

DELFT UNIVERSITY OF TECHNOLOGY

EINDVERSLAG

BACHELOR PROJECT

---

# Expect Health

---

*TU Coach*

Dr. ir. R. BIDARRA

*Teamleden*

Chantal ECKHARDT

Dion DE HOOG

*Opdrachtgever*

Prof. dr. A. EVERS

Sander VAN DEN OEVER

Shirley DE WIT

*Bachelor Eindproject coördinatoren*

M.A. LARSON & Dr. ir. F.F.J. HERMANS

1 juli 2015





## Voorwoord

Voor u ligt het eindverslag van ons Bachelor Project voor de Universiteit Leiden. Om het Bachelor diploma voor de opleiding Technische Informatica te krijgen wordt er verwacht dat de studenten een project uitvoeren voor een echte cliënt. Hierbij moeten zij hun geleerde vaardigheden in de praktijk brengen.

In ons geval hebben wij gekozen voor het project van de Universiteit Leiden. Dit project is een onderzoek van de afdeling van prof. dr. Andrea Evers. Samen met (o.a.) drs. Meriem Manaï en drs. Roel Hogervorst doen zij onderzoek naar de behandeling van chronisch zieke patiënten (specifiek, patiënten die net gediagnosticeerd zijn met reuma). Wij hebben gekozen voor dit project omdat zowel het psychologische aspect (de gedragsveranderingen) als de context ons interessant leek. Het project gaat ook daadwerkelijk gebruikt worden bij een psychologisch onderzoek. Dit gaf ons ook extra motivatie om dit project tot een goed einde te brengen.

Wij willen graag prof. dr. Andrea Evers bedanken voor de mogelijkheid om te werken aan dit interessante project. Daarnaast willen wij drs. Meriem Manaï, drs. Roel Hogervorst en de overige onderzoekers in Leiden bedanken voor hun hulp en bijdragen aan dit project. Binnen de TU Delft hadden wij uitstekende begeleiders, onze coach Rafael Bidarra en de begeleiders van het project Felienne Hermans en Martha Larson, die ons adviseerde waar nodig en die altijd voor ons klaar stond als wij dat nodig hadden.

Als laatste willen wij alle designers, overige ontwikkelaars, testpersonen, familie en vrienden bedanken voor hun bijdragen en feedback. Zonder jullie had het resultaat niet geweest wat het nu is.

Rest ons nog om te zeggen dat wij uitkijken naar het verdere verloop van dit project en de uitkomst van dit onderzoek.

*Chantal Eckhardt  
Dion de Hoog  
Sander van den Oever  
Shirley de Wit*

Delft, 1 juli 2015



## Samenvatting

Om de behandeling van recent gediagnosticeerde reuma patiënten beter te maken en ondersteuning te bieden aan een al bestaand programma genaamd E-COACH, is er een computerspel ontwikkeld. Binnen dit spel wordt geprobeerd om mensen, op een leuke manier, een gezondere leefstijl aan te leren. Het spel gaat gebruikt worden in een psychologisch onderzoek van de Universiteit Leiden. De gebruikers, bij wie de leeftijd zal variëren van 18 tot ca. 70 jaar, moeten het spel vier maanden lang, dagelijks, ongeveer 15 minuten spelen.

Alvorens het spel is gemaakt, is er een literatuuronderzoek gedaan. Hieruit is gebleken dat een verandering in gedrag plaats vindt als er een motivatie, mogelijkheid en trigger zijn. Een goed ontworpen game bevat onder andere uitdaging, motivatie, doelgerichtheid, beloningen en een 'fun-factor'. Het is belangrijk om te zorgen dat de duur van het experiment niet gaat leiden tot verveling bij participanten. Het gebruik van een avatar kan de gebruiker enthousiaster maken over het spel.

De gebruiker speelt met een avatar binnen het gezondheidscentrum Via Nova. Hier moet hij/zij elke dag oefeningen doen. Deze opgedeeld zijn in vier categorieën: slaap, leefstijl, ontspanning en immuun/toekomst. Deze categorieën zijn gekoppeld aan een kamer in het centrum, bereikbaar via de centrale binnentuin. Er is een scoresysteem ontwikkeld waarbij de voortgang (het spelen van minigames) in individuele kamers invloed heeft op de totaal-score. De toewijzing van de minigames aan de verschillende kamers gebeurt aan de hand van een ontwikkeld algoritme.

Bij het implementeren van het product is er steeds getracht om zoveel mogelijk rekening te houden met de onderhoudbaarheid van de code. Waar mogelijk zijn scripts hergebruikt en terugkerende objecten zijn gestandaardiseerd in zogenaamde 'prefabs'. Ook het testen is een belangrijk onderdeel bij het maken van software. Naast het testen met het UnTested framework is er een gebruikerstest uitgevoerd.

Dit project is een prototype voor het daadwerkelijke onderzoek. In de zomer zal het verder worden ontwikkeld tot een bruikbaar concept door het toevoegen van nieuwe/andere content.



# Inhoudsopgave

Voorwoord . . . . .	iii
Samenvatting . . . . .	vi
<b>1</b> <b>Introductie</b>	<b>1</b>
<b>2</b> <b>Probleemdefinitie</b>	<b>2</b>
2.1 Opdrachtgever en probleemdefinitie . . . . .	2
2.2 Requirements . . . . .	2
<b>3</b> <b>Research</b>	<b>7</b>
3.1 Reuma . . . . .	7
3.2 Evaluative conditioning . . . . .	7
3.3 Approach Avoidance Task (AAT) . . . . .	7
3.4 Games in de Gezondheidszorg . . . . .	8
3.5 Avatar . . . . .	9
3.6 Leeftijd bij Gaming . . . . .	11
3.7 Middelen . . . . .	11
<b>4</b> <b>Design</b>	<b>13</b>
4.1 Design van het spel . . . . .	13
4.2 Design scoresysteem . . . . .	15
4.3 Design van de minigames . . . . .	15
4.3.1 Meditatie . . . . .	16
4.3.2 Schaap . . . . .	16
4.3.3 Memory . . . . .	17
4.3.4 Catch . . . . .	17
4.3.5 Boodschappen . . . . .	18
4.3.6 Touwspringen . . . . .	18
4.3.7 Zoek object . . . . .	19
4.3.8 Quiz . . . . .	19
4.3.9 Approach Avoidance Task (AAT) . . . . .	19
4.4 Fontein . . . . .	20
4.5 Design van de tutorial . . . . .	21
<b>5</b> <b>Implementatie</b>	<b>22</b>
5.1 ISO 25010 . . . . .	22
5.1.1 Functionele geschiktheid . . . . .	22

5.1.2	Betrouwbaarheid . . . . .	22
5.1.3	Prestatie-efficiëntie . . . . .	23
5.1.4	Bruikbaarheid . . . . .	23
5.1.5	Veiligheid . . . . .	23
5.1.6	Uitwisselbaarheid . . . . .	23
5.1.7	Onderhoudbaarheid . . . . .	24
5.1.8	Overdraagbaarheid . . . . .	25
5.2	SIG . . . . .	25
5.3	Randomize algoritmes . . . . .	25
5.3.1	Toewijzen in minigame . . . . .	26
5.3.2	Toewijzen van minigames . . . . .	26
<b>6</b>	<b>Testen</b>	<b>29</b>
6.1	Test scripts . . . . .	29
6.2	Gebruikerstest . . . . .	29
6.2.1	Eerste indruk . . . . .	29
6.2.2	Beoordeling kamers . . . . .	31
6.2.3	Beoordeling minigames . . . . .	32
6.2.4	Algemene ondervindingen . . . . .	33
<b>7</b>	<b>Proces</b>	<b>35</b>
7.1	Communicatie . . . . .	35
7.1.1	Interne communicatie . . . . .	35
7.1.2	Extern communicatie . . . . .	36
7.2	SCRUM . . . . .	36
7.3	Taakbeheer . . . . .	37
7.4	Versiebeheersysteem . . . . .	37
7.5	Verloop van het proces . . . . .	37
<b>8</b>	<b>Discussie en Aanbevelingen</b>	<b>39</b>
8.1	Functionaliteiten . . . . .	39
8.2	Nieuwe minigames . . . . .	39
8.3	Design . . . . .	40
8.4	Context . . . . .	40
<b>9</b>	<b>Conclusie</b>	<b>42</b>
	<b>Appendices</b>	<b>45</b>



