initialScore).  
Score = nrOfCoins * 5 - nrOfseconds1 - nrOfLifesLost * initialScore / 5.  
General

R23  “Participant ID” must be collected: presented as “Your Name” entry field at the beginning of the game. - For home players, this is just their name (“Jane Smith” etc. …). For xDelia studies participants can be told to enter a name they are given and it could be a more technical, unique ID which will include extra information if required (e.g. “Smith_Jane-2011-12-08_001-EUR”). This can still be entered into the “Your Name” field. Therefore it is important that the “Your Name” field is set to allow a large number of characters to be entered (UTF8))  
General

R24  The game shall be developed in Flash.  
Technology

R25  The game will run within a 1024 x 768 window  
Technology

R26  The game is played on a standard computer without sensors.  
Technology

Table 2.8 - Requirements
2.5.4.1 Use cases

This section will describe two use cases divided into sub use cases (Figure 2.28).

![Figure 2.28 – Use cases](image)

Use case 1: Play game

The user has heard about mindfulness and what it can do to for his or her decision making. Therefore the user wants to learn what mindfulness can be and he or she decides to try out the Mindformer to see if mindfulness is something for him or her.

The user clicks on the link on the web site and the game launches. The first thing the player sees is a welcome screen, by clicking on the start button the player can learn more about the game. The text describes what the game is about, the intention with the game, the game’s relation to the overall learning intervention and the mindfulness game ("About the game"). The next step is to learn how to play the game. The player meets the rules, how to score and the goals of the game ("How to play") (Figure 2.29). The last thing before starting to play, the player is asked to enter his or her name. There is also an option to quit the game and this is possible in-between all of the levels. The player is then presented to the quit screen (Figure 2.30) with an option to play anyway if the player has changed his or her mind. When the game starts the character is placed to the left on the screen and as soon as the player presses the “run” key the platforms starts to move and it is up to the player to make the character jump from platform to platform. The different levels are described in the sub use cases. When the player has finished the six levels he or she will be presented with the total score. The score is presented in a high score list.
Figure 2.29 Start of game

Figure 2.30 – Quit screen
1.1: Play level 1
This level can be seen as a tutorial level. The platforms scroll according to the character’s movements and if the player stops so do the platforms.

- The player character has its start position to the left.
- The player jumps from platform to platform to get from start to finish.
- In between and on platforms the player can collect coins.
- The platforms scroll in the same speed as the player moves.
- If the player misses a platform or fall off a platform the character falls out of the screen. Thereafter the character comes falling from the top of the screen again and the player must see to that the character lands on a platform to avoid dying immediately again.
- When the character has reached the end of the level an ‘End of the level’ screen is displayed with instruction of what to do in the next level (Figure 2.31).

![Figure 2.31 – Level 1 with end of level](image)

1.2: Play level 2 (grey text indicates that the same feature is available in the previous level.)
This level is a baseline for how fast the platforms should scroll to make the game challenging but not impossible to master.

- The player character has its start position to the left.
- The player jumps from platform to platform to get from start to finish.
- In between and on platforms the player can collect coins.
- If the player misses a platform or fall off a platform the character falls out of the screen. Thereafter the character comes falling from the top of the screen again and the player must see to that the character lands on a platform to avoid dying immediately again.
- If the character misses a platform or fall off a platform the player loses a life. The player has three lives altogether to finish the level.
- The game scrolls faster and faster.
- The speed when the player falls off the third time the fastest try will be the maximum speed of the rest of the levels.
- When the character has reached the end of the level an ‘End of the level’ screen is displayed with instruction of what to do in the next level (Figure 2.32).
1.3: Practice paced breathing

The player chooses to do the paced breathing exercise. The player is told to inhale while he or she is pressing the B-key. A red circle will show and increase its radius until the key is released. The pace of the paced breathing is set and the red circle will shrink again. After the calibration a blue circle will show and the circle will shrink and expand according to the calibrated breathing pace. The player follows the pattern by inhaling while the circle expands and exhale when it shrinks. The player can adjust the pace by clicking on the arrow keys (up and down) (Figure 2.33).
1.4: Play level 3 (grey text indicates that the same feature is available in the previous level.)
This level is a level where the player can focus on jumping and collecting coins while practicing paced breathing.

- The player character has its start position to the left.
- The player jumps from platform to platform to get from start to finish.
- In between and on platforms the player can collect coins.
- If the player misses a platform or fall off a platform the character falls out of the screen. Thereafter the character comes falling from the top of the screen again and the player must see to that the character lands on a platform to avoid dying immediately again.
- The game scrolls faster and faster and the maximum speed is set to the speed from level 2 where the character died the third time.
- The clouds will get brighter and darker in the same pace as the player’s calibrated breathing pace and the player can therefore practice paced breathing while playing and thus hopefully perform better.
- When the character has reach the end of the level an ‘End of the level’ screen is displayed with instruction of what to do in the next level (Figure 2.34).
- At the ‘End of level’ screen there is a possibility to do the paced breathing exercise and gain bonuses while practicing some mindfulness.

![Figure 2.34 – Level 3 with end of level](image)

1.5: Play level 4 (grey text indicates that the same feature is available in the previous level.)
This level introduces attention and awareness to the game. These are important concepts in mindfulness. There are visual objects to pay attention to at the same time as the player has to count how many platforms he or she jumps on.

- The player character has its start position to the left.
- The player jumps from platform to platform to get from start to finish.
- In between and on platforms the player can collect coins.
- If the player misses a platform or fall off a platform the character falls out of the screen. Thereafter the character comes falling from the top of the screen again and the player must see to that the character lands on a platform to avoid dying immediately again.
- The platforms scroll in the same speed as the player moves.
- The player is instructed to count the how many platforms the character jumps on.
• Rabbits are placed randomly in the background. A dinosaur will also appear.

• During the game a random number of ducks are flying over the screen. A maximum of ten ducks are set as it is the intention that the player shall pay attention to them and remember how many he or she saw.

• The clouds will get brighter and darker in the same pace as the player’s calibrated breathing pace and the player can therefore practice paced breathing while playing and thus hopefully perform better.

• When the character has reach the end of the level an ‘End of the level’ screen is displayed with instruction of what to do in the next level.

• At the ‘End of level’ screen the player answers to two questions:
  o How many times did you successfully jump on to a platform?
  o How many ducks did you see?

Correct answers will generate bonuses. The correct answers are revealed by clicking on the ‘get answer’ button (Figure 2.35).

• At the ‘End of level’ screen there is a possibility to do the paced breathing exercise and gain bonuses while practicing some mindfulness.
1.6: Play level 5 (grey text indicates that the same feature is available in the previous level.)
This level is also about awareness, but this time the player must pay attention to sounds (i.e. bird tweets). The player must hit the space-key whenever he or she hears the tweeting at the same time as the character must be prevented from falling off the platforms.

Previous research has shown that increased levels of mindfulness are associated with improvements in various types of attention (Hodgins, Adair, 2010). One important characteristic of increased levels of mindfulness is the ability to disengage attention from objects in the foreground more easily. Therefore, while playing the game the player is asked to additionally try and listen for a particular sound and test how
quickly and if they recognise the sound: the more mindful they are, the faster they should be able to hear the sound (Lutz et al., 2009).

- The player character has its start position to the left.
- The player jumps from platform to platform to get from start to finish.
- In between and on platforms the player can collect coins.
- If the player misses a platform or fall off a platform the character falls out of the screen. Thereafter the character comes falling from the top of the screen again and the player must see to that the character lands on a platform to avoid dying immediately again.
- The platforms scroll in the same speed as the player moves.
- The clouds will get brighter and darker in the same pace as the player’s calibrated breathing pace and the player can therefore practice paced breathing while playing and thus hopefully perform better.
- When the character has reach the end of the level an ‘End of the level’ screen is displayed with instruction of what to do in the next level.
- At the ‘End of level’ screen the player get to know how many percentage of the sounds he or she heard and marked. The percentage will generate new bonuses i.e. points (Figure 2.36).
- At the ‘End of level’ screen there is a possibility to do the paced breathing exercise and gain bonuses while practicing some mindfulness.

![Figure 2.36 – Level 5 with start and end of level](image-url)
1.7: Play level 6 (grey text indicates that the same feature is available in the previous level.)
This level is a combination of level four and five. This time it is the dancing moles that are important and should be noticed.

- The player character has its start position to the left.
- The player jumps from platform to platform to get from start to finish.
- In between and on platforms the player can collect coins.
- If the player misses a platform or fall off a platform the character falls out of the screen. Thereafter the character comes falling from the top of the screen again and the player must see to that the character lands on a platform to avoid dying immediately again.
- The game scrolls faster and faster and the maximum speed is set to the speed from level 2 when the character died the third time.
- During the game rabbits are placed randomly in the background. A dinosaur will also appear.
- During the game a random number of ducks are flying over the screen.
- Dancing moles are shown in the lower half of the screen. A maximum of ten moles are set as it is the intention that the player shall pay attention to them and actually remember how many he or she saw.
- The clouds will get brighter and darker in the same pace as the player’s calibrated breathing pace and the player can therefore practice paced breathing while playing and thus hopefully perform better.
- When the character has reach the end of the level an ‘End of the level’ screen is displayed with instruction of what to do in the next level.
- At the ‘End of level’ screen the player answers a question:
  - How many moles did you see?
Corrects answers will generate bonuses. The correct answers are revealed by clicking on the ‘get answer’ button (Figure 2.37).
- At the ‘End of level’ screen there is a possibility to do the paced breathing exercise and gain bonuses while practicing some mindfulness.

![Figure 2.37 – Level 6 with end of level](image)
1.8 Fill in questionnaire

After the last level a questionnaire is shown (Figure 2.38). The questionnaire consists of three statements. The user fills in the questionnaire according to his experiences of the session. The first statement is always the same i.e. “I did the exercise as well as I could”. This statement is for the researchers to be able to determine if the results is valid for use in further analysis. The two other statements are randomly selected from the Toronto mindfulness scale (Lau et al., 2006). The scale consists of two parts, i.e. statements related to curiosity (six statements) and statements related to decentring (seven statements) (Appendix H). One statement from each part is randomly picked to be shown in the questionnaire. The user therefore seldom gets the same questionnaire after playing the game repeatedly.

When the player clicks ‘Save’ the questionnaire is closed, the answers are saved and the player goes to the end screen (Figure 2.39).
**Use case 2: Retrieve data**
All data is saved in a SQL database. The application uses three database tables. One main table that contains data from activities done in the application, one table that collects what statements has been shown in the questionnaire and the user input for each question and one table collection timestamp from the paced breathing exercise. For the user to retrieve the data from the database he or she can export the raw data to a csv-file and then into SPSS or directly into Excel for analysis.

2.5.4.1.1 Flow
The centre of the game is the paced breathing exercise that is intended as the tool for reaching higher mindfulness throughout the game. The flow chart in Figure 2.27 shows how the different screens and levels are connected. After level two there are always opportunities to do the paced breathing exercise.

![Flow through the game](image-url)

*Figure 2.40 – Flow through the game*
2.5.4.2 Implementation

The game is developed in Flash, Actionscript 3. To facilitate the game development a game-making library called Flixel is used (flixel.org). Flixel is written in Actionscript 3 and free to use. In the next section an overview is provided in form of a class diagram (Figure 2.41).

2.5.4.2.1 Class diagram

![Figure 2.41 – Class diagram]

2.5.4.3 Sound

The sound is retrieved from www.freesound.org. For the sound effects in Level 5 and 6 “bird_tweet.aif” by Tigersound (http://www.freesound.org/search/?q=bird_tweet) is used.

2.5.4.4 Logging

For the game to be of research value, it is critical that the data from the game play is collected. The necessary data is described per level in Table 2.9. The data is logged in an SQL database. The table is available in Appendix I.

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
<th>Logging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>The platforms move when the player moves, the player jumps on the platforms and collects coins.</td>
<td>Number of treats/coins collected, number of treats/coins that were in the level, points total, time in level (should always be the same in this tutorial),</td>
</tr>
<tr>
<td>Level 2</td>
<td>The game gets slowly faster and faster, but at some point, the player falls off.</td>
<td>Number of treats/coins collected, points total, time in level,</td>
</tr>
<tr>
<td>Retreat</td>
<td>Breathing exercise</td>
<td>Start and stop time.</td>
</tr>
<tr>
<td>Level 3</td>
<td>Same as level 2, but the speed when the player falls off in level 2 will be the</td>
<td>Number of treats/coins collected, number of treats/coins that were in the level, points total,</td>
</tr>
<tr>
<td>Level 4</td>
<td>Before the game starts the player is told to count the number of platforms he or she jumps on. The platforms behave as in level 3, but objects, not part of the game play, moves across the screen. After the level the answers to the tasks are presented, that is the number of platform the player did jump on. There is also a question if the player saw the objects.</td>
<td>Number of treats/coins collected, number of treats/coins that were in the level, points total, time in level, Plus: answer to the number of platforms and answer to question if they’ve noticed the object</td>
</tr>
<tr>
<td>Level 5</td>
<td>Before the level starts the player is told to listen for a specific sound and push the key ‘B’ when he or she hears it to get a bonus. The platforms behave as in level 3 and in addition the player has to react on sounds.</td>
<td>Number of treats/coins collected, number of treats/coins that were in the level, points total, time in level, Plus: number of correct and incorrect responses to the sounds and number of sounds that were presented</td>
</tr>
<tr>
<td>Level 6</td>
<td>A combination of level 4 and 5</td>
<td>Number of treats/coins collected, number of treats/coins that were in the level, points total, time in level, Plus: number of correct and incorrect responses to the sounds and number of sounds that were presented Plus: answer to the number of platforms and answer to question if they’ve noticed the object</td>
</tr>
</tbody>
</table>

Table 2.9 - Overview of what data must be logged.

2.5.5 Evaluation

The test process can be divided into three parts:

1. Functionality testing (does the prototype work as it should).
2. Acceptance test (is the product owner satisfied with the application)
3. Usability and play testing (how the game is used and how the user experience the game)

The functionality has been tested against the requirements and acceptance tests have been conducted with two partners. According to the results from the acceptance tests the requirements have been adjusted, updated and implemented. The latest requirements are shown in Table 2.8. The partners have been involved in the development process from the beginning.

The usability testing was done with four users. The usability and play test is only done with the purpose to get input to the design. The game will be part of the learning intervention that is the outcome of the xDelia project. No lab studies have been definitely planned to evaluate the usefulness of the game, although short test sessions with minimal samples can be scheduled before the final release to examine whether the short techniques indeed influence gameplay on the platform game. Essentially, such a study would comprise of approximately 10-20 students play testing the game.

In the usability and play tests done the researcher acted as a test leader. The players sat one by one in a room and the only instruction they got was that the game was about mindfulness. Before they started to play they had to fill in a questionnaire with demographical data and a question about how used they are to play platform games and if they knew about mindfulness. The player played the game while the researcher observed and took note of how the player acted. After the play session the players answered the GEQ and the SUS questionnaires (Appendix C and D).
Afterwards the player and the researcher discussed the game. What was good and what could be improved. The players were also asked if they got curious about mindfulness after having been playing the game. Three out of four players said they would like to know more about mindfulness after they have played the game. The players only played the game once.

Two of the players were very inexperienced gamers, while two of the player plays computer games several hours a day. Only one of the players stated to be experienced in the platform genre, which was confirmed by the final score.

All of the users did find the game very easy to use and no one had questions about how to play. The only difficulty was the paced breathing exercise, it was hard to understand how to use the key to make the circle shrink and expand. The breathing exercise has been redone and tested separately, and is in the present version easier to understand. It is the improved version that is described in this document.

All of the players, also the experienced one, found the game hard. The experienced user managed to finish all levels. To count the platforms at the same time as you need to keep track ducks or moles was experienced as extremely difficult especially as the platforms are so many. It is easy to lose track. It might be better to tell the user to follow the breathing patterns visualized by the clouds and instead count the breaths. But it has to be tested to see what has the best effect. As it is now the breathing and mindfulness was experienced to get out of focus.

All of the users experienced level 3 and level 6 as very stressful and it was hard for all of them to keep up. When it comes to how fun the game is there was different opinions. Two of the users thought it was quite fun. The most inexperienced player and the most experienced player found the game fun. It seems like the reason was that the inexperienced user concentrated on jumping and getting coins and to succeed to do that was fun enough. The experienced user scored much better than the others and tried to take all of the coins which appeared to motivate him. The two other players lost a little bit of interest when they lost all their points in almost every level.

In the discussions there were suggestions of how to improve the game. For example the two gamers wanted to have some power-ups, some enemies to avoid, other ways to collect coins e.g. look for them in barrels or other containers, to make the player produce coins by doing something like hitting the head in special platforms and have different kinds of platforms with special behaviours and attributes.

### 2.5.6 Summary

One of the purposes to develop the game was to make a game that could act as an introduction to the xDelia mindfulness course. The sample in the test was small, but since 75% answered that they were curious about mindfulness after they have played Mindformer shows that the game might fulfil its purposes. But to be able to say something for certain more studies have to be performed.

The user found the game easy to use but hard to master. The choice to use the platformer genre was among other things motivated by that a platform game is to a high degree self-explaining and easy to make stressful. Both statements were confirmed in the test.

To make the game more fun to play even for gamers some additional features should be added, e.g. power-ups and different ways to obtain more coins.