<b>A</b>	<b>SIG Feedback</b>	<b>46</b>
A.1	Eerste evaluatie . . . . .	46
A.2	Tweede evaluatie . . . . .	47
<b>B</b>	<b>Testcases</b>	<b>48</b>
B.1	Tutorial . . . . .	48
B.2	Linked scenes . . . . .	57
B.3	Minigames . . . . .	59
<b>C</b>	<b>Toekomstige minigames</b>	<b>61</b>
<b>D</b>	<b>Infosheet</b>	<b>64</b>
<b>E</b>	<b>Oorspronkelijke opdrachtomschrijving</b>	<b>65</b>
E.1	Opdrachtgever . . . . .	65
E.2	Context . . . . .	65
E.3	Doel . . . . .	66
E.4	Doelgroep . . . . .	66
E.5	Opdracht . . . . .	66
E.6	Extra informatie . . . . .	67

# 1 Introductie

Bijna twee miljoen volwassenen en ongeveer 3000 kinderen lijden in Nederlands aan reuma. Ruim de helft van deze mensen zijn tussen de 40 en 64 jaar<sup>1</sup>. Dit is een groot gedeelte van de Nederlandse inwoners. Universiteit Leiden gaat onderzoeken of ernstige reuma voorkomen kan worden als er vroeg genoeg wordt ingegrepen. Theoretisch gezien zou reuma genezen moeten kunnen worden als het nog in een beginnend stadium is. Hiervoor zou de persoon met reuma gezond moeten gaan leven, oefeningen doen en medicatie innemen. Universiteit Leiden heeft verschillende testgroepen om te kijken hoe effectief hun experimentele behandeling en medicatie is. De behandeling is een traject van 12 maanden, waarbij er 4 maanden gebruik zal worden gemaakt van het spel.

Universiteit Leiden wil hiernaast ook kijken of het helpt om “leuke elementen” toe te voegen aan het traject. Dat is waar de Technische Universiteit Delft komt kijken. Het leuke element moet een spel worden waarin de deelnemer het traject samen met een digitaal personage doorloopt. Dit spel moet de deelnemer een positieve ondersteuning geven. De verwachting is dat de deelnemers door het spelen van dit spel positiever in het traject staan. Daarnaast moet dit spel een gedragsverandering bewerkstelligen. De deelnemers moeten namelijk aangespoord worden om dagelijks hun oefeningen te doen en om gezonder te gaan leven.

Het spel moet dus gericht zijn op positieve feedback die de deelnemers stimuleren ook na het traject hun leefstijl gezond te houden. Het spel moet ze niet herinneren aan ziek zijn maar juist aan gezond(er) worden. De deelnemer moet dus voornamelijk beloond worden bij goed gedrag en zo min mogelijk worden gestraft.

---

<sup>1</sup>Gebaseerd op gegevens van het Reumafonds (<http://www.reumafonds.nl>)

## 2 Probleemdefinitie

In dit hoofdstuk wordt het probleem geanalyseerd en gedefinieerd. In subsectie 2.1 wordt het probleem geïntroduceerd. De requirements die volgen uit de probleemdefinitie zijn beschreven in subsectie 2.2. Subsectie 7.2 gaat als laatste in op de planning en beoogde werkwijze.

### 2.1 Opdrachtgever en probleemdefinitie

Universiteit Leiden, in het bijzonder prof. dr. A. Evers met daarnaast enkele PhD(-kandidaat) onderzoekers, gaat onderzoek doen naar de behandeling van chronisch zieke patiënten. In dit geval gaat het om een onderzoek met patiënten die net gediagnosticeerd zijn met reuma. Het onderzoek met de patiënten loopt een jaar. Er zijn drie groepen met deelnemers, waarvan een groep gebruik zal maken van de bestaande E-COACH-applicatie en een te ontwikkelen spel.

Het spel moet gedragsverandering stimuleren en motiveren tot het doen van oefeningen, waarbij het belangrijk is dat het spel als leuk wordt ervaren. De onderdelen in het spel focussen niet op het ‘ziek’ zijn, maar op het ‘gezonder worden/zijn’. Het spel richt zich op net gediagnosticeerde mensen met reuma en het spel zal worden gebruikt in een psychologisch onderzoek.

### 2.2 Requirements

Zoals uitgelegd in subsectie 2.1, is het spel dus bedoeld voor mensen die net de diagnose hebben gekregen voor reuma. De leeftijden kunnen variëren van 18 tot ca. 70 jaar, waarbij het gemiddelde rond de 45 jaar zal liggen. De gebruikers moeten dagelijks ongeveer 15 minuten spelen (dit kan enigszins variëren per dag). Het idee is dat door het dagelijks spelen van het spel, het gedrag van de mensen onbewust kan worden beïnvloed. Het spel moet dus uitdagend en uitnodigend zijn. Mensen moeten het spel gedurende de gehele periode van vier maanden blijven spelen. Na onderzoek is gebleken dat er niet voldoende tijd is om een spel te ontwikkelen dat vier maanden uitdagend, leuk en toch effectief blijft. Daarom is met de cliënt afgesproken dat er een versie van het spel wordt ontwikkeld gebaseerd op een onderzoeksperiode van vier *weken*. Er wordt dan dusdanig ontwikkeld dat dit eenvoudig op te

schalen is naar een versie die vier *maanden* bruikbaar is. Dit zal gebeuren door het toevoegen van extra content.

Binnen het spel moeten een aantal modules geïntegreerd zijn. Deze modules worden bepaald door Evers (et. al.) en zullen (deels) in lijn zijn met de reeds bestaande modules binnen E-COACH. In eerste instantie zouden de modules als volgt zijn: Ontspanning, leefstijl, immuunsysteem en slaap.

Mensen mogen (en kunnen) alleen aan dit onderzoek meedoen als ze in bezit zijn van een computer. Vandaar dat ervoor gekozen is om het spel te ontwikkelen voor de meest gangbare besturingssystemen voor onze doelgroep: Windows en Mac OS.

De requirements zijn opgedeeld volgens het zogenaamde MoSCoW-principe in Tabel 1 tot en met Tabel 4. Dit houdt in dat de requirements worden gecategoriseerd in; must-have requirements (requirements die sowieso in het eindproduct geïmplementeerd *moeten* zijn), should-have requirements (requirements die in het eindproduct geïmplementeerd *zouden moeten* zijn), could-have requirements (requirements die alleen worden geïmplementeerd bij voldoende tijd) en wont-have requirements (requirements die niet zullen worden geïmplementeerd, maar die wel een leuke toevoeging zouden zijn).

Tabel 1: Requirements volgens het MoSCoW-model (Must-have)

<u>MUST-HAVE REQUIREMENTS</u>	
<b>Platform</b>	Het spel moet ontworpen worden voor Windows en Mac OS.
<b>Avatar</b>	Het spel moet de mogelijkheid hebben een avatar te maken en deze aan te passen (haar, kleding, geslacht, etc.). De avatar moet steeds duidelijk in beeld zijn.
<b>Content</b>	Het spel moet minstens 1 maand speelbaar zijn.
<b>Interface</b>	Het spel moet een duidelijke interface en een positieve uitstraling hebben.
<b>Score</b>	Het spel moet op een positieve manier scores bijhouden en gebruikers motiveren.
<b>Dilemma's</b>	Het spel moet de gebruiker op willekeurige momenten voor een dilemma stellen waarbij de gebruiker de 'verstandige' keus moet maken.
<u>LATER TOEGEVOEGD</u>	
<b>Minigames</b>	Het spel moet verschillende minigames bevatten, waarvan een deel gerelateerd aan de verschillende modules. Het spelen van deze minigames beïnvloed de score.
<b>Uitstraling</b>	Het spel moet een vrolijke, positieve en toegankelijke uitstraling hebben. Gebruikers mogen niet het gevoel hebben 'opgesloten' te zitten.
<b>Approach Avoidance Task (AAT)</b>	Het spel moet de AAT bevatten (de AAT opzet wordt aangeleverd door Universiteit Leiden).

Tabel 2: Requirements volgens het MoSCoW-model (Should-have)

<u>SHOULD-HAVE REQUIREMENTS</u>	
<b>Gezondheidsbalk</b>	Het spel moet de score/gezondheidsverbetering duidelijk inzichtelijk maken (op een positieve manier).
<b>Audio</b>	Het spel dient toepasselijke/motiverende geluidseffecten en achtergrondmuziek te hebben.
<b>Content</b>	Het spel moet 2 maanden speelbaar zijn.
<b>Beloningssysteem</b>	Het spel moet de gebruiker belonen voor zijn/haar activiteiten.
<b>Interactie</b>	Het spel moet de mogelijkheid hebben om met virtuele personages te interacteren.
<u>LATER TOEGEVOEGD</u>	
<b>Coach</b>	Het spel moet de gebruiker helpen/sturen middels een virtuele 'coach'.
<b>Tutorial</b>	Het spel moet een tutorial hebben om nieuwe gebruikers op weg te helpen en wegwijs te maken in het spel.

Tabel 3: Requirements volgens het MoSCoW-model (Could-have)

<u>COULD-HAVE REQUIREMENTS</u>	
<b>Content</b>	Het spel moet 4 maanden speelbaar zijn.
<b>Interactie</b>	Het spel moet de mogelijkheid hebben om de interacteren met andere gebruikers.

Tabel 4: Requirements volgens het MoSCoW-model (Won't-have)

<u>WON'T-HAVE REQUIREMENTS</u>	
<b>E-coach</b>	Het spel moet een link hebben met de E-COACH.
<b>Social Media</b>	Het spel moet een link hebben met Social Media.
<b>Mobiele applicatie</b>	Het spel moet uitgevoerd kunnen worden op een mobiel apparaat; bijvoorbeeld als ANDROID / IOS applicatie.

## **3 Research**

In dit hoofdstuk zullen de resultaten van de literatuurstudie worden besproken. Er is een analyse van de context gedaan en relevante achtergrondinformatie is opgezocht.

### **3.1 Reuma**

Volgens het Reumafonds is reuma een verzamelnaam van meer dan 100 verschillende, meestal chronische, aandoeningen aan het bewegingsapparaat [Reumafonds, 2015]. De meest voorkomende vormen van reuma zijn artrose en wekendelenreuma. Ruim de helft van de mensen met reuma zijn tussen de 40 en 64 jaar. Waardoor mensen reuma krijgen is onbekend maar vermoedelijk spelen omgevingsfactoren en erfelijkheid een rol. Gevolgen van reuma kunnen pijn en stijfheid in gewrichten of spieren zijn. Mensen met reuma ondervinden een grote negatieve impact op de kwaliteit van leven. Een goede behandeling kan de klachten doen verminderen.

### **3.2 Evaluative conditioning**

Evaluative conditioning gaat over hoe je iets leuk of juist niet leuk kan vinden door middel van een associatie [Hofmann et al., 2010]. Dit kan bijvoorbeeld gebeuren wanneer een persoon je iets vervelends heeft aangedaan. Elke keer dat je de naam van die persoon hoort, krijg je een soort gevoel van afkeer. Je associeert deze naam dan met iets dat niet leuk is. Evaluative conditioning kan ook plaatsvinden in games, de term gamification wordt hier dan vaak voor gebruikt. Hierbij wordt de gebruiker positief beïnvloedt door onder andere het ontvangen van punten, het bereiken van nieuwe levels, het ontvangen van virtuele objecten en ruimtes, de leaderboards [Bunchball, 2010].

### **3.3 Approach Avoidance Task (AAT)**

De techniek behavioral design kan gebruikt worden om een bepaald gedrag aan te moedigen, of juist te ontmoedigen. Volgens Fogg vindt een bepaald gedrag plaats als er een motivatie, mogelijkheid en trigger zijn [Fogg, 2009]. In zijn model stelt hij dat gemotiveerde mensen vaker slagen (ook voor moeilijker taken) en dat minder gemotiveerde mensen sneller zullen falen (ook voor de makkelijkere taken). Bij Approach Avoidance Task wordt gemeten



of mensen eerder de neiging hebben om iets te benaderen (approach) of juist iets te vermijden (avoidance) [Watson et al., 2012].

Approach Avoidance Task kan gebruikt worden om overmatige drinkers te beïnvloeden. Bij een onderzoek werd een deel van deze drinkers getraind om foto's van alcohol te vermijden en het andere deel werd juist getraind om deze te benaderen. Dit werd gedaan door, met behulp van een joystick, foto's respectievelijk weg te schuiven of naar zich toe te trekken [Wiers et al., 2009]. Daarna werd een smaaktest gehouden, waarbij bleek dat de overmatige drinkers (die getraind waren om alcohol te vermijden) minder bier dronken dan de overmatige drinkers (die getraind waren om alcohol juist te benaderen). Vergelijkbare resultaten zijn gevonden bij onderzoeken met roken in plaats van alcohol [Watson et al., 2013]. Hieruit kan geconcludeerd worden dat het uitvoeren van de AAT gedrag kan beïnvloeden.

In andere studies (zoals de studie van Toril et al. [Toril et al., 2014]) zijn experimenten uitgevoerd waarbij gepoogd is het cognitieve vermogen (o.a. aandachtstraining) te stimuleren. Hierbij werden in trainingssessies de mensen achter games gezet of juist niet. Deze resultaten werden vergeleken. Het is gebleken dat de games wel degelijk invloed hebben (al ebt dit effect wel weg na verloop der tijd) op het cognitieve vermogen. Daarbij is ook opgemerkt dat de duur van de training / oefeningen van invloed was op het resultaat. Trainingen met een looptijd van 1-6 weken hadden een beter resultaat dan trainingen die 7-12 weken duurden. Dit had ermee te maken dat de participanten zich begonnen te vervelen. Het is dus belangrijk om te zorgen dat de duur van het experiment niet gaat leiden tot verveling bij de participanten.