In conclusion it can be said that despite the small number of participants in the test there seems to be possibilities for the game to fulfil its purposes and it will be interesting to see what results we can get from a study where the participants use the game for several weeks.
## 2.6 xDelia Mind Manager

### 2.6.1 Prototype Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>xDelia Mind Manager</td>
</tr>
<tr>
<td>Version</td>
<td>1</td>
</tr>
<tr>
<td>Status</td>
<td>Developed</td>
</tr>
<tr>
<td>Target User Group</td>
<td>Investors</td>
</tr>
<tr>
<td>Prototype User Group</td>
<td>Internal + students</td>
</tr>
<tr>
<td>Purpose</td>
<td>To be used as a tool for practicing mindfulness exercises and to follow up training and improvements.</td>
</tr>
<tr>
<td>Description</td>
<td>A toolset which gives the user opportunity to practice paced breathing or meditation. The user can set alarm and get a notice when it is time for a mindfulness exercise.</td>
</tr>
<tr>
<td>Transfer</td>
<td>Yet unknown</td>
</tr>
<tr>
<td>Context of Use</td>
<td>Within the mindfulness course which is part of the learning intervention.</td>
</tr>
<tr>
<td>Appearance</td>
<td>Set of tools (screens)</td>
</tr>
<tr>
<td></td>
<td>1. Set time of notices</td>
</tr>
<tr>
<td></td>
<td>2. Practice paced breathing</td>
</tr>
<tr>
<td></td>
<td>3. Timer for meditation</td>
</tr>
<tr>
<td></td>
<td>4. Questionnaire to measure progress.</td>
</tr>
<tr>
<td></td>
<td>5. Statistics</td>
</tr>
<tr>
<td>Inputs (inc. sensors)</td>
<td>Mouse</td>
</tr>
<tr>
<td>Guidance</td>
<td>Instructions are incorporated into the tool</td>
</tr>
<tr>
<td>Gameplay/ Challenges</td>
<td>N/A</td>
</tr>
<tr>
<td>Logging devices</td>
<td>Logging of when and how much the user has practiced mindfulness and self-measurements of the experience of the training.</td>
</tr>
<tr>
<td>Platform</td>
<td>PC and possibly smart phones</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Web based application</td>
</tr>
<tr>
<td>Testing</td>
<td>Testing by students</td>
</tr>
</tbody>
</table>
2.6.2 Introduction

The xDelia Mind Manager is a software application that puts together several components with separate functionality. The application will be an integrated part of the xDelia mindfulness course, the xDelia Mindful Trading Training (http://research.saxobank.com/MFCourse/traders/startPage.html).

The mindfulness application consist of six parts:

1. **Mindfulness bell.** A bell rings either randomly or at specified times to signal time for conducting breathing exercises or other meditation exercises.

2. **Planner.** In the planner the user can plan his or her meditation sessions. The user can define how many times a day the bell should ring and within which time frame (e.g., only ring during office hours). The bell then rings randomly during that time frame. It is also possible to set a fixed time for the bell to ring.

3. **Meditation timer.** The user can perform a meditation here (i.e., set a begin and end bell, with a specified time interval of 1,3,5 or 10 minutes, and meditate in between the bells).

4. **Paced breathing** application. The user can specify the number of minutes (1, 3, 5, 10) he or she wants to perform the breathing exercise. The instruction is to perform inhaling/exhaling in according with the shrinking/expanding of a circle. These objects are moving at the pace of the average inhaling/exhaling of shrinking/expanding from the calibration period.

5. **Questionnaire.** The user has to relate to some statements after each mindfulness training session. The statements monitor how the training was perceived by the user. The answers act as a measurement of the user’s progress.

6. **Statistics.** The user’s use of the application is logged and the data is shown in a diagram to make it possible for the user to monitor his or her progress. The statistics also show the average result from the questionnaire.

2.6.2.1 Purpose and Research Questions

The rationale to develop this application is to have a set of tools available that facilitate the training of mindfulness skills. For the day trader mindfulness training, several tools are needed, for example, the tool set described in this chapter and the mindfulness game described in the previous chapter.

The main goal of the training intervention (i.e., mindfulness training for day traders) is to teach a set of skills that help regulate emotions. One powerful element of mindfulness trainings in this respect is the ability to recognize emotions at an early stage, and using the breath as an anchor to return in stressful periods. This requires practice.

For this, we need a set of tools and exercises. Contrary to regular mindfulness trainings, our training course for day traders is entirely online. The xDelia Mind Manager plays a major role in this online course. One of the research questions would therefore be:

- Can users obtain a higher level of mindfulness skills and emotion regulation skills via an online training?
- Does the xDelia Mind Manager add value and support to the mindfulness training?

The training intervention will be tested in both lab settings (i.e., students at EUR) and in the population of investors (i.e., day traders at Saxo Bank). In this training, users will practice intensively with the xDelia Mind Manager, as breathing exercises and meditations are the foundation of any mindfulness training. To examine whether our training (and thus these tools) are effective in their purpose, people who follow the training will be compared on pre- and post-measurements of trait mindfulness and emotion regulation, and compared to a control group who did not use the training (and thus, these tools).
2.6.3 Description of Prototype

The application is developed in an iterative process with frequent meetings with developers and other partners. The requirements have been established during the development process (Table 2.10). The requirements are held on a rather high level as the purpose of the requirements was to support the communication between developers and non-developers in the team. During the meetings the prototype was demonstrated, tried out and discussed, and changes were made in line with the discussions.

All of the requirements come from a use perspective. The technical solutions are derived from the user requirements and decided on during the meetings.

Some requirements shown in Table 2.10 are not implemented. Those requirements would have been nice to have, but is not essential to the functionality and use of the prototype. R2, R6 and R16 are therefore left for future iterations.

<table>
<thead>
<tr>
<th>No</th>
<th>Prioritization</th>
<th>Description</th>
<th>Type</th>
<th>Implemented</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>Preferred</td>
<td>The alarm should not be dependent of only sound</td>
<td>Mindfulness bell</td>
<td>yes</td>
</tr>
<tr>
<td>R2</td>
<td>Optional</td>
<td>The user can choose what bell to use and the volume of the bell</td>
<td>Mindfulness bell</td>
<td>no</td>
</tr>
<tr>
<td>R3</td>
<td>Preferred</td>
<td>The user must be able to set the exact time when the bell is to ring.</td>
<td>Planner</td>
<td>yes</td>
</tr>
<tr>
<td>R4</td>
<td>Essential</td>
<td>The user must be able to set a time frame of when the bell randomly should ring. For example the time frame can be set to ring between 10 to 12 o’clock.</td>
<td>Planner</td>
<td>yes</td>
</tr>
<tr>
<td>R5</td>
<td>Essential</td>
<td>The user can choose for how long the meditation shall last.</td>
<td>Meditation timer</td>
<td>yes</td>
</tr>
<tr>
<td>R6</td>
<td>Optional</td>
<td>The user can choose background sounds for his or her meditations for example wave sound.</td>
<td>Meditation timer</td>
<td>no</td>
</tr>
<tr>
<td>R7</td>
<td>Essential</td>
<td>Before meditation the breathing has to be calibrated. The calibration can be done by pressing a button when breathing in and release it when breathing out.</td>
<td>Paced Breathing</td>
<td>yes</td>
</tr>
<tr>
<td>R8</td>
<td>Essential</td>
<td>The user can choose for how long the meditation shall last.</td>
<td>Paced Breathing</td>
<td>yes</td>
</tr>
<tr>
<td>R9</td>
<td>Essential</td>
<td>It must be possible to tweak the breathing pattern after the calibration. Functionality to adjust the circle’s perimeter to better convey to the user’s breathing patterns (bio breathing).</td>
<td>Paced Breathing</td>
<td>yes</td>
</tr>
<tr>
<td>R10</td>
<td>Essential</td>
<td>The user must state how well he or she has performed the exercises.</td>
<td>Questionnaire</td>
<td>yes</td>
</tr>
<tr>
<td>R11</td>
<td>Essential</td>
<td>The user must relate to statements from the Toronto Mindfulness scale (Lau et al., 2006)</td>
<td>Questionnaire</td>
<td>yes</td>
</tr>
<tr>
<td>R12</td>
<td>Essential</td>
<td>The total amount of minutes per a user has trained mindfulness during the last week must be shown in a chart</td>
<td>Statistics</td>
<td>yes</td>
</tr>
<tr>
<td>R13</td>
<td>Essential</td>
<td>The total amount of minutes a user has trained mindfulness during the last month must be shown in a chart</td>
<td>Statistics</td>
<td>yes</td>
</tr>
<tr>
<td>R14</td>
<td>Essential</td>
<td>The score from the questionnaire during the last week must be shown in a chart</td>
<td>Statistics</td>
<td>yes</td>
</tr>
<tr>
<td>R15</td>
<td>Essential</td>
<td>The score from the questionnaire during the</td>
<td>Statistics</td>
<td>yes</td>
</tr>
</tbody>
</table>
last month must be shown in a chart

| R16 | Optional | It must be possible to log the heart rate before and after the meditation to give the user an overview of his/her progress. (Count the pulse for 10 seconds, put in the number and calculate the rate of a minute.) | Statistics | no |
| R17 | Essential | The following data must be logged:
- Name/subject id
- Time of meditation or paced breathing
- Duration of meditation or paced breathing
- Statements in questionnaire
- Answers in questionnaire | Logging | yes |
| R18 | Essential | It must be easy for the researcher to get the logged data. | Logging | yes |
| R19 | Essential | The application must be available to all users independent of if the use PC or Mac. | Technical | yes |
| R20 | Optional | It should be possible to use a smart phone to run the application. | Technical | partly |
| R21 | Essential | The user must be able to use the application without reading a manual. | General | yes |

Table 2.10 - Final list of requirements

### 2.6.3.1 Use Cases

This section will describe a number of use cases (Figure 2.42). The relationships between the requirements and the use cases are shown in Table 2.11.

![Use case diagram](image)

**Figure 2.42 – Use case diagram**

<table>
<thead>
<tr>
<th>Use case no</th>
<th>Use case name</th>
<th>Implemented requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start the application</td>
<td>R1</td>
</tr>
<tr>
<td>2</td>
<td>Plan for practicing mindfulness</td>
<td>R1, R3, R4</td>
</tr>
</tbody>
</table>
### Table 2.11 – Relationship between requirements and use cases

<table>
<thead>
<tr>
<th>Use case</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Get notified</td>
</tr>
<tr>
<td>4</td>
<td>Meditate</td>
</tr>
<tr>
<td>5</td>
<td>Practice paced breathing</td>
</tr>
<tr>
<td>6</td>
<td>Fill in questionnaire</td>
</tr>
<tr>
<td>7</td>
<td>View statistics</td>
</tr>
<tr>
<td>8</td>
<td>Retrieve data</td>
</tr>
</tbody>
</table>

#### Use case 1: Start the application
When starting the application the first screen makes the user enter his or her name or, in case of participating in a study, the subject id must be registered. Without entering a name or id the user is not able to see the statistics of his or her mindfulness exercises. The name/id is saved locally. When the user clicks on next, the main menu will show up.

![Start screen](image)

*Figure 2.43 – Start screen*

#### Use case 2: Plan for practicing mindfulness
The user opens the app and clicks on the Planner button. The user is presented to an interface where he or she can put in how many times a day the bell must ring and in what time frames it is allowed to ring. The user can also choose to set the alarm bell to ring on a fixed time if that is preferred. The user can reset all the values by clicking ‘Reset’ or reset individual values by clicking on the recycle bin next to the time boxes. The user can also check a summary of the saved alarms by clicking ‘Show’. There is also a possibility to hear the alarm signal by clicking ‘Sound’.)
Use case 3: Get notified
To be notified the user must first have used the planner (see Use case 2). The notifications originate from the main menu and this screen (Figure 2.45) must be open to be able to get a notification. The meditation bell consist of both sound (see Use case 2) and visual notification i.e. a small popup-window shown in the lower right corner of the screen.

Use case 4: Meditate
The mindfulness bell rings. The user chooses to disregard the paced breathing exercise and do some meditation on his or her own. The user chooses for how many minutes and starts meditating. There is an option to select 1, 3, 5 and 10 minutes (Figure 2.46), but if an arbitrary time is preferred that is also possible by using the input field instead of the buttons. The user clicks on the start button when he or she is ready (Figure 2.47). There is a time delay of five seconds, for the user to arrange to sit comfortable and when the
session starts there is a signal announcing the start. When the time is passed the same signal rings again. The progress bar makes it is possible for the user to monitor how much time has passed.

![Figure 2.46 Screen to choose for how long to meditate](image)

**Figure 2.46 Screen to choose for how long to meditate**

![Figure 2.47 – Meditation timer](image)

**Figure 2.47 – Meditation timer**

**Use case 5: Practice paced breathing**

The mindfulness bell rings. The user chooses to do the paced breathing exercise. The user chooses for how long he or she wants to do the exercise (Figure 2.48). The user is told to inhale while he or she is pressing the left mouse button and to release when exhaling (Figure 2.49). A red circle will show and increase its radius until the button is released. The pace of the paced breathing is set and the red circle will shrink again (Figure 2.50). After the calibration the user clicks on the start button to start the exercise (Figure 2.51). A blue circle will show and the circle will shrink and expand according to the calibrated breathing pace (Figure 2.52). The user can adjust the pace by clicking on the buttons at the bottom of the screen (Figure 2.52).
Figure 2.48 – Screen to choose for how long to do the breathing exercise
Figure 2.49 Screen with instructions of how to calibrate the paced breathing exercise

Figure 2.50 – Calibration of paced breathing exercise
Use case 6 Fill in questionnaire
After a meditation or paced breathing session a questionnaire is automatically shown. The questionnaire consists of three statements. The user fills in the questionnaire according to his experiences of the session. The first statement is always the same i.e. “I did the exercise as well as I could”. This statement is for the researchers to be able to determine if the results is valid for use in further analysis. The two other statements are randomly selected from the Toronto mindfulness scale (Lau et al., 2006). The scale consists of two parts, i.e. statements related to curiosity (six statements) and statements related to decentring (seven
statements) (Appendix H). One statement from each part is randomly picked to be shown in the questionnaire. The user therefore seldom gets the same questionnaire after a mindfulness exercise. When the user clicks ‘Save’ the questionnaire is closed, the answers are saved and the user comes back to the main menu.

![Questionnaire](image1)

**Figure 2.53 Questionnaire**

**Use case 7 View statistics**

The user wants to see how frequently and for how long he or she has practiced mindfulness and if there has been any positive progress. The user chooses to view the statistics and in the statistic screen the user can choose between viewing how long he or she has practiced mindfulness during the past week (Figure 2.54) or month. The user can also choose to monitor how his or her mindfulness state has changed during the last week or month, i.e. monitor the mindfulness progression.

![Statistics](image2)

**Figure 2.54 – Statistics showing how many minutes a user has spent on mindfulness exercises the past week.**
Use case 8: Retrieve data

All data is saved in a SQL database. The application uses two database tables. One main table that contains data from activities done in the application and one table that collects what statements has been shown in the questionnaire and the user input for each question. For the user to retrieve the data from the database he or she can export the raw data to a csv-file and then into SPSS or directly into Excel for analysis.

2.6.3.1.1 Overview of how the screens

Figure 2.55 shows how the different screens are related. Solid lines mean navigation forward and dashed lines are navigations back to main menu.

![Flow chart](image)

Figure 2.55 – Flow chart

2.6.4 Implementation

The prototype is built in HTML5. The reason for choosing HTML5 was that it should be possible for everyone to use the application independent of what kind of operating system he or she uses (Requirement 19 in Table 2.10). It would also be feasible if the application could be used on smart phones (Requirement 20 in Table 2.10) which would be the result if using HTML5. In that way the application would be even
more useful for the investors. But the requirement (R1) that the alarm should not be dependent of sound if
the user has chosen to turn of the sound or the computer does not have a sound card lead to the decision to
mainly target Google Chrome in this first iteration as it is the only browser that supports notifications.
Theoretically the application will work on other browsers as well with the only exception that the visual
notification service will not work. The decision to mainly target Google Chrome resulted in that the
prototype is only tested in Google Chrome. Modifications to get an acceptable visual appearance in other
browser as well are left to future work.
Figure 2.42 shows what files the prototype consist of and how they are related.

Figure 2.56 – Overview of files in prototype

The PHP files are divided into four different files mainly because it should be easy for anyone to
understand what query belong to what activities in the application. The style sheet (mfapp.css) is used in all
HTML files. RGraph.js (http://www.rgraph.net/) is a third party component used for the charts in the
statistic view.

Except for the external javascript RGraph.js all the javascript code is embedded into the HTML-files. This
is also due the conviction that it might be quicker to understand and modify the code if everything is in the
same file.
2.6.5 Evaluation

The test process can be divided into three parts:

4. Functionality testing (does the prototype work as it should),
5. Acceptance test (is the product owner satisfied with the application)
6. Usability testing (how the prototype is used and how the user experience the application)

The functionality has been tested against the requirements and acceptance tests have been conducted with two partners. According to the results from the acceptance tests the requirements have been adjusted, updated and implemented. The latest requirements are shown in Table 2.10.

The usability testing was done with four users according to the protocol in Appendix F. The usability test is only done with the purpose to get input to the design. The usefulness of the program is evaluated in studies at EUR and Saxo Bank.

In the usability tests the researcher acted as a test leader and gave the user some scenarios to accomplish. The time for the user to complete the task was recorded and the test leader was observing what the user was doing and made notes of what was going on. After each scenario the user and the test leader discuss the user’s experience. Before the discussion the user also had to fill in the questionnaire in Appendix G (System Usability Scale (Brooke, 1996)). The test approach is similar to the UTUM (UIQ Technology Usability Metrics) test process (Winter at a., 2007) (a test case can be seen at http://www.youtube.com/watch?v=5iIRIVwgeo). The results are summarized in Table 2.12.