### **3.4 Games in de Gezondheidszorg**

Wanneer je aan een ziekte lijdt, en al helemaal wanneer deze chronisch is, is het belangrijk om goed voor jezelf te zorgen. Het bereiken van een positieve gedragsverandering is hierbij dus erg belangrijk. Dit kan onder andere worden bereikt door het spelen van een health game, mits goed ontworpen [Brown et al., 1997, Lieberman, 2001]. Een voorbeeld van een health game is de game 'Re-Mission' voor jongeren met kanker. Uit een onderzoek met deze game bleek dat de patiënten die de game speelden een hoger level van chemo in hun bloed hadden, hun antibiotica consistentier innamen, meer

kanker-gerelateerde kennis hadden en ze lieten een snellere toename van eigen werkzaamheid zien [HopeLab, 2015].

Een goed ontworpen game bevat een aantal elementen. Namelijk interactie, immersie, uitdaging, motivering, doelgerichtheid, beloningen, sociale elementen en een ‘fun’-factor (de game moet leuk zijn). Daarbij moeten de doelen die je stelt in je game overeen komen met de te behalen gezondheidsdoelen. Wanneer een gebruiker niet op de gezondheidsaspecten let, moet hij/zij er tijdens het spel achter komen dat dit wel echt nodig is. Gebruikers proberen dingen uit en kijken wat er gebeurt, dus ze kiezen soms bewust voor de verkeerde keuze. Tijdens het spelen moet de gebruiker feedback krijgen en de mogelijkheid krijgen om te verbeteren. Wanneer mensen het spel spelen en ze vinden het leuk dan werken ze harder om hun einddoelen te bereiken. Het zijn of helpen van een karakter beïnvloed het zelfbeeld en de houding tegenover gezondheid. Des te leuker het spel, des te beter dit werkt [Lieberman, 2010a].

Informele health games, games die in een thuisomgeving gespeeld kunnen worden, moeten concurreren met andere vrijetijdsactiviteiten. Daarom moet de game een hoog entertainment gehalte hebben of handig zijn voor de gebruikers. Bij informele games is er meer ruimte in termen van kwesties, verhaallijnen, spelen met falen en sociale interacties zodanig dat de spellen leuk en spannend zijn om te spelen. Verder kunnen informele games gericht worden op een kleine populatie met specifieke en unieke eigenschappen [Lieberman, 2010b].

### **3.5 Avatar**

Een avatar is een grafische weergave van de gebruiker in een spel. Uit onderzoek, dat gedaan is op gebied de effecten en het gebruik van avatars in een virtuele wereld, blijkt dat de meeste mensen een avatar kiezen die menselijke kenmerken bevat [Conrad et al., 2010]. Games kunnen een hoop opties geven in hoe de avatar eruit ziet, waarvan sommige uiterlijke kenmerken niet eens mogelijk zijn om in het echt te veranderen. Zoals opnieuw een tiener zijn, groene huidskleur krijgen of opeens 3 meter lang zijn. Uit meerdere onderzoeken blijkt dat het zien van jezelf in third-person Point Of View (POV) in een game invloed kan hebben op hoe je je in het echt gedraagt. Hoe avatars ons gedrag kunnen veranderen in een virtuele omgeving heet het proteus ef-

fect [Yee, 2009].

Het blijkt dat als je vijf minuten een slechterik of een held speelt in een game, dat dit al effect kan hebben of je een onbekende juist straft of beloont. Er is een onderzoek gedaan waarbij deelnemers willekeurig ingedeeld werden om vijf minuten lang of Voldermort (boosaardige avatar), Superman (heldhaftige avatar) of een cirkel (neutrale avatar) te spelen [Yoon and Vargas, 2014]. Vervolgens moesten deze avatars vechten tegen hun vijand. Daarna moesten de deelnemers een blinde smaaktest doen. Hierbij moesten zij proeven wat zij hadden gekregen, waarna zij chocoladesaus of chilisaus door moesten geven aan de volgende deelnemer. Het bleek dat de deelnemers die Superman hadden gespeeld bijna twee keer zoveel chocolade saus dan chili saus doorgaven aan de volgende deelnemer en dat deelnemers die Voldermort speelden bijna twee zoveel chili saus dan chocolade saus doorgaven aan de volgende deelnemer. Hieruit valt te concluderen dat het karakter waar je mee speelt, invloed heeft op welke handelingen je verricht.

Ook de aantrekkelijkheid van de avatar is van invloed. Bijvoorbeeld: bij een onderzoek kregen deelnemers een lelijke of een knappe avatar en hadden ze contact met een (virtuele) onbekende [Yee, 2009]. Het bleek dat de deelnemers die een knappe avatar speelden dichter bij de onbekende liepen en ook eerder persoonlijke informatie verschaften dan de deelnemers die een lelijke avatar hadden. Een vergelijkbare situatie vond plaats toen de deelnemers of een avatar langer dan de onbekende kregen of een avatar kleiner dan de onbekende. Het bleek dat de deelnemers die een langere avatar hadden eerder agressief onderhandelden dan de personen die een kleinere avatar speelden [Yee, 2009].

Er zijn meerdere factoren bij avatars die het enthousiasme van de speler in games kunnen beïnvloeden. Een voorbeeld zit bij het perspectief. Bij first-person POV is het enthousiasme groter dan bij third-person POV [Lim and Byron, 2009]. Bij first-person POV is de game omgeving meer direct. Dit komt doordat in third-person POV je de avatar bedient, maar in first-person POV word je zelf juist de avatar (je kijkt door de ogen van de avatar). Het enthousiasme is groter wanneer de speler zelf een avatar mag kiezen in plaats van dat er een voor hem gekozen wordt [Lim and Byron, 2009]. Echter geldt wel dat in first-person POV het enthousiasme niet hoger wordt door de mogelijkheid te krijgen je eigen avatar te kiezen.

### 3.6 Leeftijd bij Gaming

Uit een onderzoek van Nielsen Games, uitgevoerd in opdracht van de ISFE (Interactive Software Federation of Europe) blijkt dat 40% van de 16 tot 49 jarigen tussen de 6 en 14 uur per week gamen. 81% van de ouders die gamen spelen samen met de kinderen [NVPI, 2008]. Er is ook onderzoek gedaan naar de leeftijdsgroepen bij het spel Everquest [Griffiths et al., 2004]. Bij volwassenen (tussen de 20 en 70 jaar) is er een hoger percentage van de spelers vrouwelijk, namelijk 20.4% in tegenstelling tot 6.8% mannelijke spelers. Alle leeftijdsgroepen speelden gemiddeld meer dan 20 uur per week, met een piek bij 20-22 jarigen (29.1 uur per week). Het grootste gedeelte (44.3% van de jongeren en 54.5% van de volwassenen) vindt vooral het sociale aspect belangrijk aan het online gamen. Volwassenen offeren vooral de “echte” sociale activiteiten op voor het spelen terwijl jongeren vooral school of werk achterwege laten voor het spelen.

### 3.7 Middelen

Indien er ontwikkelt moet worden voor mobiele apparaat, is het nuttig om de meest kale environment te gebruiken. Het gebruik van engines zoals Unreal en Unity zitten standaard al vol met functionaliteiten die niet benut zullen worden met een eenvoudig spel. De beste performance zal hoogstwaarschijnlijk zonder engines worden behaald, mits de technieken goed worden toegepast (efficiënte implementaties). Door gebruik te maken van een kale ontwerp environment zullen de spellen zo efficiënt mogelijk kunnen draaien, geoptimaliseerd voor de individuele platformen.

De deelnemers aan het onderzoek van Manai (et al.) moeten een computer hebben om mee te mogen doen met het onderzoek. Niet de volledige groep zal noodzakelijkerwijs beschikken over een mobiel apparaat. Het toevoegen van deze eis zou de selecte groep deelnemers alleen maar verder uitdunnen. Om deze redenen is er (in overleg met Evers (et al.)) voor gekozen om een spel te ontwikkelen voor de computer. Wij zullen hierbij primair ontwikkelen voor Windows en Mac OS, de grootste besturingssystemen<sup>2</sup>.

---

<sup>2</sup>**Besturingssystemen** (Statistieken over de meest gebruikte besturingssystemen en software) <https://www.netmarketshare.com/>

### **Unity 3D engine**

De Unity 3D engine heeft als voordelen dat het spel direct naar meerdere platformen is te exporteren. Het nadeel is echter dat er altijd een ‘core’ aanwezig is, functionaliteiten die we niet nodig hebben voor het spel. In dat opzicht lijkt het gebruik van Unity een flink nadeel. Ook mist Unity de ondersteuning van netwerken, wat bij het gebruik van externe data (database) niet praktisch is.

### **Unreal Engine, CryENGINE en Source**

De Unreal engine lijkt erg op de Unity 3D engine. Voor het ontwikkelen van een (relatief simpel) spel zoals dat nu beoogd is, maakt het niet veel uit welke van de engines gekozen wordt.

De opdrachtgever heeft aangegeven dat er gefocust worden op het ontwikkelen voor de computer (Windows of Mac OS). Daarbij moet er wel de mogelijkheid zijn om vanuit dat punt eenvoudig een mobiele Android / iOS applicatie te maken. Doordat contacten van het ontwikkelteam ook Unity gebruikten, de vele beschikbare tools en de gevonden (duidelijke) documentatie is ervoor gekozen om het spel te ontwikkelen met de Unity Engine.

## 4 Design

In dit hoofdstuk zal worden besproken hoe het spel is opgezet. Het design van het spel zal worden besproken in subsectie 4.1, het ontwerp van het scoresysteem in het spel in subsectie 4.2 en het design van de minigames in subsectie 4.3.

### 4.1 Design van het spel

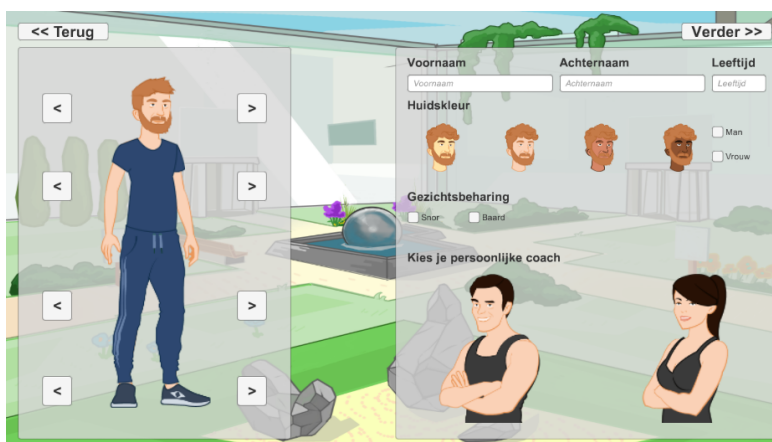
Het spel is gericht op een doelgroep van 18- tot 80-jarigen, van wie de gemiddelde leeftijd ongeveer 45 jaar is. Het is dus belangrijk dat het spel voor alle leeftijdsgroepen duidelijk is. Het is hierbij belangrijk om na te denken over hoe het spel voor iedereen leuk is, maar niet te moeilijk. Denk hierbij bijvoorbeeld aan de reactietijd van mensen; een 80-jarige heeft over het algemeen een veel trager reactievermogen dan een 18-jarige. De spellen kunnen dus niet al te snel en onoverzichtelijk zijn. Daarnaast moet het spel wel voor alle spelers leuk en uitdagend zijn; ouderen vinden vaak een spel als hartenjagen leuk terwijl dit minder in de smaak valt bij jongeren. Er moet dus een juiste mix worden gevonden om het spel uitdagend, overzichtelijk en leuk maken.

Het spel wordt gebruikt door mensen die meedoen aan het onderzoek van Manaï (et al.). De onderzoeksgroep heeft benadrukt dat het belangrijk is dat het spel de spelers een positief gevoel geeft. Het spel moet de deelnemers het gevoel geven dat ze op de goede weg zijn. We hebben dus manieren gezocht om de spelers altijd te belonen. Het spel moet het effect laten zien dat de oefeningen zouden moeten hebben, namelijk vooruitgang. Om de spelers een goed gevoel te geven moet het spel een open, vrolijke en uitnodigende uitstraling hebben. De ruimtes moeten de spelers geen benauwd en/of opgesloten gevoel geven.

Het spel volgt hetzelfde traject als de deelnemer aan het onderzoek. De deelnemer moet 4 maanden lang oefeningen doen die ervoor moeten zorgen dat hij/zij gezonder gaat leven. De speler moet dagelijks oefeningen doen die zijn opgedeeld in de vier eerder genoemde modules / categorieën: slaap, leefstijl, ontspanning en immuun/toekomst.

De speler maakt eerst een avatar, dit is te zien in figuur 1. Het is de

bedoeling dat de speler zich gaat identificeren met de avatar. Deze avatar bevindt zich in een gezondheidscentrum om zo gezond mogelijk te leven. De avatar bevindt zich standaard in de binnentuin, het centrale punt van het spel en dit is te zien in figuur 2. Hier hebben we voor gekozen omdat een binnentuin een ruime en open ruimte is. Vanuit de binnentuin kan de avatar naar vier verschillende kamers. Elke kamer is gekoppeld aan een module. In deze kamers kunnen er minigames gespeeld worden.



Figuur 1: Avatarscherm



Figuur 2: Tuin

## 4.2 Design scoresysteem

Elke kamer heeft een eigen balk die de gezondheid in die module voorstelt. Naast deze balken is er een balk die de totale gezondheid voor stelt. Door het spelen van spellen gaat deze balk vol. Na drie keer een spel gespeeld te hebben in dezelfde kamer is de gezondheidsbalk van die kamer vol. Zodra deze balk vol is wordt deze geleegd. Tevens wordt de totale gezondheidsbalk wat meer gevuld. Op deze manier werkt de speler voor een volle (totale) gezondheidsbalk door middel van het vullen van de balken voor de kamers. Na een week (een maand in de uiteindelijke versie van het spel) is de totale gezondheidsbalk vol. Zodra de gezondheidsbalk vol is verdient de speler een uitje (naar andere locatie). De speler speelt hier een aantal dagen minigames in een ander thema. Als hij/zij terugkomt van het uitje is de gezondheidsbalk groter geworden. Dat houdt in dat hij/zij weer meer gezondheid kan verdienen. Op deze manier kan de speler alleen maar vooruit en gaat zijn totale gezondheid nooit achteruit.

## 4.3 Design van de minigames

De minigames moeten net zoals de rest van het spel vrolijk en uitnodigend zijn. Het is de bedoeling dat de meeste minigames een bepaalde vorm van gedragsverandering opwekken. De minigames zullen een voor een worden besproken. Er zal worden uitgelegd wat de gedachtegang achter de minigame is en wat de eventuele verwachte gedragsverandering is.

De cliënt heeft verteld dat de gebruikers verplicht zijn om het spel dagelijks te spelen. Het belangrijkste van de minigames is dus dat ze effectief zijn. Het plezier dat de speler aan de minigames beleeft komt hierna pas. Het blijkt echter dat plezier een belangrijk element is in het succesvol maken van een spel (zie sectie 3.4). Er is geprobeerd in elke minigame een element te verwerken dat gedragsverandering stimuleert, maar desondanks de minigames toch een leuke ervaring te laten zijn.