<table>
<thead>
<tr>
<th>No</th>
<th>Test leader asks questions</th>
<th>…and take notes (What does user do?)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Have you heard of mindfulness? Have you practice mindfulness or other type of meditation?</td>
<td>One of the five users had practiced mindfulness before. Neither of the other users knew anything about mindfulness.</td>
</tr>
<tr>
<td>2</td>
<td>Try out the breathing exercise</td>
<td>All the users had problems to understand how to make the calibration work. The test leader had to help out.</td>
</tr>
<tr>
<td>3</td>
<td>What did you think of the breathing exercise?</td>
<td>When they got the breathing exercise to work they found it quite good, except for the experienced user that did not think it added anything to the mindfulness training. The experienced user experienced it as the visual pattern (expanding and shrinking circle) draw the attention from the breathing itself.</td>
</tr>
<tr>
<td>4</td>
<td>Try out the mediation timer</td>
<td>All the users managed the meditation timer without trouble. As the users had performed the paced breathing just before this exercise they all practiced the breathing again. The experienced user closed the eyes while the other users stared at the progress bar.</td>
</tr>
<tr>
<td>5</td>
<td>What did you think of the meditation planner?</td>
<td>All the user found the timer ok.</td>
</tr>
<tr>
<td>6</td>
<td>Try out the statistics. What does it tell you?</td>
<td>The user didn’t have any trouble to understand the charts except that they did not do the questionnaire, why they could not see anything on those charts, which was confusing.</td>
</tr>
</tbody>
</table>
7 Is the shown statistics what you would expect/like or have use of? They did not have any opinion of what to get statistics of. They found that what there is was ok. The experienced user was unsure if the training would benefit from the feature.

8 Planner. Set the planner to alert three times a day. All users manage fairly quickly.

9 Set the time to alert twice a day. Lunch, after work 2h. All users manage fairly quickly.

10 What did you think of the planner? How was it to use? The experienced user thought that the randomized time periods are something that would not be used. The experienced user preferred to use the fixed time options. All the users though that it was fairly easy to use when you got the hang of it.

11 What did you think of the app design? None of the users had any comments on the design more than it was rather easy to us it.

Table 2.12 Summary of the outcome from the user tests.

From the user test it was revealed that the calibration of the breathing pace was hard to understand (No 2 in Table 2.12). It resulted in a re-make of the calibration and better instructions. The experienced user also questioned if the breathing exercise add something to the mindfulness experience or if it just draw attention from the intended activity (No 3 in Table 2.12). That is something that should be explored in the studies at EUR and Saxo Bank. Another issue that is important to look into is if the progress bar in the meditation timer influences the meditation positively or negatively (No 4 in Table 2.12). The outcome of the test also indicates that the usefulness of the statistics (No 6 in Table 2.12) and planner (No 10 in Table 2.12) must be investigated further.

The System Usability Scale was also applied (Brooke, 1996). The questionnaire confirmed that the overall design was rather sufficient (Figure 2.57). The only issue was statement one “I think that I would like to use this program frequently”. As three of the participants didn’t plan to practice mindfulness and the fourth participant was not sure of the usefulness of the application it is hard to tell anything concerning this matter. How useful the application is and how frequent the users are motivated to use it will be explored in upcoming studies. The xDelia Mind Manager is planned to be used in a study involving day traders at Saxo Bank in April- May 2012.
2.6.5.1 Summary

The xDelia Mind Manager is a toolset for training mindfulness and it is developed in cooperation with partners from BTH, EUR and OU. The toolset is going to be used as a support tool within the xDelia mindfulness course (http://research.saxobank.com/MFCourse/traders/startPage.html). The design contains basic mindfulness exercises as well as reminder functions and planning tool. The tool set also embrace functionality to be able to follow up on individual progress, e.g. statistics. This functionality is in somewhat diametric to the intentions in the concept of mindfulness where the present should be in focus. What this functionality adds to the concept and the xDelia mindfulness course is a systematic approach that is believed to appeal to the target group, e.g. investors. Whether this approach is useful will be evaluated in future studies.

<table>
<thead>
<tr>
<th></th>
<th>Strongly disagree</th>
<th>Strongly agree</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>2.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>3.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>4.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>5.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>6.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>7.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>8.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>9.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>10.</td>
<td>Strongly disagree</td>
<td>Strongly agree</td>
</tr>
</tbody>
</table>

Figure 2.57 – Result from SUS questionnaire. The numbers in the grey cells refer to how the four participants answered the statements.
3 WP3 Concepts and Games

3.1 Overview of the Design Process in WP3

The process to design and develop prototypes for WP 3 started already Y2 (see D 14-4.2). The process can be divided into two segments:

1. Explorative prototyping with the purpose to gain knowledge and common understanding of how the overall learning intervention should be designed and developed (the explorative prototypes are described in D14.4.2 and in section 3.3-3.5 in this document).
2. The design and development of the learning intervention itself.

The design process consists of iterative brainstorming meetings, evaluation sessions and discussions with WP3 partners in a participatory design manner. The overall design process is described in Deliverable D17-4.1. The input to the design process in WP3 involved analysis of existing games related to financial capability. Another essential input came from existing financial capability frameworks (FSA/BSA, 2004 and NSG-FC, 2009) and from the UPPS Impulsive Behaviour Scale (UPPS stands for urgency, premeditation, perseverance, sensation-seeking) (Cyders & Smith, 2009).

Each prototype cycle (duration usually one month) consisted of a specification phase where the development team worked closely with representatives from WP3. Initially WP3 produces a prototype rationale taking into account not only implementation, but also play testing, evaluation and experiments. The rationale is then worked up into prototype specifications. Video conferences, Google documents and Google wave have been used as communication channels. An iterative development approach was used where the prototype were kept runnable at all times and in the middle of the cycle the status of the prototype was demonstrated and discussed with representatives of WP3, and feedback is given. The demonstration makes it possible to clarify details and misunderstandings and to take into account aspects that were not highly prioritized from the beginning.

The prototyping of the learning intervention takes a different approach with meetings every second day where implemented features are demonstrated and new ones are selected from the specification document. The frequent meetings ensures that the learning intervention not deviate from the WP3 intentions. As soon as the prototype is considered “good enough”, play testing take place and the prototypes can be tuned to better meet the requirements of WP3. Play testing is done by participants from both WP3 and WP4. There are two types of play testing: (1) play testing by the participant in WP3 and WP4 involved in the development process. This kind of test checks the latest updates while (2) play testing by participants not involved in the development process follows a play testing protocol (see D17-4.1).

During the past six month period we have developed several prototypes and concepts, i.e.

1. FinBoard game (board game)
2. FinBoard game for iPhone (prototype)
3. Banking game (concept)
4. Prototypes series of impulsivity implementing the go/nogo task (eight prototypes)
5. MINDswap on Facebook (prototype, will be wrapped up in a couple of months).

3.2 Relationship between Y1 and Y2 prototypes

The prototype development during Y1 (D14-4.2) was highly explorative. We produced a set of table top games (Y1:A), concepts (Y1:B) and micro games (Y1:C) (the tags refer to Figure 3.1). Several elements and lessons learnt have been taken care of in the prototypes produced in year two. The first games
developed year one where a set of table top games. One lesson learned (I) from this development process where that descriptive knowledge should not be in focus. More interesting is to develop games that foster awareness and how to manage inappropriate behaviour. Some elements of the table top game were considered worth developing further and can be found in the concept of the learning intervention (Y2:E), namely the notion of life quality from the Financial Card game and the collaboration and the negotiation with other players in the Evil Spouse game (a).

A first step towards managing inappropriate behaviour is to be aware of one’s behaviour and the concept of First Person Shopper (Y1:B) embraced that by trying to get the player to be aware of his or her desires. But when presenting the concept to the consortium it was decided that it was too early to go into the direction of consumer research (II) before for example exploring how to measure behaviours. The game play in FirstPerson Shopper included the ability to handle impulsivity and to be aware of one’s behaviour and those elements are taken care of in the new set of prototypes and thereby in the learning intervention (c).

The choice to take another direction lead to a decision to explore how to implement psychological tests in games, resulting in a set of small micro games implementing the Iowa gambling task (Y1:C) that measures risk behaviour. We learned from those prototypes that we have to, more systematically, explore the relationship between behaviour, game design and game play (III), which have guided us in the prototype development which led to the FinBoard (section 3.3 and the go/nogo task prototype series (section 3.5).

The first attempt was to build a board game (Y2:A) that we believed could act as a test bench for future prototypes by allowing quick incorporation of new aspects of game play. The concept was driven by the financial capability framework (FSA/BSA, 2004 and NSG-FC, 2009). It turned out not to be as useful as we had hoped but some ideas are preserved (f) in the learning intervention like temptations the player has to manage. Instead the concept was used to explore the technical possibilities with iPhone games (Y2:B) and the ability to do location based games. This development was driven by the idea that a realistic setting would promote transfer. The prototype taught us what can be done and how a mobile game can be incorporated into a web based game, but since WP3 is terminated the prospect to use mobile phones is not valid any more.

With the experience of the development of the FinBoard games as a base, a concept aimed at the overall learning intervention (Y2:E) called the Banking game (Y2:C) were developed. The banking game showed some potentials but the bank setting was not perceived as suitable when discussing it in WP3 (see section 3.4). But the basic ideas (e) in the Banking game got a new framing and is now incorporated into the concept of the learning intervention.

The lessoned learned (III) from the micro games year one made us aware of the necessity of systematically explore how behaviour, game design and game play relates. It resulted in a prototype series implementing a psychological test measuring impulsivity, the go/nogo task (Y2:D). The specification of the prototypes where driven by one of the financial capability determinants, namely impulsivity specified in the DoW. The series is designed in a way that explicitly addresses how the task can be incorporated into the learning intervention (IV).
3.3 FinBoard Game

As mentioned above the development and evaluation of the Micro games revealed the need to explore the relationship between the behaviour, the game design and the gameplay. From a brainstorming meeting where we started out from that position and added two financial capability framework (FSA/BSA, 2004 and NSG-FC, 2009) to guide the brainstorming the outcome was an idea to make a simple board game where is would be easy to add new features and test their bearing. The concrete result was the FinBoard game.

The FinBoard game exists in two versions, a table top game and an iPhone game. The iPhone game served the purpose of exploring the potentials of extending a web based game with a mobile application.

The FinBoard game is played on a board divided into different colours (Figure 3.2). The players toss a dice and walk around the board. When the player stops on a square he or she have to take a corresponding card. The colours have different meaning. Some are events and some are temptations or action the player can take. For example if the player enters beige squares something happens that make his or her life improve. A red square is a shopping mall and the card will challenge the player to concur different temptations. Other cards force the player to evaluate different offers or agreements. An orange square is a bus stop and the player has to decide if he or she wants to take the bus, walk or maybe spend some money on renting a car which will give the player certain advantages but yet costs some money.
Figure 3.2 – FinBoard game, table top version

Figure 3.3 – Screen shots of the iPhone version of the FinBoard game
3.3.1 Evaluation of the FinBoard game

The FinBoard was not found as useful as a test bench as expected. We experienced that the relationship between the implementation in the FinBoard game was too far away from a possible implementation in a computer game for us to say anything of the computer implementation as such. A fun and interesting feature in the FinBoard game might not be relevant or fun in a computer game as the engaging thing with the Board game was the interaction with co-players.

Instead we used the concept to try out the technology of location based games on mobile phones as mentioned above.

From play testing and evaluating the FinBoard, within the development team, some aspects were found to be valuable in a financial capability game.

- (Aspect 1) Possibility to social interaction, the players can collaborate, challenge or just talk.
- (Aspect 2) The notion of temptation, the players for example urge to buy something when approaching a shop
- (Aspect 3) The player has an opportunity to learn how to handle specific situations by using real life strategies like if I need a loan then I must learn about it and be able to calculate the interest rate to be able to make a grounded decision, or if the player has the behaviour that makes him or her urge for buying things then you had the possibility to avoid certain places on the board.
- (Aspect 4) The activities can be tweaked to be fun e.g. the cards are written in an amusing way. It proposed an unrealistic touch to the game, which made the players laugh and joke.

The outcome from the evaluation acted as an input to the next brainstorming session which resulted in the Banking game concept.

3.4 Banking Game

Except for the outcome of the experiences from the FinBoard game (Aspect 1-4) the financial capability frameworks (REF) were used as input to the opening brainstorming meeting. In addition the theme “Secure your financial future” was introduced.

In the Banking Game, the players will advance through the hierarchy of a bank, meaning they will get the opportunity to play the roles of several different bank employees. The purpose of the game is to assist different customers with their financial problems and thus learn about financial issues through the eyes of the customers (Aspect 3). By advancing on the career ladder the player get to deal with many different FinCap problems of different levels of difficulty and ambiguity, and together with more social aspects such as being a part of something bigger, e.g. a corporation, this may be a way to change the players’ attitude to financial capability issues. Another way to implement social elements is to create a common social multiplayer environment like a coffee room, allowing players to socialize with each other and give and ask for advice as well as challenging co-players (Aspect 1).

In the game, the player has to help customers make decisions in dealing with different financial issues which of course require the player to acquire some financial knowledge and skill. These advices are however not always in the customers best interest but rather beneficial to the bank. The reason for taking an “evil” approach in dealing with customers is to take the opposite approach and teach the players in identifying the tricks and strategies used by corporations.

So initial simplified concepts:

- Working at a bank
- Supporting customers
- Collaborate with and challenge co-players
- Starting from the bottom and working your way up the corporate ladder through:
3.4.1 Game play

When players play this game, they take on certain tasks for which they receive experience points. These experience points will then benefit the "level up"-system where sufficient experience points will present the player with the opportunity to get promoted, similar to a traditional "level up".

This process will however not be automated. Instead, a test regarding FinCap will be presented which represents a job interview or such. Successfully completing this test will result in the player receiving a promotion. There is also the possibility to bribe your way to new levels if, for example, you don’t have enough experience points, but you still must take the FinCap knowledge test. Because of this game concept of promotions we can make smooth transitions between the different areas of financial capability, such as loans, insurance, savings or investments.

In short there are:

- Experience points enable level ups or promotions
- Tests have to be successfully completed in order to level up/ get a promotion
- Promotion means more complex tasks and difficult customers
- Can introduce different areas of Fin Cap almost seamlessly

3.4.2 Game elements

The game is designed to be a generic and relatively open platform to which mini-games will be added. Mini-games are small, classical game concepts, dealing with some very specific financial learning objectives. The mini-games are optional but will be beneficial to the player for two important reasons. The first reason is that they will provide the player with the financial knowledge and skills needed in order to be able to pass the promotion tests. Secondly, successfully completing mini-games will provide the player with experience points needed for advancing to new levels. The benefit of having a generic platform is that later patches and expansions can contain additional mini-games, which means that the game can be developed gradually and iteratively.

From the start, most mini-games are locked. Unlocking mini-games require experience points and higher levels. This makes it possible to steer the players learning according to the level of financial knowledge. Once a mini-game is unlocked, the player can call the customers and request to repeat the mini-game for a possibility to improve the result.

Simplified, it results in:

- The game is a generic platform
- Mini-game is a small classical game concept dealing with a specific financial issue.
- Mini-games could be implemented gradually.
- Mini-games are optional but beneficial.
- The mini-games are unlocked as the player progresses through the game.

3.4.3 Mini-games

The Bank game is a generic platform to enable simple and continuous implementations of so-called mini-games, small stand-alone games, which generally focus on a specific learning outcome or knowledge
category. In this chapter, some of these mini-games are presented with a short description linking each and everyone to the main theme as well as to their respective learning outcome.

### 3.4.3.1 Memory

Just like in normal memory games you should find two matching targets. However, the matching here are different saving options with the approximately same amount of risk. The memory should be action-based and not time-based, since it might support a deeper coding of knowledge. Result condition is based on how many moves you have to do in order to complete the whole memory. With regards to the theme this could be a mini game where the player are in a meeting with clients and try to recommend savings/consumer rights/etc, but the avatar has forgotten this so the player needs to do this manually.

**Learning objectives**

- Memory training
- Concentration training
- Recognize groups of risk in savings
- Which person to turn to with a specific problem, e.g. contract problems with a cell phone or shoes that are worn out after three weeks (consumer rights)
- Can also be used to assess other aspects of FinCap when pairing if possible

### 3.4.3.2 Quiz

The idea of Quiz is to challenge the player's financial knowledge. This could be done in two slightly different approaches. The first one would be to simply present arbitrary FinCap questions and provide some alternatives or options. The other way would be to let the player solve problems or derive the most suitable approach to a specific problem. For example, the game could present different financial products or advice which the player then have to base an, as informed as possible, decision. There would therefore not be right or wrong choices per se, but rather varying degrees of good or bad decisions. The actual content of the game can deal with whatever topic is found to be the most relevant for the different game concepts (m24 vs. m36). There can also be different versions of the same game where the financial themes are the only difference. This game is in line with the general game theme since it is presented in such a way that customers come to you as an employee at the bank with questions and proposals which makes up the mini-game content.

**Learning objectives**

- Financial planning - Saving and spending
- Risk and return
- Consumer rights and sources of advice
- Understanding financial information

### 3.4.3.3 Falling items

The goal of the game is to control your avatar and collect different items. Every item has a different cost and the player must therefore spend budget money in order to receive the items. The items can be products, services or actions. Player who collects more objects within reserved budget is considered better and will thus receive more experience points.

**Learning objectives**

- Financial planning - saving, spending, budgeting
- Personal choices and the financial implications
• Budget for your daily spending

3.4.3.4 Game of Life
The players control customers’ life and guide them through all the different financial decisions they have to make. Life-changing events will come their way and they (you) have to deal with them as “well” as possible. This might include having children, getting fired, having the house burn down and so forth and the idea is then to manage financially.

The outcome will vary a lot depending on the different choices you make for the avatar and mini-game scores will depend on the financial success which then will be transformed into experience points. The strength of this game is that one gets to gaze forward into the far future and see the results of choices made at an early age. This is then meant to increase the player’s understanding of long term thinking and the importance of financial planning.

Learning objectives
• Financial planning
• Risk and return
• Financial implications of personal choices
• Financial information and advice

3.4.3.5 Jeopardy
This is an “employee-of-the-month” competition. The mini game is very similar to ordinary Jeopardy where the answers are given and the competitors have to formulate the question. The result will be posted on a high score list. The game will act as a self diagnostics of how much the player knows about the topics in the mini game, but it might also be an opportunity to learn more as an answer will be mapped with a question before another answer is posed. Three competitors compete in knowing the most about consumer rights, responsibilities and sources of advice. Two competitors are computer controlled (NPC), but the mini-game may also support multiplayer mode. The response times of the NPCs are random and so are the answers. A high score list will be available so that the player can compete with his or her friends.