### 4.3.1 Meditatie

Deze minigame is gekoppeld aan de module ontspanning en simuleert een meditatieoefening voor de avatar. In deze minigame is het de bedoeling dat je gefocust blijft op je meditatie. Je stelt jezelf voor dat je in een rustig, bosrijk gebied bent en dat je hier mediteert. Terwijl je aan het mediteren bent komen er afleidingen die ervoor zorgen dat je je focus verliest. De afleiding is in dit geval een mobieltje dat naar de voorgrond komt wat vervolgens de avatar zijn focus doet laten verliezen. Zolang deze afleiding er is komt de echte omgeving, namelijk de ontspanningsruimte, weer naar voren. Zodra je op de afleiding klikt gaat deze weg en wordt de meditatieomgeving weer hersteld. Als de echte omgeving te veel naar voren is gekomen, verlies je je focus en bevindt je je weer in de originele kamer. Des te langer je de meditatieomgeving in stand weet te houden, des te beter is het resultaat.



Figuur 3: Meditatie

Het idee is dat mensen doorhebben dat ze zich af moeten sluiten voor afleidingen en dat ze zich focussen op hun meditatie (en zodoende zichzelf dus ontspannen). Mensen zouden doorkrijgen dat externe afleidingen niet thuishoren in hun meditatie en ze zouden hierdoor beter tot ontspanning kunnen komen.

### 4.3.2 Schaap

Deze minigame heeft een connectie aan de module slaap. Iedereen kent het concept wel van schapjes tellen om in slaap te vallen. In het spel komen er schapen, wolven of konijnen over een hek heen springen. Zodra er een schaap in beeld komt moet je hier op klikken en zodra er een wolf of konijn in beeld komt moet je juist niet klikken. Je krijgt een beter resultaat naarmate je op meer schapen hebt geklikt en geen andere dieren. Als achterwege wordt gelaten dat het tellen van schapjes niet schijnt te werken om in slaap te vallen, herinnerd het de meeste mensen



Figuur 4: Schaap

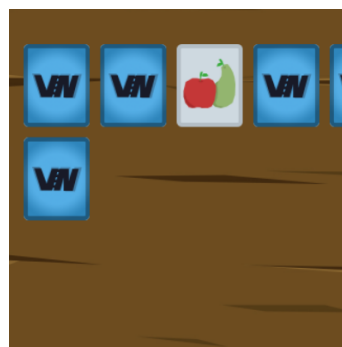
wel aan slaap. Hiernaast richt dit spel zich ook op concentratie, snelheid en herkenning. Doordat je van tevoren niet weet of er een schaap, wolf of konijn tevoorschijn komt moet je snel herkennen welk van de dieren het is en je moet er ook snel op klikken.

De opzet van deze minigame kan worden hergebruikt voor bijvoorbeeld de leefstijl kamer. Het kan zo aangepast worden dat er stukken fruit van een fruitmand naar de mond van de avatar vliegen, met af en toe wat ongezonds er tussendoor (denk aan een stroopwafel o.i.d.). De speler moet dat de ongezonde items wegklikken zodat de avatar alleen de gezonde items opeet.

### 4.3.3 Memory

Iedereen kent het spel memory, draai twee kaarten om en als het plaatje niet hetzelfde is draai je ze weer terug.

Deze minigame is gericht op geheugentraining en zou door de plaatjes een associatie maken met bijvoorbeeld een bepaalde module of gezonde voorwerpen. Doordat de speler steeds gezonde items bij elkaar zoekt krijgen deze items een positieve associatie.

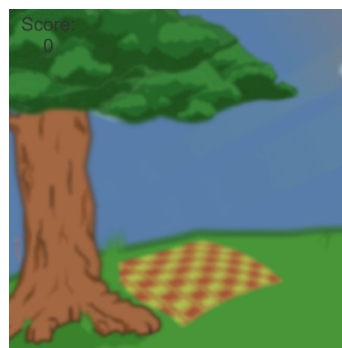


Figuur 5: Memory

Memory kan ook worden uitgebreid tot het zoeken van 2 items die bij elkaar horen, maar die niet identiek zijn. In dat geval moet de speler niet alleen de locatie van voorwerpen onthouden, maar de speler moet ook beredeneren welke gezonde items in dezelfde categorie vallen.

### 4.3.4 Catch

In deze minigame bevindt de speler zich in een weiland. De speler heeft een picknickmand die hij heen en weer kan bewegen met de muiscursor. Er komen allemaal gezonde en ongezonde voorwerpen van bovenaan het scherm naar beneden vallen. De speler moet de gezonde voorwerpen in zijn mand opvangen en de ongezonde



Figuur 6: Catch

juist niet. De speler krijgt punten voor de gezonde items en minpunten voor de ongezonde.

Dit spel is erop gericht dat de speler herkent welke items gezond zijn bij de picknick en dus ook buiten de picknick. Het idee is dat de speler bewuster gaat nadenken over wat hij/zij eet en dat hij/zij er op let dat hij/zij zo veel mogelijk gezonde, maar toch lekkere, dingen eet en drinkt. Het andere doel van het spel is dat de speler doorheeft dat de ongezonde items een minder goede optie zijn.

Dit spel is enigszins gebaseerd op de Approach Avoidance Task (AAT), waarbij je de goede afbeeldingen naar je toe trekt en slechte afbeeldingen van je af duwt. In subsectie 4.3.9 wordt hier verder op ingegaan.

#### 4.3.5 Boodschappen

Deze minigame laat de gebruiker een lijstje van etenswaren zien. De speler moet deze voorwerpen onthouden. Nadat de tijd is verstreken ziet de speler een tafel, met meer voorwerpen dan dat hij/zij moest onthouden. Hier moet de speler alleen de voorwerpen kiezen die op het lijstje stonden. In principe komt het er dus op neer dat je een boodschappenlijstje moet onthouden in beperkte tijd. Het lijstje met boodschappen wordt langer naarmate de moeilijkheidsgraad omhoog gaat.

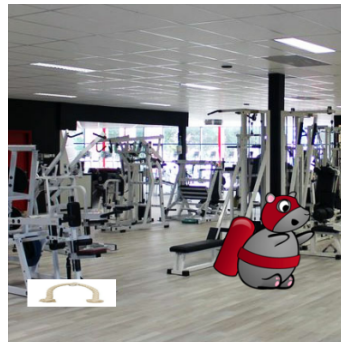


Figuur 7: Boodschap

De gedachte achter deze minigame is dat de gebruikers hun geheugen trainen en eraan gewend raken om met lijstjes werken. Het werken met lijstjes kan het dagelijks leven namelijk een stuk overzichtelijker maken. Een dag- of weekplanning, een boodschappenlijstje en andere situaties kunnen veel structuur geven aan de dagelijkse bezigheden. Deze structuur zorgt vervolgens weer voor meer rust.

#### 4.3.6 Touwspringen

Bij het touwspringen bevindt de avatar zich in een fitnessruimte en gaat hij/zij touwtjespringen. Er beweegt een touw heen en weer waar de avatar overheen moet springen. Als de speler klikt springt de avatar. De speler moet dus goed timen om over het touw heen te springen.

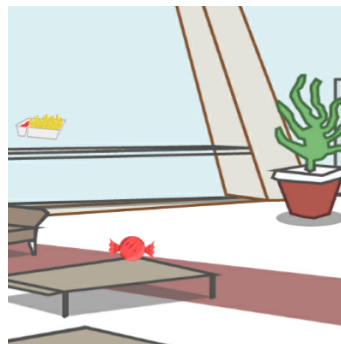


Figuur 8: Touwspringen

Aangezien het de bedoeling is dat de speler zich gaat identificeren met de avatar zou dit de speler moeten motiveren om zelf ook fysieke oefeningen te doen. Daarnaast is het een oefening om een bepaalde actie goed te timen (in dit geval het klikken met de muis).