Learning objectives
• Several knowledge categories

3.4.3.6 Argumentation
An alien customer is entering the bank wanting some money, and the bank employee has to discuss cultural and personal values surrounding money. The solution from a game perspective is to have a decision tree with different arguments and services that the bank employee offers. Depending on what path the discussion takes the player gets different amounts of experience points.

Learning objectives
• Financial planning

3.4.3.7 Sorting game
The Sorting game is about having a number of categories and being able to classify different kinds of stimuli into these. This is time-based and the goal is to finish as soon as possible, or to be able to do as
many as possible within a given time. Here, one could try to sort different kinds of loans in accordance with their interest rates or something else stated below. The players could, for example, accidentally lose all their papers on the floor which then get mixed up before an important meeting. They then have to sort them out by getting as many as possible in the correct piles with regards to risk factor, budget plans for people etc.

Learning objectives

- Risk and return
- Financial planning

3.4.3.8 The evil spouse

This game focuses on the everyday negotiation between two partners. One of the partners is trying to spend their collective money and the other is trying to save. At the end of each month they then have to successfully pay all their bills and other costs. The game forces the players to keep track of their costs and balance their account. The bank may also have financial counselling and this game is an activity that can only be done in a real counselling situation or in schools. The game is a table top game and the different pieces and the board itself are tagged to make it possible to get the results seamlessly into the overall game.

Learning objectives

- Financial implications of personal choices
- Balance your own financial experience and habits with those of your dependants, partners, family and friends

3.4.3.9 Local based game

The player can at a certain level choose to play with a friend. The story is that the bank employee has to go undercover and experience the customers’ real world to gain invaluable knowledge. One of the players has an iPhone with a map of the neighborhood. The map is populated with virtual items located somewhere in the town and you have a mission to buy a certain item, for example a car. You have to buy the item, but before you can do so you have to pull your finances together and investigate loans etc. The bank employee meets a lot of ‘evil’ salesmen and bank clerks etc. to experience how it is to be on the other side of the counter. The co-player stays at home and can for example provide the geo-player with information from the game. The goal is to get as good deal as possible and the players compete against other couples as the result is collected in a high score list.

Learning objectives

To teach the player what is needed to make educated decisions and where to find information.

3.4.4 Evaluation of the Banking Game

The concept was presented to WP3 partners for them to evaluate the validity of the game concept from a financial capability perspective. The outcome from the discussions where that the WP3 partners did not perceive the story line as favourable. The motivation was that young people associate financial matters too much with banking as it is, and it is something that should be avoided. Instead a focus on everyday life would be more sufficient for the target group as young people must see financial capability as a natural part of life and not something that should be left to the banks to handle.

The rest of the ideas from the concepts where found valid, that is

- Collaborate with and challenge co-players
• Experience points enable level ups
• Tests have to be successfully completed in order to level up
• Possibility to introduce different areas of Fin Cap almost seamlessly.
• The game is a generic platform
• Mini-game is a small classical game concept dealing with a specific financial issue.
• Mini-games could be implemented gradually.
• Mini-games are optional but beneficial.
• The mini-games are unlocked as the player progresses through the game.

The ideas acted as a base for the next concept, the overall learning intervention, the MINDswap (section 3.6). The notion of mini games where also kept even if the nature of the mini-games not necessarily corresponds to the presented mini-games described above.

3.5 Go/nogo prototype series

Young adults have recently been subject to attention due to lack of proper financial capabilities (FSA/BSA, 2004; NSG-FC, 2009). In order to help these people, xDELIA is investigating different technological implementations in order to train certain constructs which are believed to be essential for financial capabilities. One important construct seems to be how impulsive people are, see WP3 for detailed description why this conclusion was made. There are several views of impulsivity, but is by many understood as a multifaceted concept made up of distinct psychological constructs. One model sees impulsivity as composed of five different factors (Cyders and Smith, 2009):

• Sensation seeking, the tendency to seek out novel and thrilling experiences
• Lack of planning, the tendency to act without thinking
• Lack of perseverance, the inability to remain focused on a task
• Negative urgency, the tendency to act rashly in response to distress
• Positive urgency, the tendency to act rashly when experiencing extremely positive emotion

A commonly used self-report tool is the Barratt Impulsiveness Scale (BIS; Patton et al. 1995). It consists of the six first-order factors: Attention, focusing at a task at hand; Motor impulsiveness, acting on the spur of the moment; Self-control, planning and thinking carefully; Cognitive complexity, enjoy challenging mental tasks; Perseverance, a consistent life style; Cognitive instability, thought insertions and racing thoughts. These factors are also combined into second-order factors, where attention and cognitive instability are combined into attentional impulsiveness; motor impulsiveness and perseverance are combined into motor impulsiveness; self-control and cognitive complexity are combined into non-planning impulsiveness.

Many cognitive tasks such as stroop tasks, intelligence measures, implicit association test exist, but most of them require a lot of time, take a heavy load on attention levels and are tedious to go through. A first step in order to reach out to young adults was to create implementations of an impulsivity assessment tool and then add game features in order to make it more enjoyable. There are a variety of impulsivity tasks such as continuous performance tests, stop tasks, and Go/nogo tasks (Arce & Santisteban, 2006). The Go/nogo exists in a wide variety, such as reacting to certain numbers but not others (Gomez et al. 2000), different emotional facial expression (Schulz et al., 2007) and colored arrows (Vocat et al. 2008). The last in this case, seemed as a good candidate for implementation. It was chosen since it was deeply specified and easy to replicate the stimuli and do.
Within xDELIA the goal has been to provide technology enhanced learning tools for financial capabilities. Since a strong link between impulsivity and financial capabilities exist, and measurement devices to assess this (at least partially) exist, the issues are to get it out to people and to get people to use it. As stated earlier most cognitive tasks are quite boring, and would benefit from a more contextualized solution that is associated with something positive. As a first step to create such a contextualization for the task is to see if the assessment quality transfers from the go/nogo task used in laboratories into a game. This leads up to research question (RQ) 1:

RQ1. Effectiveness of standard task in game - Can the standard cognitive Go/nogo task be embedded as mini-games without losing its effectiveness as a measurement device?

In order to test for both positive and negative urgency, emotion induction procedures were needed. Common sources for induction are film clips (Rottenberg et al., 2007), pictures (Bradley et al. 2001a, 2001b; Bradley et al. 2007; Lang et al. 2008), and music (England & Clark, 1985; Etzel, Johnsen, Dickerson, Tranel, & Adolphs, 2006; Jallais & Gilet, 2010; Schneider, R. C. Gur, R. E. Gur, & Muenz, 1994). Film clips were disregarded since it was judged they are harder to fit into the game play of MINDswap, a Facebook game that was under development when this prototype was built. Regarding pictures the international affective picture system (IAPS; Lang et al., 2008) have pictures that are made to induce positive, negative and neutral emotional states. Over a thousand pictures exist in the database, all which have been validated with psychophysiological equipment. Of these 30 were chosen, based on the arousal and valence levels they would induce. Several pieces of music have been validated through experiments with self-reports (Mayer et al., 1995) to induce negative emotions as well as positive emotions. The effect of negative emotions where tested in this prototype, thus leading up to RQ2:

RQ2. How is induced negative emotional states related to performance in Go/nogo task?

3.5.1 Description of prototype

Two categories of implementations were made of the Go/nogo task, one category is simply called Go/nogo and the other one is called LineRacer. The Go/nogo is a direct implementation of the task in Vocat et al. (2008) whereas LineRacer has the added game elements of a robotic character walking on a surface and a score board (see Figure 3.4 & Figure 3.5). In addition to this each of the two categories have several implementation (see Table 3.1):

- Standard without music or pictures (S)
- With affective pictures: IAPS-pictures in the background (P)
- With music: Emotion-inducing music (M)
- Both pictures and music (PM)
Table 3.1 – Summary of the different versions of Go/nogo and LineRacer

<table>
<thead>
<tr>
<th></th>
<th>Standard</th>
<th>Pictures</th>
<th>Music</th>
<th>Pictures + music</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Go/nogo</strong></td>
<td>Go/nogo(S)</td>
<td>Go/nogo(P)</td>
<td>Go/nogo(M)</td>
<td>Go/nogo(PM)</td>
</tr>
<tr>
<td><strong>LineRacer</strong></td>
<td>LineRacer(S)</td>
<td>LineRacer(P)</td>
<td>LineRacer(M)</td>
<td>LineRacer(PM)</td>
</tr>
</tbody>
</table>

### 3.5.1.1 Game play

The player's actions are to react to a certain stimuli; first there is shown a black arrow pointing up or down, then another arrow is shown. If the second arrow is green AND pointing in the same direction as the first arrow the player is supposed to press "left ctrl" as fast as possible (go). If not, the player is supposed to not press (nogo) (see figure 3.6).

There are two different phases (without the player knowing this beforehand) that exchange each other. Both these phases together complete one cycle. The first phase is a "calm" phase in which the player reacts as soon as possible, but without any other stressors (besides for music, in those Go/nogo(M), Go/nogo(PM), LineRacer(M), and LineRacer(PM)). The second phase is where the player is subject to attempts of elicitation of emotions from the game. This is a picture, but also warning sound and text that tell the player to speed up.
3.5.1.2 Game logic and elements

Each turn is divided into a relaxation period (500ms), a pre-response period (black arrow, 500-1000ms), and a response period (green or blue arrow, 1500ms), see figure 3.7. The character has a constant walking speed of 3.0 m/s, when a faulty answer is given, he stops for the rest of the response period, when a correct answer is given he is running (with a speed of 6.0 m/s) for the rest of the response period. The mean reaction time (on go-trials) is calculated for the 14 first turns in every other phase, then multiplied with 0.8, as done in (Vocat, Pourtois, & Vuilleumier, 2008) This number, comparison number, is then compared to every new go-answer that is given. If the reaction time is higher than the comparison number, stress feedback will be displayed that tells the player that he or she is reacting too slowly. After 33 turns with stress feedback a cycle is completed (see figure 3.8) and a new one begins. There are three cycles throughout the game.

Figure 3.8 – A cycle broken down into its different parts
In the versions where pictures are used, P (with pictures) and PM (pictures and music), phase 2 is where all pictures are shown. Each cycle has its own induction, one cycle of neutral pictures, one with positive pictures, and one with negative pictures. This was done in order to be able to test for positive and negative urgency.

Music was chosen to be Léo Delibes - Mazurka for positive emotional induction and Bernard Hermann - The Murder for negative emotional induction, both validated by Mayer et al. (1995). In the pilot study described below only the negative music was used. Ten pictures of each neutral, positive, and negative were chosen from the international affective picture system (IAPS; Lang et al. 2008).
3.5.2 Technical Description

The Technical description is a constant work-in-progress during the prototype development process. An in-depth description of the prototypes, requirements, and design solutions are found in this chapter.

3.5.2.1 Prototype Specification Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>LineRacer series with/without induction of emotions.</td>
</tr>
<tr>
<td>Version</td>
<td>1</td>
</tr>
<tr>
<td>Status</td>
<td>Prototypes</td>
</tr>
<tr>
<td>Target User Group</td>
<td>Young adults</td>
</tr>
<tr>
<td>Prototype User Group</td>
<td>Internal + students</td>
</tr>
<tr>
<td>Purpose</td>
<td>Assess impulsivity</td>
</tr>
<tr>
<td>Description</td>
<td>A robot character is running on a line (both uphill and downhill), the player controls the acceleration by performing the go/nogo task.</td>
</tr>
<tr>
<td>Training Principle</td>
<td>No direct training, instead used as a tool in order to get formative feedback on impulsiveness training.</td>
</tr>
<tr>
<td>Transfer</td>
<td>Unknown</td>
</tr>
<tr>
<td>Context of Use</td>
<td>Evaluation</td>
</tr>
<tr>
<td>Appearance</td>
<td>Arrows with different colors. A black arrow is first shown and then either a blue or a green. If the green arrow is pointing at the same direction as the black was, then it is a go signal, all other signals are nogo. A robot character is racing on a road hovering in the air. Emotions are induced by pictures in the background and/or sound.</td>
</tr>
<tr>
<td>Inputs</td>
<td>Left Ctrl key</td>
</tr>
<tr>
<td>Feedback</td>
<td>Cumulative score that shows how far the robot character has travelled. The distance is dependent of how the go/nogo task was performed.</td>
</tr>
<tr>
<td>Guidance</td>
<td>Before the player starts to play he or she has to read the instructions of how to play the game.</td>
</tr>
<tr>
<td>Simulation</td>
<td>n/a</td>
</tr>
<tr>
<td>Gameplay/Challenges</td>
<td>To react as fast as possible under the go condition, but also make sure not to press during the nogo condition.</td>
</tr>
<tr>
<td>Emotional reporting</td>
<td>Self-assessment scales</td>
</tr>
<tr>
<td>Human instructor</td>
<td>Experiment instructor</td>
</tr>
<tr>
<td>Social network</td>
<td>None in this version</td>
</tr>
<tr>
<td>Logging devices</td>
<td>Cumulative score logging; reaction time and errors.</td>
</tr>
<tr>
<td>Development environment</td>
<td>Unity3D 2.8 Pro</td>
</tr>
<tr>
<td>Platform</td>
<td>PC</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Stand-alone application</td>
</tr>
<tr>
<td>Testing</td>
<td>Functionality and heuristic verification of gameplay, scoring and feedback. Tested on student groups in order to get behavioural data.</td>
</tr>
</tbody>
</table>
3.5.2.2 Technology
The programming environment Unity3D 2.8 Pro (http://unity3d.com/) is used for the implementation, because of its built in game engine and has support for connecting sensor devices into the game, via included classes. Also Unity3D can compile the game into different environments such as Windows, Mac, Wii, iPhone, and browsers. The programming languages used are JavaScript and C#, which is supported by Unity3D's own editor, and the compilation is running in Windows-environment (although can easily be compiled to run on other systems).

3.5.2.3 Implementation
Articles of Go/nogo tasks were collected (Gomez et al. 2000; Schulz et al. 2007; Vocat et al. 2008) in order to find a suitable version to implement. The version by Vocat et al. (2008) was chosen because of the in-depth description of the task and the stimuli they used were easy to reproduce. (August 2010)

Game logic from Go/nogo task was implemented and then also extended into LineRacer. In-game emotion induction procedures that were discussed were film clips, music, and pictures. Context stimuli to induce emotions that were discussed were different smells, distracting sounds (such as someone drilling in the wall, loud environment etc.), and visually disturbing setting. In order to maintain an easy controllable induction procedure context stimuli were disregarded. Of the in-game induction procedures, film clips was disregarded since they were judged they would not fit into the game without disturbing the player from completing the task. Also, film clips emotion induction are very dependent on if the player has seen the movie before, making the onset of the emotion earlier for some people than other. Thus music and pictures were chosen in order to induce emotions.

A collection of articles of pictures (Bradley et al. 2001a, 2001b; Bradley et al. 2007; Lang et al. 2008) and music (England & Clark, 1985; Etzel, Johnsen, Dickerson, Tranel, & Adolphs, 2006; Jallais & Gilet, 2010; Schneider, R. C. Gur, R. E. Gur, & Muenz, 1994) was analysed in order to make a design suggestion. (September 2010)

Music was chosen to be Léo Delibes - Mazurka for positive emotional induction and Bernard Hermann - The Murder (Mayer et al. 1995) and ten pictures of each neutral, positive, and negative was chosen from the international affective picture system (IAPS; Lang et al. 2008). Prototypes of Go/nogo and LineRacer were built with music, pictures and music.

Pilot testing was performed on students and initial analysis completed. WP3 was cancelled and therefore Go/nogo and LineRacer development and testing ceased. (October 2010)

3.5.2.4 System Description
The use is described in this section as well as the flow of the game, both in time and in a class diagram.

3.5.2.4.1 Use cases
Currently, the players can play the game by themselves, but an administrator is required in order to analyse the results, as well as to report the players' score (Figure 3.9).
3.5.2.4.2 Sequence diagram
The game states are constantly progressing, independent of the player's performance. No way of regressing back to a previous game state exists (see figure 3.10 on the next page).
Figure 3.10 – The sequence in which the game is played
3.5.2.4.3 Class diagram

Game classes and their relationship are presented in Figure 3.11, with following information on each class below.

![Class diagram of LineRacer](image)

**Game Handler**
This is the main class, which controls the game start, flow, and finish. It keeps track of time and controls each turn, with according graphics, and listens for player input. It also detect when a player reacts to slow and gives feedback accordingly.

*Figure 3.11 – Class diagram of LineRacer*

**Game Handler**
This is the main class, which controls the game start, flow, and finish. It keeps track of time and controls each turn, with according graphics, and listens for player input. It also detect when a player reacts to slow and gives feedback accordingly.
Player Controller
This is were data from the player, not the play session, is kept. It is simply created once and exists for Score Counter to fetch player information from.

Score Counter
Writes player results to file, files are named after which player who plays and also which phase the score is for.

Line Racer Init
Initiates the game and collect player data. When the player is ready it starts up the level and the Game Handler.

Level Handler
Contains the level information and builds the level.

Player Speed Controller
Keeps track of maximum reaction time the player is allowed to have before given a warning.

Camera
The placement of the camera. Keeps the character in focus and scrolls sideways.

Character
The character the player controls. Speed levels (normal, stopped, and accelerated) are set in this class. Collisions (only with the ground) are detected here.

Audio Controller
Has information of where all music and sound effects are located and activates them on command.

GUI
Contains all arrows and instructions in the game.

Image Presenter
Each of the emotional induction conditions (neutral, positive, and negative) are stored here. The object is attached to the character object in order to maintain a stable background picture.

3.5.2.5 Measures and data logging
Following in this section there is explained how the data is logged and what is logged. The WP3 was cancelled in October and thus the psychophysiological measurements were not implemented. All in-game data logging are working properly.

3.5.2.5.1 Psychophysiological measurements
Intended measuring devices for this game was galvanic skin-response and electromyography, in order to determine players’ valence and arousal during gameplay. However, this was never implemented and activities regarding this ceased when WP3 was put down. However, it is planned to be studied at BTH outside of the xDELIA project.
3.5.2.5.2 In-game data logging

All data is stored in txt-files, which are named [results + "player entered data" + "phase number".txt] and a new one is created for each new phase. Example data from one output file:

```
Codes: 0 - Go condition; 1 - Nogo condition 1; 2- Nogo condition 2;

For trial number: 1 code: 1 your answer was: True. reaction time: 1500
For trial number: 2 code: 0 your answer was: True. reaction time: 294,964
For trial number: 3 code: 1 your answer was: True. reaction time: 1500
For trial number: 4 code: 0 your answer was: True. reaction time: 319,5023

[...]
For trial number: 32 code: 2 your answer was: False. reaction time: 253,4662
For trial number: 33 code: 0 your answer was: True. reaction time: 335,4661

Total distance: 110,703
```

The data logged are trial number, which is logged in order to easy find which picture that was shown and also in order to see if players are faster/slower after a specific answer. Code refers to if it was a go or nogo trial, go is 0, green arrow point opposite direction is 1, and blue arrow is 2. Answer is whether the player answered correct (True) or incorrect (False) on that specific trial. Reaction time is the player’s reaction time in mille seconds.

3.5.2.6 Prototype Design

The complexity of design and functionality requirements in software and game development calls for a systematic approach in the initial phase to ensure that all requirements have been accounted for (Andrew & Greene, 2005). In the development of the Go/nogo and LineRacer prototypes all requirements regarding both graphical design and functionality should have an explicit design solution. In this chapter the requirements are presented as they were interpreted by developers and agreed upon among involved partners. Each requirement is then redefined as one or several design solutions, which are required in order to fulfil that specific requirement.

3.5.2.7 Requirements, Design Solutions and Issues

The purpose of the requirements is to cover all the desired design- and functional features in an application. A requirement should be stated in such a way that it can be easily explained by a set of design solutions. Requirements are listed with unique ID numbers as an abbreviation of R_ID, Category and Sub-category, such as R_ID12.3. Following each requirement, the design solutions for that specific requirement is listed as well as issues associated with the designs. The design items in are described by unique design ID, reference to the specific requirement in the requirement list, as well as a description of how the requirement was ultimately implemented.
The requirements are listed with all follow-up requirements which are dependent on some other requirement.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:1.1</td>
<td>Go/nogo task</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:1</td>
<td>Arrows were created and game logic implemented. The choice of the first arrow (the black one) is randomized so there is 50% chance of it pointing upwards or downwards, respectively. Input was implemented to be left Ctrl. Turns and cycles were implemented. Instructions of how the game works were written and placed in the beginning of the game.</td>
</tr>
</tbody>
</table>

**Issue ID**

<table>
<thead>
<tr>
<th>Issue ID</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ID:1</td>
<td>The text shown is not centred, and is also insensitive to screen resolution changes, making it hard to read at some resolutions. In Unity3D 3.x there is support for dynamic fonts, which automatically scales itself to the right resolution.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:1.2</td>
<td>Correct output</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:2</td>
<td>The output was defined, as describe above. For each player a set of files, one for each phase, are created. This was done in due to be able to compare the different phases separately.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:1.3</td>
<td>Two phases in each level - training and test</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:3</td>
<td>Each cycle were divided into two phases, training and test. The reaction time on go-trials are saved from the training phase, averaged, and then multiplied with 0.8. This does then feed into the test phase and as soon as the player's reaction time is larger than this number, a warning sound will play and the text &quot;Too slow, press faster&quot; are displayed. The warning sound is attached to the character in order to keep the sound effects proportional. Lists for each phase were created with a random generator, skewed for the right properties (2/3 go trials and 1/6 of each nogo-trials). These lists were placed in txt-files so that it would be easy to exchange for people without programming skills.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:2.1</td>
<td>LineRacer character</td>
</tr>
</tbody>
</table>

**Design Solution ID**

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:4</td>
<td>A robot character was added, given a constant speed of 3.0 m/s moving to the right when the player does not answer, with the speed 6.0 m/s when the player has answered correctly, and stops when the player answering incorrectly on nogo-trials. A scoreboard was implemented in order for player to see his or her score. Different animations for walking, running, and stopping were defined. The robot also makes a footstep sound at each step.</td>
</tr>
</tbody>
</table>

**Issue ID**

<table>
<thead>
<tr>
<th>Issue ID</th>
<th></th>
</tr>
</thead>
</table>
| I_ID:2   | There is an open issue about how the robot (or the message) should react when the player gets the message the he/she is too slow. Currently the robot
is reacting as if it were a completely normal go-press.

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:2.2</td>
<td>A level the character can walk on</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:5</td>
<td>A level was created, consisting of platforms with small &quot;hills&quot; so the character is not simply moving forward but also a bit uphill and downhill. The level was made large enough so the player have no chance to reaching the end of it, even if optimally played. A spawn point for the character was also created.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.1</td>
<td>Pictures in the background</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:6</td>
<td>In order to have a picture in the background a thin box was created that covered that whole screen. This box was placed behind (seen from the camera) the character and level. The box was attached to the character and follows it, so from the view of the player, it seems like it is a solid background picture.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue ID</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ID:3</td>
<td>The pictures are making the game demanding to run on slow computers. Also, there is an issue of scaling the box to cover the whole screen at different resolutions. Creating a second camera with responsibility for a GUI texture, which will be rendered first, thus giving a more system efficient and scalable background picture.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.2</td>
<td>Three induction states (neutral, negative, and positive) with pictures</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:7</td>
<td>Each of the pictures from the three categories (neutral, positive, and negative) was stored in three arrays. During the second phase in each cycle, a script changed the box's texture into a new picture every turn.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_ID:3.3</td>
<td>Music</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design Solution ID</th>
<th>Design Solution Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_ID:8</td>
<td>Music was imported and attached to the character as an audio source. Since the audio listener is in the camera the different sounds given from the robot will be proportional from each other, no matter which configuration that is used. Choice of music is made before compiling the project.</td>
</tr>
</tbody>
</table>

3.5.3 Evaluation

Following in this section about data gathering and analysis in a pilot study is described and some preliminary results are discussed. Then a heuristic evaluation where usability and playability issues and possible solutions to these issues are presented.
3.5.3.1 Pilot study - Data gathering and analysis

Twenty-three players volunteered to try out the game and were divided on the different types as follows:

- Go/nogo(S) - six
- Go/nogo(M) - six (five successful)
- LineRacer(S) - three
- LineRacer(M) - three
- LineRacer(PM) - four

Out of these one player of the Go/nogo(M) misunderstood the task and did not finish the test. All players received the BIS and were instructed to ask questions if something was unclear. After they had filled in them they got a short description of the Go/nogo task in addition to the instructions in the game. After each cycle they were instructed to fill in a questionnaire asking about different emotions during gameplay, and also if they focused on anything in particular in the game in order to assess this if a redesign was to be made.

![Figure 3.12 – Test of Go/nogo task](image-url)
Average reaction times were computed for all players across phases in the different prototypes (see Table 3.2) and are also shown as a diagram in Figure 3.14.

<table>
<thead>
<tr>
<th></th>
<th>Phase (1)</th>
<th>Phase (2)</th>
<th>Phase (3)</th>
<th>Phase (4)</th>
<th>Phase (5)</th>
<th>Phase (6)</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Go/nogo(S)</td>
<td>371</td>
<td>296</td>
<td>286</td>
<td>288</td>
<td>274</td>
<td>277</td>
<td>298</td>
</tr>
<tr>
<td>Go/nogo(M)</td>
<td>467</td>
<td>379</td>
<td>433</td>
<td>369</td>
<td>358</td>
<td>385</td>
<td>399</td>
</tr>
<tr>
<td>LineRacer(S)</td>
<td>363</td>
<td>355</td>
<td>339</td>
<td>342</td>
<td>332</td>
<td>335</td>
<td>344</td>
</tr>
<tr>
<td>LineRacer(M)</td>
<td>373</td>
<td>367</td>
<td>333</td>
<td>340</td>
<td>309</td>
<td>327</td>
<td>341</td>
</tr>
<tr>
<td>LineRacer(PM)</td>
<td>436</td>
<td>406</td>
<td>357</td>
<td>408</td>
<td>394</td>
<td>383</td>
<td>397</td>
</tr>
</tbody>
</table>

Table 3.2 – Reaction times (in ms) for each phase (cycle number in brackets) in the different prototypes.
The game score, as well as reaction time, were compared to each of the second-order factors from the BIS in order to control if the different games measure the same impulsivity construct. During that analysis some correlational trends were between BIS and game score, specifically a negative correlation between attentional impulsivity and game score in the Go/no-go games. These are shown in Figure 3.13 and Figure 3.14. There were also a slight trend between motor impulsivity and game score in both LineRacer(S) (Figure 3.15) and LineRacer (M) (Figure 3.16), although they are contradicting each other. No relationships between the BIS and reaction time were found. As seen when comparing between prototype series (Figure 3.18 and Figure 3.19), low reaction time equalled few mistakes. However, within each prototype there were no trade-off between speed and accuracy.
Figure 3.16 – Relationship between game score and attentional impulsivity in Go/nogo(S)

Figure 3.17 – Relationship between game score and attentional impulsivity
Even though the numbers of participants were few and thus real statistical test would be invalid some trends that was seen are discussed in this section.

Regarding RQ1 "Effectiveness of standard task in game - Can the standard cognitive Go/nogo task be embedded as mini-games without losing its effectiveness as a measurement device?" the first part of answering that question was to see if the Go/nogo task was sufficiently satisfying. The average reaction time in Go/nogo(S) was 298ms, compared to Vocat et al. (2008) where the mean reaction time was 286ms, thus drawing the conclusion that an adequate version of the Go/nogo task was implemented. When comparing the trend results from Go/nogo(S) and LineRacer(S) with the BIS, it was clear that they were correlating with different impulsivity aspects in this test. This is an indication that the LineRacer(S) cannot directly replace the Go/nogo task as a measurement device for the same type of impulsiveness.
Lower attentional impulsivity was correlated with higher game score in both Go/nogo tasks, one possible reason this was not present in the LineRacer versions of the prototype could be that less attentional resources was required since LineRacer have more features that draws attention (Corbetta & Shulman, 2002).

Regarding RQ2 "Which types of emotional inducing procedures have an impact when used in the game", when looking at the reaction times from the pilot study (Table 3.2 and Figure 3.14) we can clearly see that Go/nogo(S) and Go/nogo(M) are far from each other, on average 101ms. This could be a result from different emotional responses, but at the same time it is also possible that it is an issue of the attentional system. Given the theory of visual dominance (Posner et al., 1976) there are four characteristics of vision when discussing multi-modal input. First, visual stimuli are less automatically alerting than other modalities. Secondly, visual events must be actively processed before they become effective alerting stimuli. Thirdly, there is limited processing power that has to be divided between the different modalities, which means that if one modality gets more attention, the others get less. And fourth, there is a bias towards giving extra attention to the visual system when reliable information from the visual modality. People who score high on attentional impulsivity have a hard time focusing on a task at the time, thus they should have a harder time focusing on the visual stimuli. When tested on the Go/nogo(S) this is reflected in game score. This can also be seen, even clearer, in the Go/nogo(M). When a comparison between these two prototypes is made, a clear distinction can be seen in reaction time. Two possible reasons are discussed here, visual dominance and emotional impact.

According the Posner et al. (1976) fourth statement that people have an attentional bias towards vision when reliable information comes from visual input. However, it can be argue that music is more attentional alerting than pictures of arrows that changes without any animations. If attention was pulled towards audio when doing the Go/nogo(M), longer reaction times would be required for the players. When looking at the LineRacer(S) and LineRacer(M) data they differ with 3ms. The animated robot character would maintain visual attention from the subjects, even though the music was playing, negating the effect seen at Go/nogo(M). To further strengthen this claim, no correlations were seen between any of the LineRacer versions and attentional impulsivity.

The model describe in Wallace (2009), where the handling of emotions takes a toll on cognitive load, could account for the differences between the reaction time in Go/nogo(S) and Go/nogo(M), however it will encounter problems when LineRacer(S) and LineRacer(M) is in the picture. Nothing conclusive can be said about the subject, further investigations are needed.

### 3.5.3.2 Heuristic evaluation and improvements

A heuristic evaluation was conducted in order to point out possible design flaws from a gaming perspective, and suggest improvements to them. The heuristic evaluation used on the Go/nogo and LineRacer is part of a generic evaluation tool kit, which is being developed at Blekinge Institute of Technology and is used on all prototypes produced at BTH. The heuristics are divided into a set of categories inspecting different aspects of the game prototype. Framework and evaluation results are presented below.

If two or more reviewers agree on an issue, it’s immediately noted. If only one has an issue, other reviewers have to be consulted and convinced. In this section the different findings and issues (I) from the heuristic evaluation are discussed briefly and where it is due, possible solutions (PS) are suggested.

#### Table: Heuristic Evaluation

<table>
<thead>
<tr>
<th>H_ID1 CONSISTENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td>The game is not consistent with regards to menus and pause screens, which are not following the same theme between them.</td>
</tr>
</tbody>
</table>
### H_ID2_CONSISTENCY

| When the player has written his or her name it is required to press a button with the mouse, the enter key does not work. | General usability rules such as both keyboard and mouse work to proceed in the game menus |

### H_ID3_FEEDBACK

| Feedback is contra dictionary when the player reacts too slowly. On the one hand, the character is running faster, but on the other hand the text "Too slow, press faster" is displayed. | Error message can be changed to not give the impression that a complete failure was done, e.g. "No high score for you" or "You're better than that, speed up!". |
### H_ID4_FEEDBACK

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>No feedback of the current speed is given, this might enhance the player's realization of how big the effect from playing good is.</td>
<td>Current speed can be shown in some way, either a clear coded version such as changing the colour of the screen or to add a value of meters/second in relation to meters travelled.</td>
</tr>
</tbody>
</table>

### H_ID5_USE EASY-TO-UNDERSTAND TERMINOLOGY AND PROVIDE HELP

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Because of the structure of the task it is hard to describe it in only words in a game. It should be complemented with pictures or a demonstration in order to more easily describe how the game is played.</td>
<td>Provide an optional slideshow of how the game is played that a new player can watch, but which an experienced player can skip.</td>
</tr>
</tbody>
</table>

### H_ID6_SCREEN LAYOUT

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text on introduction menu is not fitted to screen, a bit small at some resolutions.</td>
<td>Use a dynamic font that is adjusted to the resolution.</td>
</tr>
</tbody>
</table>

### H_ID7_SCREEN LAYOUT

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score display is too small.</td>
<td>Enlarge score display, make it dependent on resolution so it will take up the same screen area at different resolutions.</td>
</tr>
</tbody>
</table>

### H_ID8_SCREEN LAYOUT

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>At some resolutions the pictures are not seen as well as with other resolutions. At other resolutions the picture does not cover the whole background.</td>
<td>Make a solution that makes the picture fit no more and no less of the total screen area.</td>
</tr>
</tbody>
</table>

### H_ID9_GAME CONTROLS

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only one button to press, need to be customizable in order to let left-handed people play at the same convenience level.</td>
<td>Make optional keys for people to press, also make it possible to play with the mouse.</td>
</tr>
</tbody>
</table>
**H_ID10_AUDIO-VISUAL REPRESENTATION**

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very disturbing sound when reacting too slowly.</td>
<td>The point for the sound is to be stressful, but an alternative solution could be to have a nicer sound but make it a competition against someone or something, thus inducing stress this way instead.</td>
</tr>
</tbody>
</table>

**H_ID11_CLEAR GOAL**

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unclear goal, the game becomes dull.</td>
<td>Make a clear goal, with different sub-goals, to motivate the player.</td>
</tr>
</tbody>
</table>

**H_ID12_CLEAR GOAL**

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game does not provide player created goals (like a high-score list).</td>
<td>Create a high score list, possibly also on-line competition. A levelling, or achievement, system could be used to make people more likely to set up goals for themselves.</td>
</tr>
</tbody>
</table>

**H_ID13_PROGRESS**

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>There is no comparing of results from various players.</td>
<td>As suggested above, high score of on-line competition would help against this issue.</td>
</tr>
</tbody>
</table>

**H_ID14_PROGRESS**

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Players have no idea on how much time is left till the end of a level or how far the level has progressed. Level progress indicator is missing.</td>
<td>Level indicator, description of level beforehand would be a solution.</td>
</tr>
</tbody>
</table>

**H_ID15_REWARDS**

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Players are not rewarded at all.</td>
<td>Provide achievements related to their personal score or friends' score to motivate the player.</td>
</tr>
</tbody>
</table>
### $H_{\text{id}16}$ THE PLAYER IS IN CONTROL

| The difference in scores between good players and bad are small. | Perhaps scale up the difference could be an idea. Up scaling the points, and also give feedback of how well a player is performing, according to pre-set criteria. |

### $H_{\text{id}17}$ THE PLAYER IS IN CONTROL

| The difference in scores between good players and bad are small. | Perhaps scale up the difference could be an idea. Up scaling the points, and also give feedback of how well a player is performing, according to pre-set criteria. |

### $H_{\text{id}18}$ THE PLAYER IS IN CONTROL

| Each phase ends a bit abruptly, no feedback of how long time is left in the current phase. | A goal state or countdown timer would minimize the risk of players feeling that the game suddenly ends. |

### $H_{\text{id}19}$ CHALLENGE, STRATEGY AND PACE

| The player does not have any idea if his or her performance is good or bad. | Some type of levelling system, achievements, average for player etc. could be used in order to let the player know about the performance. Feedback can be given after each turn. Right now only negative feedback is supported, but really good reaction times could be positively reinforced as well. |

### $H_{\text{id}20}$ CHALLENGE, STRATEGY AND PACE

| Neutral pictures are the first ones to be shown to the player, these are a bit boring to look | Put the neutral pictures in the middle in order to keep up the interest more from the start. |

### $H_{\text{id}21}$ CHALLENGE, STRATEGY AND PACE

<p>| Neutral pictures are the first ones to be shown to the player, these are a bit boring to look | Put the neutral pictures in the middle in order to keep up the interest more from the start. |</p>
<table>
<thead>
<tr>
<th><strong>H_ID22</strong> <em>THE FIRST-TIME EXPERIENCE IS ENCOURAGING</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>After the first few turns the player knows what the game is about, so there is no motivation to keep playing.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>H_ID23</strong> <em>GAME STORY</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>No story in the game.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>H_ID24</strong> <em>REPETITIVE TASKS</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>The task in the game is repetitive.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>H_ID25</strong> <em>DIFFERENT PLAYING STYLES</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>The task in the game is repetitive.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>H_ID26</strong> <em>GAME STAGNATION</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>Player has no feeling of progression. No feedback on how the game is developing.</td>
</tr>
</tbody>
</table>
3.5.4 Reflections

A game for Facebook, MINDswap, has been implemented, where a go/nogo task has been implemented as thought bubbles. A design implication from this prototype series is to make sure that the visual and audio are kept on a stable over the playing of the game in order to keep a reliable assessment of impulsiveness. Also, much of the results from the heuristic evaluation could feed into the MINDswap game.