#### 4.3.7 Zoek object

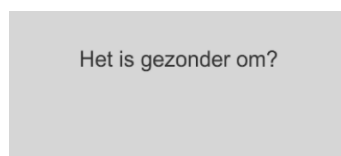
In deze minigame moet de gebruiker een gezond object zoeken in een kamer die gevuld is met ongezonde objecten. De speler ziet slechts een gedeelte van de kamer en moet de kamer doorzoeken om het gezonde voorwerp te vinden. Dit spel zou de gebruiker moeten motiveren om zelf ook niet voor het gemak van een ongezonde snack te gaan, maar juist verder te kijken voor een gezond alternatief.



Figuur 9: Zoek Object

#### 4.3.8 Quiz

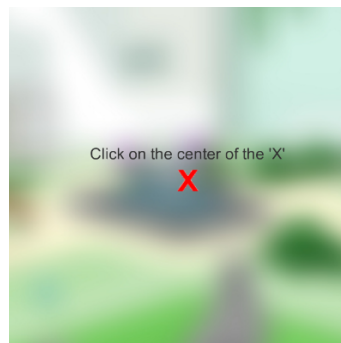
Dit spel is een quiz die bestaat uit vragen die zijn opgesteld door Manaï (et al.). Deze vragen moet de gebruiker bewust na laten denken over de keuzes die je op een dag maakt. Deze vragen gaan bijvoorbeeld over voeding, beweging of sociaal contact.



Figuur 10: Quiz

#### 4.3.9 Approach Avoidance Task (AAT)

Deze (voorlopig) laatste minigame is de minigame die al een psychologische onderbouwing heeft. De speler ziet steeds een afbeelding en moet kiezen of dit een goede afbeelding is of juist niet (in ons geval gezond of ongezond). Een goede afbeelding moet de gebruiker dan naar zich toe trekken, terwijl hij de niet goede afbeeldingen van zich af duwt. De theorie hier achter is dat als een speler een ongezond item vaak genoeg van zich af duwt dat hij/zij in het echte leven ook minder de neiging heeft dit voorwerp te gebruiken (zie sectie 3.3). Om een voorwerp van zich af te duwen kan de speler de muis naar de bovenkant van het scherm bewegen, om een object naar zich toe te trekken beweegt de speler de muis naar de onderkant van het scherm. Nadat de afbeelding ofwel weggeduwd of naar zich toe is getrokken, wordt de muis opnieuw gepositioneerd in het midden van het scherm. Dit gebeurt door het klikken op het rode kruis in het midden van het scherm.



Figuur 11: AAT

## 4.4 Fontein

Wanneer de gebruiker een minigame opnieuw wil spelen kan dat door in de tuin op de fontein te klikken. Er verschijnt een scherm waar de highscore van elke minigame staat, dit scherm is te zien in figuur 12. Elke minigame kan in verschillende moeilijkheidsgraden gespeeld worden.



Figuur 12: Speel oude minigames

## 4.5 Design van de tutorial

De eerste keer dat de gebruiker het spel speelt wordt de tutorial doorlopen. In de tutorial wordt uitgelegd hoe het spel werkt en een coach zal steeds vertellen wat de gebruiker moet doen. De gebruiker krijgt na het startscherm eerst een introductieverhaal over het spel, dit is te zien in figuur 13. Daarna kan de gebruiker een avatar maken en een coach kiezen. Na het aanmaken van de avatar komt de gebruiker terecht in de tuin. De coach vertelt dat de gebruiker naar de eerste kamer moet gaan. De gebruiker kan dan tijdelijk niet naar de andere kamers vanuit de tuin. Eenmaal in de eerste kamer aangekomen kan de gebruiker niet meer terug naar de tuin, de terug-knop is namelijk tijdelijk uitgeschakeld. Hierdoor loopt de gebruiker tijdens de tutorial een vaste route door het spel. In de eerste kamer moet de gebruiker een minigame spelen. De minigame is bepaald door het algoritme dat beschreven staat in subsectie 5.3.2. Na het klikken op Start in het startscherm is dit algoritme uitgevoerd en dus is voor elke kamer al de minigame voor die dag bepaald. Nadat de eerste minigame is gespeeld moet de gebruiker van de coach op de gezondheidsbalk klikken en de gebruiker zal zien dat de gezondheid van de eerste kamer omhoog gegaan is. Vervolgens keert de gebruiker terug naar de tuin om daarna in de andere drie kamers ook een minigame te spelen. De gebruiker keert dan terug naar de tuin en de coach vertelt dat de gebruiker klaar is met de tutorial.



Figuur 13: Introductiescherm

## 5 Implementatie

In dit hoofdstukken zal er toegelicht worden hoe er is omgegaan met softwarekwaliteit, de resultaten van SIG en zal er gekeken worden naar random algoritmes.

### 5.1 ISO 25010

Bij het implementeren van dit product is er gekeken naar code kwaliteit. Softwarekwaliteit wordt o.a. beschreven door de ISO 25010 norm [iso.org, 2011]. Binnen deze norm wordt er gefocust op acht kenmerken, namelijk: functionele geschiktheid, betrouwbaarheid, prestatie-efficiëntie, bruikbaarheid, veiligheid, uitwisselbaarheid, onderhoudbaarheid en overdraagbaarheid. Een goede softwarekwaliteit is in het belang van zowel de klant als van de developers. Goede kwaliteit zorgt er onder andere voor dat het makkelijker is functionaliteiten toe te voegen of aan te passen. Ook het vinden van een mogelijke bug is makkelijker bij goede software.

#### 5.1.1 Functionele geschiktheid

Bij functionele geschiktheid wordt er gekeken of het product voldoet aan de vooraf gestelde eisen. Gedurende het gehele project is er, door met Scrum te werken, continu gekeken naar welke functionaliteiten er gemaakt moesten worden en zo een prioriteitenlijst afgewerkt. Hoe de functionaliteiten hebben bijgedragen aan het voldoen aan de eisen is verder te lezen in het hoofdstuk 4: Design.

#### 5.1.2 Betrouwbaarheid

Om te zorgen voor een betrouwbaar product is het product getest onder normale omstandigheden, maar is er ook geprobeerd om het spel juist fout te laten lopen. Een voorbeeld hiervan is het scherm waar je een karakter kan maken, we hebben een aantal mensen gevraagd om een karakter te maken die niet gemaakt zou morgen worden. Zo kon er in de eerste een baard en snor geselecteerd worden voor een vrouwelijk karakter. Op deze manier hebben we zo veel mogelijk fouten verholpen. Meer over het testen is te vinden in het hoofdstuk Testen.

Bij het leeg maken van het register, waar de instellingen van de gebruiker zijn opgeslagen, of bij het verwijderen van configuratie mappen, zal het spel dit zelf opvangen. Helaas is het wel zo dat in het ergste geval, de gebruiker opnieuw moet beginnen. Dit kan in het spel niet voorkomen worden aangezien de gegevens lokaal worden opgeslagen. Een mogelijke oplossing hiervoor zou het online opslaan van de gegevens zijn, wat weer andere problemen met zich mee zou brengen.

### **5.1.3 Prestatie-efficiëntie**

Prestatie-efficiëntie is gemeten door te kijken naar de gebruikte processor capaciteit op een gemiddelde laptop. Hieruit bleek dat ons spel niet veel eist van de processor.