In order to separate how big the effects from visual dominance and emotional influence the game needs to be rebuilt in order to address this. Also this would have the benefit of using new features in Unity3D 3.x, where many of the problems are solved that were discovered during the implementation process and heuristic evaluation.

As discussed previously, correlations between behavioural tests and self-report measures have not been very successful. This prototype series did not show any correlations between the normal measures of go/nogo tasks (false positives on nogo trials), but there were seen a couple between BIS and game score. This might of course just be a random effect due to the small sample size, but given the possible bridge between weighted behavioural tests and self-report questionnaires it is a field to look into the future.

3.5.5 Conclusions and future work

It has been seen that the game score correlated better with BIS than the number of errors or reaction time. Even though it might seem adequate, further investigations of how to balance the game score is needed in order to be able to give players feedback of their impulsiveness, if this concept is to be developed further in the MINDswap game.

The difference in Go/nogo(S) and Go/nogo(M) is still subject to the question whether it is emotional processing of the music or directed sensory attentional resources towards that affect the reaction time. In future studies this ought to be controlled for and tested separately with neutral music (that is, music that does not induce, or induce very little, emotions) as well.

Impulsiveness measure differed from the Go/nogo and the LineRacer series, which might of course only be because of the number of players, but can also be because of qualitative differences between the games. In order to understand game play better and what effects different basic modalities have on small, casual games future investigations in the subject are needed.

One property of the go/nogo task is that is has a relative high test-retest reliability, which needs to be tested for if the work with this prototype will continue.

Prototypes developed need to be able to pass as games right from the start, otherwise it will be hard to draw any conclusions about them. In order to do this, player motivational aspects need to be addressed, thus making the game not only an assessment/training tool, but also a game that people might want to play. The lack of story, progression, rewards, and encouraging first-time experience found in the heuristics will need to be solved in order to make the game more enjoyable for the players.
3.6 **MINDswap**

The overall learning intervention in WP3 has gradually formed from the previous concepts and prototypes. The story line from the banking game was replaced by the notion of young people getting along with their everyday life. An everyday life that contains temptations and obstacles that has to be dealt with. The game is a 3D online game that can be played with friends in Facebook (Figure 3.20) (the game can be found here http://apps.facebook.com/finboard), and where the main characters face a series of challenges and obstacles that they have to overcome to reach their final destiny. The main twist is that players are not in full control of their avatars’ actions, and that they have to find ways and devise in-game and real world strategies to overcome the avatar’s reluctance to behave appropriately. Figure 3.21 shows how the avatar can be controlled.

![Figure 3.20 – Screen shot from the game](image)

### Figure 3.20 – Screen shot from the game

#### 3.6.1 Game synopsis

The game is built around a variety of financial capability themes, that set the scene for the main challenges in each level, and the backdrop against which the player will apply knowledge and skills to steer the avatar through the world and towards the final goal. We will use Level 1 (L1) of the game to help illustrate the more abstract concepts and ideas behind the game.

### 3.6.1.1 Level 1 (L1): Synopsis and setting

Your avatar, let’s call her Juanita, has decided to move out of her parents’ home to go and live close to her best friends in Curbisheim. After she arrives at the small market town, she explores the town and tries to find out where her friends live. The money she got from her father however doesn’t last, and she needs to look for small jobs to pay for food and for the Youth Hostel. The biggest problem however is that Juanita get’s easily distracted by the town’s social life, and the little money she gets from her jobs never lasts very long. Neither is she able to keep the better paid jobs, because of her disastrous time keeping. A lack of perseverance and a disposition to rash action are the obstacles that the player is asked to overcome. Simple actions, such as keeping to a schedule, not leaving the avatar to long on its own, and strategies such as
preventing the avatar from becoming too hungry or carrying too much cash permit increasing control over and transfer of abilities to the avatar.

### 3.6.1.2 Key features

- The avatar takes decisions and behaves autonomously
- The player can control the avatar by changing elements of the environment, employing strategies that prevent maladaptive behaviour, and increasing the avatar’s knowledge and abilities
- Gameplay extends to the real world by enabling the transfer of player skills to the avatar
- The game is built around financial capability themes that condition the avatar’s behaviour
- The game world is inhabited by the player and her friends, offering a diverse range of social interactions

### 3.6.2 Characters

The main character of the game is the player’s avatar. Avatars at the start are difficult to control (Figure 3.21). They show attitudes and behaviours that create much trouble and money-related problems for them. They may decide at the spur of the moment to splash out on an expensive DVD player. Or they may not care much about saving money for a rainy day. They have difficulties keeping up with their credit card and mortgage payments and they try to make ends meet by having several part-time jobs that don’t pay well. At the same time, avatars can learn if they are shown or given the right tools. They can acquire knowledge at the same time than the player does, and can even change their psychological make-up.

Equally important are the player’s friends, who live in the game world as proxy-avatars\(^1\). You can always find them somewhere hanging around or doing things, even if they are not online. If you go offline for a prolonged period of time – a week perhaps – you may have to ask a friend to look after your avatar, otherwise they may get themselves into problems. Friends may also be able to help you out, if you are short of money or if you need some advice. Friends provide the gameplay with a strong link to the real world and are a great to play with (or against!).

There are also many non-player characters (NPCs) in the game, so that there is always something happening in town. You can talk to NPCs, ask them for help or for things you want to buy from them. NPCs can be strangers to you, or friends. They may have a particular role in the game, such as the shopkeeper, the truant, the neighbour, or merely be passersby.

---

\(^1\) Since the game emulates networking for reasons of robustness, the actions of the player’s friends are simulated, and are different in each instance of the game client. This doesn’t result in major problems as long as multiplayer gameplay enforces consistency.
3.6.3 Story (Level 1)

3.6.3.1 Story (L1) from the avatar’s perspective

Juanita has arrived in Curbisheim and doesn’t quite know what to do next. She feels peckish and starts walking around the town to find something to eat. Finally she finds a restaurant and has a meal which cost her an arm and a leg. Juanita knows that before it becomes dark, she needs to find a place to stay. She can either find a friend’s house or go to the Youth Hostel. So, next she asks people in the street for directions to one of her friend’s place, but in the end she can’t find it. Already tired, she decides to go to a bar, and stays there for most of the evening. And then has nowhere to stay but the park bench...

At the break of dawn, Juanita wakes up and looks around to find somewhere to have breakfast. She has to be careful not to spend the money too quickly, but then surely, she will find a job somewhere. On her way she comes across the owner of the newsstand who asks her whether she’d like a job distributing the papers in the morning. It is good money and you accept the job. The only caveat is that you need to be there three times a week, at 7am, to do the job. And ‘you’ means the avatar and the player. Beside the money, it also gives you a lot of experience points. Now, you just need to find a way of preventing Juanita to spend the money she earns on accessories and drinks at the bar. Perhaps if she wouldn’t carry all that money around all the time...

Finally you found one of your friends, just by coincidence, walking in the street. He gives you his address and invites you for lunch... A message just arrived in your inbox. It’s your friend who says that he got a job for you to move his things to the storage warehouse. At 5pm tomorrow. Can you make it? It’s important to take care of your friends, so you say yes. Meaning you will have to come online at around that time and make your avatar do this job. Might be a problem – you do it for free, it’s your friend – but the avatar doesn’t care much about this at this time. For sure, your avatar will do it if you take it out for a pizza.
3.6.3.2 Story (L1) from the player’s perspective

The player, just as Juanita, didn’t have much of a clue at the beginning of the game what to do. Challenges are revealed gradually throughout the game, and for some things, the player needs to complete tasks and reach goals in a particular sequence. Since Juanita’s thoughts are visible to you in thought bubbles, you know she wants to go for food. You can steer her to a place that’s not too expensive – with some help and hints from the game you might be able to stay clear of the three star restaurant. Perhaps you can even find out whilst walking around where the Youth Hostel is. Better to check in now... Oops, you haven’t noticed the department store on the way mapped out by the pathfinder, and because Juanita is already too close, you can’t prevent her from getting a new pair of sunglasses. Before you log off, you manage to steer Juanita to the Youth Hostel and leave her there for the night. That way she won’t cause problems whilst you are away.

As you come back online the day after, Juanita is already up, showered and ready to look for a job. First breakfast though. A thought comes to your mind. Why not save on breakfast, so the money lasts a bit longer. Unfortunately, the more hungry the avatar is, the more impulsive it becomes, at least in terms of food intake. If you are not careful, and your avatar is deprived of food, it might pop into binge eating mode, and spend all the money in an expensive eatery. So better find some breakfast... Later that day she meets the newspaper agent who offers her a job. Of course you want to accept, and fortunately she seems to have something to do during the day as well. But only on three days, not the five you wanted. She needs to deliver the papers at 7am, which means that you, the player, need to come online at around that time to get her out of bed and accompany her for a few minutes. If you forget, she’ll forget – unless of course you’ve trained her by being on time online regularly. Then she can do it on her own and you can stay in bed. Next thing before going offline is to move her safely back to the Youth Hostel. Before arriving there, you receive a pop-up message that asks “How much did Juanita spend today?” Can you remember? If yes, you’ll get some experience points. Experience points are important since they give you access to the next level, and to some fanciful tools and opportunities.

Back online after two days away. A message awaits you saying that a substitute had to do your paper delivery, and that your boss is pretty mad. One more time and you are out. Oh, and another message telling you about the state of your avatar and what it has been up to in your absence. More accessories... On the way from the YH, you meet a friend of yours in the street who invites you for lunch. You aren’t sure whether you can come, but he gives you his address anyway. Later, he sends you a message asking whether you can help him with moving some stuff to storage. You feel obliged, your avatar doesn’t. Being on good terms with your friends, especially in such a small town, is quite important. So you bribe your avatar with the promise of a pizza and a night out. You feel a bit weary of your avatar’s capricious behaviour and take it to therapy. There you have a selection of techniques that can help reduce your avatar’s impulsivity. You play the go/no-go mini-game and are quite pleased with the outcome – a bit better than last time around. Your avatar will sleep much better tonight, and tomorrow it will be mostly docile and do as you wish. A good opportunity to get some of the more tedious things done.

3.6.4 Game world and player interface

We are proposing a game world with a cartoon environment and characters because 3D models of that kind are commercially available for a modest price, and many of the ‘realistic’ 3D models we’ve seen for cities have quite a depressing aesthetic feel. The cartoon world comes quite close to what you see in some of the popular Facebook games such as Yoville or in the Sims. The only caveat is that the interiors look very pink, orange, and blue, and we may have to make some changes to the colours.

3.6.4.1 Navigation

The player will mainly use the left mouse button to steer the avatar around the world. The right mouse button opens the contextual menu, wheel click will rotate the screen, and wheel scroll will zoom in and out. The left mouse button has triggers different actions depending on where the player points with the mouse. Left-clicking on a walkable surface suggests to the avatar to walk to that location. Left-clicking on an object that can be picked up will do this. To visualise to the player the different actions that the left mouse
button triggers, and overlay symbol indicates what the action will be. Moreover, if the player is close to an entrance or exit, these are marked by overlay arrows pointing in the respective direction.

3.6.4.2 HUD

Information about game and avatar state is displayed in the head-up display. The player can access the settings menu from the HUD, and elements displayed in the HUD can be toggled in the settings menu. Users can choose to display state information that is available for a particular level. Friends’ profile pictures and online status can also be displayed in the HUD, and players can use the context menu to get profile information, send a message, or challenge a friend with a mini-game for instance.

3.6.4.3 Menu and settings

In addition to the using the left mouse button to navigate and interact with the game, players can get information and interact via the context menu (right mouse button). The context menu provides a range of menu options depending on the current location and context, although interaction via the context menu is typically much slower than via the left mouse button.

The player can enter the settings menu at any time during the game. While doing this, the game pauses. The menu contains several parts: basic user information (partly pulled from the social graph), notification settings, HUD customisation, avatar customisation, and profile aggregation.

3.6.5 Online fincap game (use-centric view)

The high-level use cases derived from the system requirements (see D 16.4.3.1) are shown in Figure 3.22 below. All system elements of the game-centric view are shown, except those in the category ‘internal technology’, which site inside the system boundary. Requirements generate use cases that can be communicated to game designers and refined through the posterior iterative development. In fact, the use cases below are mixes of player actions (the ‘use case’) requirements. The next step now is to refine the high-level use cases and generate a set of briefs for the design of the game system (Table 3.3).
Table 3.3 – Refining the high-level use cases

<table>
<thead>
<tr>
<th>Actor</th>
<th>Use case</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UC-1</td>
<td>Player</td>
<td>Play online game</td>
</tr>
<tr>
<td></td>
<td>&lt;&lt;include&gt;&gt; UC-1.1 – UC-1.4</td>
<td>The player accesses the game through Facebook. For first time access, the game application asks for permission to access profile data and the Wall. Also, the player will have to install the Unity Web Player on the browser. A splash screen is shown while the game loads. The game automatically loads the previously stored session for the player. Finally, there may be important announcements made at start-up. Especially if anything important happened to the avatar or in the game world while the player was offline.</td>
</tr>
<tr>
<td>UC-1.1</td>
<td>Start game</td>
<td>The player engages with the game world through her avatar. Unlike in typical games however, players do not have full control over the avatar. This allows us to enforce behaviour in the avatar that players would otherwise not select, because it would be strategically sub-optimal. This feature presents a challenge for the player, because she now has to find different ways to effectively control her avatar. In some cases, she will need to perform a real world task, and the outcome is then transferred to the avatar in form of skills gains. In the same vein, the player may answer knowledge questions or be asked to do some calculations. If successful, the knowledge of the avatar is increased, which may allow it to perform a new behaviour, or perform is in a better way. Every level of the game</td>
</tr>
<tr>
<td>UC-1.2</td>
<td>Play game</td>
<td></td>
</tr>
</tbody>
</table>
poses a series of challenges to the player that she will have to overcome by carrying out specific tasks and reach prescribed subgoals. The main challenge or goal in any level is built around themes that reflect issues in financial capability of young people, and that draw on existing educational frameworks and also on our own work in xDelia (see D10-3.2). In addition to the intrinsic personality traits and dispositions of our avatar, the themes are also affecting the behavioural repertoire of the avatar. Players will have to train and control the avatar with sundry cognitive and behaviour strategies in order to achieve the goals and pass to the next level.

| UC-1.3 | Pause/resume game | The player can pause the game, although this doesn’t affect the game clock, which is synchronised with real world time and continues to run even if the game is paused. The player may want to pause the game for instance if she wants to look around the world while doing some other activity. |
| UC-1.4 | Save game | The game state is automatically saved to the database in reasonable time intervals (e.g. 10 sec.) to avoid performance degradation. |

| UC-2 Player | Get visual information and feedback | The player can enter the settings menu at any time during the game. While doing this, the game pauses. The menu contains several parts: basic user information (partly pulled from the social graph), notification settings, HUD customisation, avatar customisation, and profile aggregation. |
| UC-2.1 | View settings menu | Information about game and avatar state is displayed in the head-up display. The player can access the settings menu from the HUD. Elements displayed in the HUD can be toggled through in the settings menu. Users can choose to display state information that is available for the particular level. |
| UC-2.2 | View HUD with state information | This is an out-of-game feature. Players can access their behavioural and psychological profile data via an external web page, where basic tools allow them to compare their performance and dispositions with that of the player pool, or segments of the pool. |
| UC-2.3 | View behavioural and socio-psychological profile | Via the HUD, the player can toggle the display that shows which friends have signed up to the game and who is currently playing. It shows the name and profile photos of the friends, together with their online status. The contextual menu (right mouse button) gives the user various options to interact with the friend (see UC-4.3). |
| UC-2.4 | See friends | In addition to the using the left mouse button to navigate and interact with the game, players can get information and interact via the context menu (right mouse button). The context menu provides a range of menu options depending on the current location and context. |
| UC-2.5 | View context menu | Players can display their current inventory via the HUD. The inventory is multi-tabbed in case there exist more than one category. The inventory contains information on |

| UC-2.6 | View inventory | Create a tool to generate requirements and design specifications for financial capability games, based on the idea underlying UC-1.2

A first version of the tool is described in section 8 below.
material assets, the avatar’s knowledge, skills, tools and strategies. In exchange and other interactions, the user can select items from the inventory.

Assets, knowledge, skills, and so on can be acquired from a market place. The market place displays all items, those that the player can acquire, and those that are still locked. Players can acquire the items that are unlocked if they get sufficient money, experience points, knowledge or satisfy any other condition for access.

Users can customise their avatar by selecting basic profile data, aesthetics and look & feel, and personality and other individual characteristics. Players start off with very little possibility of influencing the character of the avatar, since gameplay is about controlling one’s avatar.

All navigation and interaction is done with the left mouse button, with a few exceptions. The avatar is guided by a path finding algorithm, so users only need to click on an allowable location to indicate to the avatar to move to that location (it won’t necessarily do this though!). Overlay, semi-transparent symbols indicate the left mouse button action as a function of the position of the mouse cursor, and thus increase navigability. Interactions avatar-NPCs and avatar-objects is done using the same logic. The right mouse button extends the available options with less often used actions and commands.

Other players (= friends) are represented in the game as identifiable NPCs (or better ‘proxy-avatars), independently of whether they are online or not. An interaction is triggered either by a friend, the game engine, or the player clicking on a friend. Interactions are scripted and can unravel relatively automatically. Interactions include talking, exchanging things, paying money, and helping out. A different type of interaction available to the player that doesn’t require direct interaction with the other player is to put challenges to a friend. These can consist of quizzes, minigames, or even changes to the goal structure (see more on this in section 8 where we discuss the game specification tool).

The player can interact with their friends through their avatar. Typically, she does this by triggering an interaction sequence. Consistency in interaction sequences has to be enforced by the game engine. If the friend is online, they will typically be in a different location doing different things, and hence sequences that fit the player’s context don’t necessarily fit their friend’s. One way to reduce the problem of proximal-distal communication (and exchange if we add a fantasy element of virtual exchange of material objects) is that while on one game client, communicative interaction is face-to-face, on the player’s friend’s game client, the communication is established through a (mobile) phone call. There are still considerations of consistency here, but the space of allowable interactions can be increased this way.

The Facebook social graph API allows the game engine to
### 4.2.2 Interact with other players through networked minigame play

| UC-4.2.3 | Social network | Interact with other players through networked minigame play | Users can also directly play with friends via minigames that are networked. Running such a game entails spawning a new scene which synchronises, via Unity3D’s networking feature, game state between all players. Since this is largely a minigame feature (and thus sits outside the fincap game system boundary), we will not discuss it here further. |

| UC-5 | Social network | Provide access to social graph | Facebook provides a series of APIs to developers through which information can be accessed and also written to Facebook’s social graph. This use case is thus fully met by D-SY-1.2. |

| UC-6 | Minigames | Plug into the game | Minigames that are developed outside this specification should be able to be ‘plugged’ into the fincap game and do their magic. |

| D-1.UC-6 | Database server | Store/retrieve game session | When the player starts up the game, the game engine has to retrieve the last session from the database. Data stored in the session includes that of the player’s game settings for the player and the snapshot of the pruned game state. |

| UC-8 | Database server | Synchronise game state | The game engine writes the pruned game state periodically to the database, so that the game clients can synchronise on a subset of that state. If the player goes offline, the last stored pruned game state persists and can be retrieved in the next session. |

| UC-9 | Database server | Record behaviour and psychological data | The game engine (or better, the xDelia extensions) may compute various behavioural and psychological measures and aggregates that are stored in the database. This can be a one-off computation, or time series data. Players have a few options to select the data they want to have stored in the database or which they want to download as e.g. a .csv file. |

### 3.6.6 Non-functional requirements

The non-functional requirements for the game system that can plausibly be defined at this stage are outlined in Table 3.4 below.

**Table 3.4 – Non-functional requirements**

| R-NF-1.1 | Look and feel | The game should adopt the look and feel of games that share core features with it. Specifically, we are looking at Zynga’s Facebook games and at the Sims to adopt aesthetic and usability features that work well. |

<p>| R-NF-1.2 | More generally, the appearance and style of the game should meet young people’s |</p>
<table>
<thead>
<tr>
<th>R-NF-1.3</th>
<th><strong>Usability</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The game should have a simple interface and intuitive, again in line with similar kind of online games. Navigation and interaction patterns from successful games are usually good candidates for adoption.</td>
</tr>
<tr>
<td>R-NF-1.4</td>
<td>Text, also that represented by graphics, should be language-labelled to support a basic internationalisation feature.</td>
</tr>
<tr>
<td>R-NF-1.5</td>
<td><strong>Performance</strong></td>
</tr>
<tr>
<td></td>
<td>Synchronising state information via the database should be kept at a minimum.</td>
</tr>
<tr>
<td></td>
<td>Requirements of availability, reliability, and scalability are extremely important for a large user base. However, given our resources and objective, we will not consider these factors. Since we do not use networking functionality, the only real bottleneck, apart from possible bugs in the software code, is the database and the game download. If we had to cope with thousands of players, then a solution might be to (1) create multiple databases on Amazon’s elastic cloud, and (2) to stream the game from EC3 servers. All this could deal with high and variable loads, but of course there would still be the problem of maintaining a game with several thousands of players.</td>
</tr>
</tbody>
</table>
3.6.7 Online fincap game (game-centric view)

The technology and conceptual elements that make up the fincap game are shown in Figure 3.23 below.

![Figure 3.23 – Game-centric view of the online financial capability game system](image)

**Internal technology** – core technology components of the online game
- Database servers: where persistent session data, pruned\(^2\) game state data, and behavioural and psychological data from players
- Game engine: the part of the game where most game logic and graphics computations take place (except computations that are related to the subject matter, the application domain (i.e. financial capability), and any world, avatar, and AI model that isn’t part of the standard package
- Domain models and AI: all models and artificial intelligence that isn’t part of the game engine, for example an avatar decision model, pathfinder, or market model

**External technology** – core technologies that are provided by third parties and thus are external to the online game
- Social network: a social site with an API that allows us access to the social graph, e.g. friends, groups, photos, events, and so on
- Mini-games: small games developed as part of the exploratory prototyping, and that can be plugged into the online fincap game

---

\(^2\) Pruned game state refers is that part of the game state that is fed into the models and persisted across sessions. Only a small part of the game state is important for the core gameplay.
• Sensors: not part of Y2 learning intervention, but some trialling with commercial sensors is planned
• Game admin: not part of Y2 learning intervention; to manage the players and monitor and maintain the game

**Conceptual** – content, models, and educational and pedagogical frameworks

- Financial capability – educational frameworks (e.g. BSA/FSA) and evidence-based analyses of needs, behaviours, barriers, and so on (e.g. the ‘requirements base’ that we proposed in D10-3.2)
- Pedagogy – learning models and approaches that are suitable in the context of learning design for serious games; guiding principles and design elements
- Behaviour research – evidence and models from decision science, psychology, sociology and so on, e.g. to construct an avatar behaviour or decision model

### 3.6.8 High-level architecture

Figure 3.24 below shows the game system of the online fincap game with its various components, and additionally the most important external software components and the three user types.

**Game system** – The core of the game system is Unity3D, a popular multiplatform development environment that comes with a 3D game engine and a game editor, offering a range of powerful feature off-the-shelf, including networking, physics, audio, native-code scripting, scene construction, asset pipeline and so on.

![Game system (top level)](image)

**Figure 3.24** – Online financial capability game system
Since only foundational game objects and interactions are provided by Unity3D, we have to complement the object model using our own xDelia collections (see e.g. the classes in the Game director collection shown in Figure 3.25). The main collections are: domain model (financial capability related objects such as selector of feasible behaviours), avatar model (mostly decision and behaviour models), world model (for instance a market or simple economy, either agent-driven or data generation process), game logic (rules of the game, goal hierarchy, level structure, and so on), and pruned game state (the game state that ‘makes a difference’ and is stored in the state database).

We use two databases (DBs) as a repository for game and other data. The first DB stores persistent (inter-session) data and the pruned game state. Inter-session data are maintained even though the player is offline. Game state data (pruned, i.e. data that is gameplay-relevant) is stored because this is our robust and cheap solution to circumvent networking issues. The second DB stores behavioural data emanating from gameplay for further analysis and research. Where behavioural data has to be fed back into the game, we store it in the pruned game state DB.

![Game director classes](image)

*Figure 3.25 – Online financial capability game system*

### 3.6.9 Summary

**Social gaming** – We wanted to develop a multiplayer game with a social component because the stakeholder requirements reported in D10-3.2 suggest that this could be (1) an effective route to behaviour change and (2) an effective delivery mechanism. The game is integrated with Facebook, using the Graph API that provides access to users’ friends, pages, events, profiles, groups, and so on. Some games on
Facebook have more users than the most popular social sites such as Twitter, even though they are relatively simple in game logic and game art. The fact that you can find your friends in the game is hugely attractive, and therefore we have included features that leverage this.

The player accesses the game via a web player, which functions as her game client. Multiplayer functionality usually requires a game server keeping game state synchronised across the different clients. Networking is a highly complex affair, and even though Unity3D comes with full support to build networked games, this is only a recently added feature and we’ve encountered some problems with it. We have chosen a simpler way that uses a database to store the most relevant game state data. Instead of syncing game state real-time between players, we simulate other-player actions in the game client, which will not affect the core motivator of friend presence in the game.

**Minigames and experimental prototypes** – We have created a simple way to incrementally add the experimental prototypes that are developed by BTH for the core constructs of impulsivity. Experimental prototypes are developed in Unity3D as well, and we specify a procedure and interface for game developers to implement these prototypes in a way that they can be imported as scenes into the online fincap game.

**User types** – The principal mode of use of the online financial capability game is by communities of players. There are a number of other ways how the online game and the mini-games can be used. First, the game can be played by a single player; playing with friends is optional in this respect. Even if you play with friends, the nature of the game (engagement over longer time periods) makes it unnecessary that they are online. Even friends who are offline live in the game in some form. Second, the inline game and especially the mini-games can be used by educators to enrich their own educational content or curricula. Third, behavioural data gathered from gameplay in real-time offers a good opportunity for Internet-based behavioural research.

Since WP3 has terminated we will during the forthcoming months wrap up the game to get a feeling of a complete but limited game that can be demonstrated as a proof of concept game.
4 Conclusions

The design process during the past year has consisted of a chain of activities. The workshops, collaboration between the physical meetings and the prototypes have all contributed to the progress. The overall conclusion from the design processes is that we have gained extensive experience in how to carry out participatory design in a multidisciplinary research environment. We have also experienced how to let participatory design permeate the whole design process from game design to implementation and evaluation. It has also been apparent that a close and continuous collaboration between the work packages are required to make games relevant for the application areas.

During year two we have started to develop what we think is a viable learning intervention for WP3, and even though WP3 is terminated we perceive MINDswap to be able to act as a demonstrator for how fincap games can look like in opposition to traditional approaches to fincap education.

In WP2 we have three different games (and in year tree a couple of variations will be developed) that have shown potential in the play testing. The games will be developed further and incorporated into the overall learning intervention during year three. This document has reported on evaluation efforts during the development process to guide the development. The impact on emotion regulation and how the disposition effect is handled will be thoroughly evaluated in studies starting in May at EUR and FZI.
5 Appendix A – Two Index Game

COMPUTATIONAL MODULES FOR TWO INDEX GAME

JEFFREY TODD LINS
JUNG TAY YEE
SAXO BANK A/S

ABSTRACT. We give descriptions for the generation of two time series for the Two Index Game and discuss computational methods for the detection of the disposition effect for incorporation into game play and post-game analysis. We provide pseudo-code for all computations corresponding to forthcoming C++ programming modules for reference or direct implementation.

1. INTRODUCTION

The Two Index Game (TIG) is a serious game under consideration for game-based learning within intervention techniques related to the role of emotions and emotional regulation in financial decision making. In particular, it emulates in a non-specific manner a deliberative process within investment and trading and incorporates feedback from exhibited cognitive biases into game play in real-time.

Game play includes decisions about holding positions with reference to changes in one time series given possibly predictive information observable in a second time series. Difficulty in the game is not only a function of the quality of predictive information contained in the relationships in and between these signals, but also of the level of indication of cognitive biases, such as the disposition effect.

In this short paper, we present formulations for the construction of signals for game play and related configurable parameters and for the measurement of the disposition effect, based on its description in the relevant literature. These formulations correspond to forthcoming computer code for C++ modules for these computations.

2. TWO INDICES

In TIG, we consider two stochastic processes $X(t)$ and $Y(t)$, each of which are described by the following equations,

\[ dX(t) = \mu_X(t)dt + \sigma_X dz_X, \]
\[ dY(t) = \mu_Y(t)dt + \sigma_Y dz_Y, \]

where $dz_X$ and $dz_Y$ are two independent Brownian motions. We design $X(t)$ as the predicted index and $Y(t)$ as the predictor index such that a player may interpret some information content of $Y(t)$ in order to predict innovations of $X(t)$ on the basis


Key words and phrases. Two Index Game, Disposition Effect.

Archive No: WP1.310810.2 XT O.
of that information and of that contained in $X(t)$ itself. For this purpose, we design $X(t)$ to simulate $Y(t)$ except for some intervals of deviation. Additionally, the two series $X(t)$ and $Y(t)$ are designed with time dilation such that the any predictive information in $Y(t)$ arrives in advance of that of $X(t)$ – but not uniformly in time. The predictive information is encoded in drifts $\mu_X$ and $\mu_Y$, while $\sigma_X$ and $\sigma_Y$ are added for the purpose of obscuring this information commensurate with the degree of difficulty.

The general scheme for generating the two time series $X(t)$ and $Y(t)$ are summarized as follows (See algorithm 1).

- Initially, we randomly generate $\mu_Y(t)$ and add noise $\Delta Y$ to generate the $Y(t)$ index. The noise level $\sigma_Y$ can be adjusted to change the level of difficulty.

- For the entire time horizon, $T$, we choose a collection of intervals, such that within any of these chosen intervals $\mu_X$ deviates from the prediction of $\mu_Y$, and outside of those intervals, $\mu_X$ simulates precisely the prediction of $\mu_Y$. Let us define the collection of time intervals as $T_I$ when $\mu_X$ deviates from the prediction of $\mu_Y$ and correspondingly define $T_S$ as when $\mu_X$ simulates the prediction of $\mu_Y$. In order to choose time intervals, we use a Poisson process method. Here Poisson events are the beginnings (events $I$) and endings (events $F$) of any intervals where $\mu_X$ deviates from the prediction of $\mu_Y$. Initially $\mu_X$ simulates the prediction of $\mu_Y$, but whenever events $I$ occur, then $\mu_X$ deviates from the prediction of $\mu_Y$, and whenever the events $F$ occur during these deviations, then $\mu_X$ simulates the prediction of $\mu_Y$ again.

The rate parameters for the Poisson events are $\lambda_I$ and $\lambda_F$, respectively. Hence, the average arrival time or the waiting time between the events will be $1/\lambda_I$ and $1/\lambda_F$. Then the ratio of the sizes of two sets $T_S$ and $T_I$ are

$$R(T_S, T_I) \approx \frac{\lambda_F}{\lambda_I}.$$ 

The number of disjointed intervals in the sets are

$$n(T_S, T_I) \approx \frac{T}{1/\lambda_I + 1/\lambda_F}.$$ 

$R(T_S, T_I)$ and $n(T_S, T_I)$ can be used to vary the difficulty of the game.

- The time dilation between $X(t)$ and $Y(t)$ are given by $S(t)$. In TIG, we design $S(t) = S_0 + \alpha \sin(\omega t)$ and thus vary the time dilation itself in time. $S_0$, $\alpha$ and $\omega$ are adjustable parameters.

- During the time in $T_I$, we set $\mu_X(t) = \eta \mu_Y(t - S(t))$, and during the time in $T_S$, we set $\mu_X(t) = -\kappa(t) \eta \mu_Y(t - S(t))$, where $\eta$ is the global scaling parameter which is positive and $\kappa(t)$ is a positive local parameter generated randomly.

After this scaling, $X(t)$ and $Y(t)$ will be related as $X(t) \sim \eta Y(t)$, given any noise and the deviation. Note that the scaling is applied to avoid the straightforward association of $X(t)$ to $Y(t)$. 

2

D14-4.2.3 – Game prototypes: design, implementation, and evaluation

© xDELLA Consortium – All Rights Reserved
During the deviation, the movement of $X$ is directed oppositely to the movement of $Y(t)$, in order to increase the obscurity of information in the signal to the player. Since the allocation of time intervals to the sets $T_S$ and $T_U$ is random, the player has to decide probabilistically whether to use the information of $Y(t)$ for the future prediction of $X(t)$.

- In the final step, we add noise. Again, the level of noise, $\sigma_X$ can be adjusted to vary the difficulty level of the game.

At each time step, the player is quoted with $X(t)$ and $Y(t)$ and her decision to buy and sell the index $X(t)$ is registered as the exposure $E(t)$, which therefore can have a value of 0 or 1. The total present value of her investment will be $I(t) = E(t) \times X(t)$. Her cash account is registered as $C(t)$. The total asset of the player will be $V(t) = I(t) + C(t)$.

![Graph](image1.png)

**Figure 1.** An example of two time series $X(t)$ and $Y(t)$. The time dilation is about 100. The graphical interface for game play may contemplate different representations for the series, as may be appropriate.

### 3. Disposition Effect

There are several ways to measure the disposition effect and one widely used method is to compute the proportions of realized gains and losses, here PGR and PLR, respectively [1]. The original PGR/PLR measure of disposition effect counts
the number of paper gains and paper losses and realized gains and losses over a
given period of time. This is practical if one measures the disposition effect for a
long time and the unit time interval of the trading data is a trading day. However,
in a real-time environment such as TIG, we may directly apply the holding-time
difference between gains and losses. This simple temporal measure is equivalent to
the PGR/PLR measure but seems more intuitive. Lastly, we present more advanced
measures, which may shed new understanding on the disposition effect.

3.1. PGR and PLR. PGR and PLR are defined as

\[
\begin{align*}
\text{PGR} &= \frac{\text{Number of Realized Gains}}{\text{Number of Realized Gains} + \text{Number of Paper Gains}}, \\
\text{PLR} &= \frac{\text{Number of Realized Losses}}{\text{Number of Realized Losses} + \text{Number of Paper Losses}}.
\end{align*}
\]

Typically, PGR and PLR are measured for a long term period, for example, a
trading year, and the gains and losses are registered each trading day. Note that
in (3.1.1), we are counting the number of events, not the amount of events.

For TIG, a player’s paper value change \( G(t) \) is given as

\[
G(t) = E(t)(X(t) - X(t_B)),
\]

where \( t_B \) is the time when the index was bought. If \( X(t) > X(t_B) \), then \( G(t) \) is
positive and the player is said to be making a paper gain, but if \( X(t) < X(t_B) \), then
\( G(t) \) is negative and the player is said to be making a paper loss. See algorithm 2.

Let us consider the case that the player makes a paper gain for a time interval
\( \tau_+^{(i)} = (t_p^{(i)}, t_q^{(i)}) \), where \( (i) \) indexes the interval after starting the game and \( t_p^{(i)} \)
and \( t_q^{(i)} \) are the starting time and the ending time of that gaining interval. The +
sign is used to indicate that the player is gaining for the period. Mathematically,
this means that \( G(t_p^{(i)}) = G(t_q^{(i)}) = 0 \) and \( G(t) > 0 \) for the time \( t \in \tau_+^{(i)} \). In this
case, the length of the holding time is \( H_+^{(i)}(\tau_+^{(i)}) = t_q^{(i)} - t_p^{(i)} \). The length of the
holding time for losses can be defined in a similar way.

Note that \( t_p \) and \( t_q \) are different from actual buy and sell times, since \( t_p \) and \( t_q \)
are registered not only by the actual buying and selling but also by changes in the
index. For example, if the player buys the index and initially makes a loss but
then after a while the index rises and subsequently he makes gains from holding
the index, then his holding time will be recorded first as the period of losses and
then second as the period of gains, separately.

We define the sets of holding time lengths for gains and losses,

\[
\begin{align*}
H_+(t) &= \{H_+^{(i)}(\tau_+^{(i)})| \tau_+^{(i)} \subseteq [0, t]\}, \\
H_-(t) &= \{H_-^{(i)}(\tau_-^{(i)})| \tau_-^{(i)} \subseteq [0, t]\}.
\end{align*}
\]

In the TIG environment, we are actually counting the total holding time of gains
and losses, since the number of paper gains or paper losses is ticked per every unit
time when the player is gaining or losing. Thus we get the following relations

\[
\begin{align*}
\text{Number of Paper Gains} &= \sum_i H_+^{(i)}(\tau_+^{(i)}), \\
\text{Number of Paper Losses} &= \sum_i H_-^{(i)}(\tau_-^{(i)}).
\end{align*}
\]

\[\text{(3.1.2)}\]
If PGR > PLR, then it is argued that the player suffers from disposition effect [1]. From the formula (3.1.1), this disposition effect condition can be rewritten into an equivalent form

\[
\frac{\text{Number of Paper Gains}}{\text{Number of Gains Realized}} < \frac{\text{Number of Paper Losses}}{\text{Number of Losses Realized}}
\]

(3.1.3)

Selling or realizing gains/losses enforces an end to a gaining or losing interval. Thus counting the number of realization corresponds to counting number of intervals for gains or losses. Combined with (3.1.2), the inequality (3.1.3) is equivalent to the following inequality

\[
M(H_+) < M(H_-),
\]

where \(M(Z)\) is the mean of the set \(Z\). Thus under a continuous real time environment, we do not need to calculate PGR or PLR, but instead we only need to compute mean holding times for gains and losses to see the presence of disposition effect.

A note about the subtlety for computing “Number of Gains/Losses Realized”: One way is to count the number of selling times \(t_s\). A selling time \(t_s\) is defined as the time when \(\Delta E(t_s) = -1\). If \(G(t_s) > 0\), then gains are realized, and if \(G(t_s) < 0\), then losses are realized. Another way is to count the number of events \(t_q\) that \(G(t)\) crosses the line where \(G(t) = 0\). As we mentioned in the paragraph above, this crossing could be caused not only by selling but also by index movement. Both measures are complementary but do not make a difference if small fluctuation effects caused by the small random movement are taken out.

3.2. Holding Time Difference. An interpretation of equations (3.1.3) and (3.1.4) is that the disposition effect is expressed as the difference of the mean holding time for the periods of gains and the periods of losses. To quantify this difference, we use the following definition of the disposition score \(D(t)\),

\[
D(i) = \begin{cases} 
\exp\left(\frac{1}{k}d(M_-, M_+) - 1\right) - 1, & d(M_-, M_+) > k, \\
0, & d(M_-, M_+) < k,
\end{cases}
\]

(3.2.1)

where

\[
d(M_-, M_+) = \frac{M_- - M_+}{(M_- + M_+)/2}.
\]

Here \(M_+(t) = M(H_+(t))\) and \(M_-(t) = M(H_-(t))\).

Since there is no scale in the distribution of \(H_-\) and \(H_+\), we have to adjust \(k\) manually. As we discussed in subsection (3.1), the small but nonzero \(k\) will also serves to take out noise effects. If the mean holding time for a losing period is significantly longer than the holding time for a gaining period, such that \(M_- - M_+ > k(M_- + M_+)/2\), then this may be considered as an indication of disposition effect. See algorithm 3.

The mean is a convenient measure to represent a set, but we can use other statistical measures like the median or maximum of sets \(H_+\) and \(H_-\).

3.3. Gain/Loss Weighted Disposition Score. In the sections above, we have only considered the amount of holding time. The relationship of the amounts of gains or losses to the indication of disposition effect has not been included into the formulation. However, it will be interesting to consider a gain/loss weighted disposition score, since disposition effects are maximally represented for large amounts of losses/gains. For this, we introduce the concept of total paper gain/loss.
For the period of gains \( t \in \tau^{(i)}_+ \), the total weight \( Q_+ \) are defined as
\[
Q_+^{(i)}(\tau^{(i)}_+) = \int_{\tau^{(i)}_+} G(t).
\]

We can define \( Q_- \) and \( W_- \) for losses in a similar way. The disposition score (3.2.1) can be reformulated using \( Q_+ \) and \( Q_- \) instead of \( H_+ \) and \( H_- \). If we use \( M_+(t) = M(Q_+(t)) \) and \( M_-(t) = M(Q_-(t)) \) for (3.2.1), then what results we define as the gain/loss weighted disposition score.

3.4. **Gain/Loss vs Mean Holding Time.** It is known that some of human dynamics show heavy tails in the temporal distribution [2]. The probability distribution function of the heavy tailed dynamics is
\[
P(i) \sim t^{-\alpha},
\]
with some universal exponent \( \alpha \). Researchers have found \( \alpha = 1 \) for email reply and web browsing times, and \( \alpha = 3/2 \) for snail mail replies and financial trading [3, 4]. However the exponent \( \alpha \) has been computed without considering other variables such as levels of gains and losses when considering financial markets, or for emails, metrics for the importance of email content [2, 3, 4]. In the study of disposition effect, it would be interesting to see whether the holding time distribution shows heavy tails and whether the the universal exponent \( \alpha \) is affected by the amount of gain or loss. In the case that a player suffers from the disposition effect, then the tail of the distribution will be heavier for the losing period, giving the inequality \( \alpha(loss) < \alpha(gain) \).

If the average gains \( W_+ \) are defined as
\[
W_+^{(i)}(\tau^{(i)}_+) = \frac{Q_+^{(i)}(\tau^{(i)}_+)}{H_+^{(i)}(\tau^{(i)}_+)},
\]
The probability distribution function will take the form of
\[
P(H, W) \sim H^{-\alpha(W)}.
\]
The computation of \( \alpha \) is straightforward when the holding time \( H \) and the average gain/loss \( W \) are given.

**References**


**Quantitative Analysis, Saxo Bank A/S, Philip Heymans Allé 15, 2900 Hellerup, Denmark**

E-mail address: jtu@saxcobank.com, juy@saxcobank.com
Algorithm 1 Generate time series $X[t]$ and $Y[t]$

$T$ : total play time
$T_d$ : amplitude of time dilation variation
$w$ : frequency of time dilation variation
$S_0$ : mean time dilation
$r_X \in [0, 1]$ : variation rate of $\mu_X$
$\lambda_f \in [0, 1], \lambda_f \in [0, 1]$ : deviation rate of $\mu_X$ from $\mu_Y$
$\eta > 0$ : proportion of $X[t]$ to $Y[t]$

$m \leftarrow 0$

for $t = 1$ to $T$ do

$q_x$ : a random number $\in [0, 1]$
if $q_x < r_X$ then
  $\mu_X[t] \leftarrow$ a random number $\in [-\mu_{max}, \mu_{max}]$
else
  $\mu_X[t] \leftarrow \mu_X[t-1]$
end if

if $m == 0$ then
  $q_i : a$ random number $\in [0, 1]$
if $q_i < \lambda_i$ then
    $m \leftarrow 1$
    $\kappa[t] : a$ random number $\geq 0$
    $\mu_Y[t] \leftarrow -\eta * \kappa[t] + \mu_X[t]$
  else
    $\mu_Y[t] \leftarrow \eta * \mu_X[t]$
  end if
else
  $q_f : a$ random number $\in [0, 1]$
if $q_f < \lambda_f$ then
  $m \leftarrow 0$
  $\mu_Y[t] \leftarrow \eta * \mu_X[t]$
else
  $\mu_Y[t] \leftarrow \mu_Y[t-1]$
end if
end if

$X[t] \leftarrow X[t-1] + \mu_X[t]$ + a Gaussian noise
$S[t] \leftarrow S_0 + T_d \sin(wt)$
$Y[t] \leftarrow Y[t-1] + \mu_Y[t + S[t]]$ + a Gaussian noise

end for
Algorithm 2 Compute paper gain $G[t]$, holding time $H$, total weight $Q$ and average gain $W$

$E[t]$ : exposure that takes 0 or 1
$X_B$ : buy price

$i \leftarrow 1$

for $t = 1$ to $T_M$ do

if $E[t] == 0$ then
$G[t] \leftarrow 0$
else
if $E[t] \neq 0$ & $E[t - 1] == 0$ then
$G[t] \leftarrow 0$
$X_B \leftarrow X[t]$
end if
else
if $E[t] \neq 0$ & $E[t - 1] \neq 0$ then
$G[t] \leftarrow X[t] - X_B$
end if
end if

if $G[t] == 0$ & $G[t - 1] \neq 0$ then
$W[i] \leftarrow Q/clock$
$H[i] \leftarrow clock$
$i \leftarrow i + 1$
else
if $G[t] \neq 0$ & $G[t - 1] == 0$ then
clock $\leftarrow 1$
$Q \leftarrow G[t]$
end if
else
if $G[t] \times G[t - 1] < 0$ then
$W[i] \leftarrow Q/clock$
$H[i] \leftarrow clock$
$i \leftarrow i + 1$
clock $\leftarrow 1$
$Q \leftarrow G[t]$
end if
else
if $G[t] \times G[t - 1] > 0$ then
clock $\leftarrow clock + 1$
$Q \leftarrow Q + G[t]$
end if
end if
end for
Algorithm 3 Compute disposition score $D$

$D[i]$: disposition score
$k > 0$: bound for nontrivial disposition score

for index $i$ do
    if $W[i] > 0$ then
        $H[i] \rightarrow H_+ :$ holding time during gain
    else
        $H[i] \rightarrow H_- :$ holding time during loss
    end if
end for

$M_- \leftarrow \text{mean}(H_-):$ mean holding time during loss
$M_+ \leftarrow \text{mean}(H_+):$ mean holding time during gain

$d \leftarrow (M_- - M_+)/(M_- + M_+)/2$

if $d > k$ then
    $D[i] \leftarrow \exp(d/k - 1) - 1$
else
    $D[i] = 0$
end if
6 Appendix B – Play testing Two index

The play testing took place at BTH with six participants. We had one person who has used professional investor systems so I'll write out his opinions for you since these probably are particularly interesting to you. We also asked some free form questions and I'll provide you with the most relevant answers (questions stated in bold):

How long before you figured out what the game was about?
- Three participants understood the goal of the game after the tutorial, and also which tools they were supposed to use in order to get there. Two of these had a problem to understand when they could and couldn't push the button in the beginning because the shading of the buttons were not apparent to them.
- Two required detailed instructions even after the tutorial, but still had trouble understanding the basic concepts (especially the "sell" option, since they hadn't bought anything to sell).
- One person could start to manage the game upon the last phase the person played, before that the person didn't have a clue how to do it, despite several detailed instructions.

Was something missing in the tutorial?
Four people thought the tutorial was fine. The other two thought it was a bit vague, the first person had trouble specifying exactly what was vague about it. The other person (the experienced investor) was upset about the prediction curve and that no further explanation existed for it.

What strategy did you use? Did you change it during playing?
Four people tried to make look at the curve and see when it was low, and bought stocks then. Also when they judged it to be high they sold. Two of these tried to bet small amounts of money, in order not to loose too much.
One relied purely on luck.
The last one (the investor) first though that it might be an randomly rescaled prediction curve, but couldn't really make that one out. Tried to find correlation differentials between the curves, sometimes it was a good prediction but sometimes worthless. Then tried to see if it was a randomly mirrored, but couldn't make that out either. In the end the person thought that it was numbers drawn from the same population with the same mean and standard deviation, but couldn't really use that information in order to make a profit.

Do you think you would have played the same way if you could gain money?
Four people thought they would have been more thorough if they could win money. One would have bet more in order to be able to gain more money.
The investor would have played the same way since the predictor was useless and he did the best he could.

Did you feel motivated the whole time? Why/why not?
Two people wasn't motivated because it was just frustrating and they felt stupid. Also, the money was worthless (you couldn't do something with them).
Two felt that the total score and the distance to optimal performance were really fun to have since they could really compete against the game.
Two (one of the investor) didn't like that the predictor was so hard to predict. But they would also like more features like goals and sub-goals, achievements, and perhaps some other features that would keep it more interesting.
General notes (one statement = one person)

1. The rescaling of the indexes in the beginning of the game was perceived as very annoying and would have been preferred as a set scale from the beginning.
2. Close position was handled like an “ok”-button. That is, when a stock should be bought the perception was that he had to press “Buy 1” followed by “close position” in order to confirm the purchase.
3. Frustration because the correlation between prediction and index couldn't be found.
4. No correlation backward of forward between the curves.
5. Interesting game for a non-gamer.
Appendix C – Game Experience Questionnaire, GEQ

Game Experience Questionnaire (Core)
The alternative answers are “Not at all”, “Slightly”, “Moderately”, “Fairly” and “Extremely”.

1 I felt content
2 I felt skillful
3 I was interested in the game’s story
4 I could laugh about it
5 I felt completely absorbed
6 I felt happy
7 I felt tense
8 I felt that I was learning
9 I felt restless
10 I thought about other things
11 I found it tiresome
12 I felt strong
13 I thought it was hard
14 It was aesthetically pleasing
15 I forgot everything around me
16 I felt good
17 I was good at it
18 I felt bored
19 I felt successful
20 I felt imaginative
21 I felt that I could explore things
22 I enjoyed it
23 I was fast at reaching the game’s targets
24 I felt annoyed
25 I was distracted
26 I felt stimulated
27 I felt irritable
28 I lost track of time
29 I felt challenged
30 I found it impressive
31 I was deeply concentrated in the game
32 I felt frustrated
33 It felt like a rich experience
34 I lost connection with the outside world
35 I was bored by the story
36 I had to put a lot of effort into it
37 I felt time pressure
38 It gave me a bad mood
39 I felt pressured
40 I was fully occupied with the game
41 I thought it was fun
42 I felt competent
8 Appendix D – System Usability Scale, SUS

The word system in the original version of the SUS is replaced by game system (Nacke & Schild, 2010)

A Likert scale from 1 to 5 (strongly disagree to strongly agree) is used.

1. I think that I would like to use this game system frequently.
2. I found the game system unnecessarily complex.
3. I thought the game system was easy to use.
4. I think that I would need the support of a technical person to be able to use this game system.
5. I found the various functions in this game system were well integrated.
6. I thought there was too much inconsistency in this game system.
7. I would imagine that most people would learn to use this game system very quickly.
8. I found the game system very cumbersome to use.
9. I felt very confident using the game system.
10. I needed to learn a lot of things before I could get going with this game system.
9 Appendix E

CONSENT TO PARTICIPATE IN RESEARCH STUDY

Play Testing of the Aiming Game

You are asked to participate in a research study conducted by Henrik Cederholm, Olle Hilborn and Petar Jercic, from the School of Computing at Blekinge Institute of Technology. Your participation in this study is completely voluntary. Please read the information below and you are welcome to ask questions about anything you do not understand, before you decide if you are going to participate or not.

If you suffer from epileptic seizures or game addiction you are requested not to participate in the study.

The facilitators, Henrik Cederholm, must explain the procedure to you before you sign the consent form.

Any information that is obtained in connection with this study and that can be identified with you will remain confidential and will be disclosed only with your permission or as required by law. Confidentiality will be maintained by only using the participant number when correlating the questionnaires with the data from the game sessions. The consent form will be archived safely.

You can choose whether or not to participate in this study. You may withdraw at any time without consequences of any kind. You may also refuse to answer any questions you do not want to answer.

If you afterwards have any questions or concerns about the study, please contact Henrik Cederholm (hce@bth.se)

I understand the procedures described above. My questions have been answered to my satisfaction, and I agree to participate in this study.

Printed Name of Subject   Participant number
_________________________  ________________________

Signature of Subject   Date
_________________________  _________________________
References