### **5.1.4 Bruikbaarheid**

Voor de bruikbaarheid is er gekeken of ons product goed en consequent werkt voor de gebruikers. Hiervoor hebben we gebruikerstesten gedaan en daarbij gefocust op het uiterlijk van ons product, de duidelijkheid van het spel en de bediening. De gevonden resultaten hiervan zijn te vinden in hoofdstuk 6: Testen. Of het product de beleving van de behandeling veranderd en of de leerdoelen van het product, het hebben van een betere leefstijl, behaald gaan worden, zal moeten blijken uit het onderzoek door Universiteit Leiden.

### **5.1.5 Veiligheid**

Voor veiligheidsredenen is ervoor gekozen geen internetconnectie op te zetten. Verder is het niet mogelijk zomaar gegevens aan te passen, zoals scores, in het register. Alle gegevens die het spelverloop kunnen beïnvloeden zijn versleuteld.

### **5.1.6 Uitwisselbaarheid**

Doordat er geen internetconnectie wordt opgezet is het niet mogelijk om gegevens uit te wisselen binnen het systeem. Het enige wat uitwisselbaar is, zijn de logfiles. Deze moet de gebruiker zelf versturen naar de therapeut zodat deze de voortgang in de gaten kan houden.



### 5.1.7 Onderhoudbaarheid

De onderhoudbaarheid van code is iets waar tijdens het implementeren extra op gefocust is. Zo is er gekeken naar het zo dynamisch mogelijk maken van het product, duplicatie voorkomen en de functionaliteiten splitsen. Het opleveren van een dynamisch product maakt het eenvoudig om wensen van de klant relatief snel en efficiënt door te kunnen voeren. Dit is zowel op grote als kleinere schaal doorgevoerd. In het framework is het geen probleem om, code technisch, een kamer meer of minder te maken of minigames toe te voegen of juist te verwijderen. Op kleinere schaal, de minigames, is er ook gekeken naar het dynamisch houden van de code. Het is vrij eenvoudig om game aspecten als duur en moeilijkheid aan te passen. Dit dynamisch maken en houden van de code zit hem er voornamelijk in dat er veel met variabelen gewerkt is. Door deze variabelen van duidelijke namen te voorzien is het eenvoudig om deze aan te passen, zonder dat er naar de methodes gekeken moet worden.

Zo is er om duplicatie en redundantie te voorkomen is er binnen Unity met Prefabs gewerkt. Deze Prefabs zijn objecten die hergebruikt worden. Zodra een zelfde object dus meerdere keren in ons programma voorkomt, is hier een Prefab van gemaakt. Zodra er iets in de Prefab wordt aangepast, wordt dit doorgevoerd in alle instanties van dit object.

Verder zijn functionaliteiten die meerdere malen toegepast worden in een apart script ondergebracht. Een voorbeeld hiervan is het script die zorgt voor een panel dat verschijnt na het spelen van een minigame. Zodra de minigame is afgelopen wordt er altijd hetzelfde script aangeroepen, waaraan de score wordt meegegeven. Dit script bepaalt vervolgens welke tekst er aan het panel moet worden toegewezen. Zodra er nu een aanpassing moet worden gemaakt aan dit panel kan dit eenmalig worden veranderd in het script en zal het toegepast worden voor alle minigames.

Om te checken of de code doet wat het moet doen moet er uiteraard getest worden. Hiervoor zijn zo veel mogelijk testen geschreven. Helaas is dit binnen Unity niet altijd een makkelijke taak, het is lastig tot onmogelijk om daadwerkelijke kliks op het juiste moment te simuleren. Ook overgangen tussen scenes is niet te testen. Maar omdat testen wel erg belangrijk is, zijn dingen die niet in code getest konden worden handmatig getest. Meer hierover in het hoofdstuk Testen.

### 5.1.8 Overdraagbaarheid

Het spel is zowel op een Windows computers als op een Mac te spelen. Doordat er in Unity gewerkt is, is het zelfs mogelijk om een deel van het werk mee te nemen voor bijvoorbeeld een tablet/smartphone versie. Hierbij moet wel opgemerkt worden dat het niet mogelijk is voor de gebruiker te beginnen op de computer en vervolgens verder te gaan op een mobiel apparaat. Dit omdat de spelgegevens lokaal worden opgeslagen. Het spel kan verwijderd worden door simpelweg de bestanden in de prullenbak te zetten.

## 5.2 SIG

Voor het beoordelen van onze code is deze tweemaal naar SIG <sup>3</sup> gestuurd. De feedback van SIG was erg positief en heeft ervoor gezorgd dat het team op scherp bleef staan, deze feedback is te lezen in appendix A.1 en A.2. Er is nog extra gelet op de indeling van methodes en scripts. Hierbij is de lengte van de methodes een goede indicatie geweest. Dit omdat lange methodes vaak het gevolg zijn van meerdere functionaliteiten in een enkele methode. Als richtlijn is aangehouden dat een methode onder de 12 regels code moest zitten. Zodra de lengte van een methode toch te lang bleek te zijn is hier nog eens extra naar gekeken hoe dit kon en of deze methodes niet uitgesplitst moesten worden. Door een methode echt maar een ding te laten te doen, wordt het overzicht behouden. Zo kan er bij een bug gerichter gezocht worden waar het fout gaat en kan de methode makkelijker hergebruikt worden. Het enige kritiek punt is gerelateerd aan het testen. Dit valt te verklaren doordat, zoals eerder al aangegeven, het door het gebruik van Unity soms lastig is om code goed te kunnen testen. Dit is opgevangen door handmatig testen, wat verder uitgewerkt is in onderstaande sectie 5.3, iets wat SIG niet mee kan nemen in de beoordeling.

## 5.3 Randomize algoritmes

Binnen het product is er een aantal keer gebruikt gemaakt van algoritmes die lijsten willekeurig indelen. Een voorbeeld hiervan is het toewijzen van kaarten in memory. Elke keer als de minigame memory wordt geopend zullen de kaarten zich op een andere positie bevinden. Ook het toewijzen van de

---

<sup>3</sup>SIG: <https://www.sig.eu/nl/>

minigames gaat random. Echter werken deze twee random algoritmes anders. Deze zullen in onderstaande secties worden uitgelegd.

### 5.3.1 Toewijzen in minigame

Voor het willekeurig toewijzen in een minigame, toewijzen van voorwerpen en posities, is er gebruikt gemaakt van het shuffle algoritme. Hiervoor is gekozen voor een moderne versie van het Fisher–Yates algoritme <sup>4</sup>. Wat er gebeurt is dat er een random object uit de array wordt gewisseld met een ander object, dit wordt dan  $n$  keer herhaald.

---

**Algorithm 1** Shuffle algoritme van Fisher-Yates

---

```
1: toShuffle  $\leftarrow$  Array met lengte  $n$ 
2: for  $i = 0 \rightarrow n$  do
3:   Object temp  $\leftarrow$  toShuffle[ $i$ ]
4:   int  $x \leftarrow$  random getal tussen 0 en  $n$ 
5:   toShuffle[ $i$ ]  $\leftarrow$  toShuffle[ $x$ ]
6:   toShuffle[ $x$ ]  $\leftarrow$  temp
7: end for
```

---

### 5.3.2 Toewijzen van minigames

Het toewijzen van de minigames gebeurt dynamisch en eens per dag. Hiervoor wordt er een lijst bijgehouden met de alle minigames. Het algoritme werkt met wegingen/kansen. Elk spel kan met een bepaalde kans geselecteerd worden. Deze wordt bepaald door de spelweging en de kamerKans. Wanneer een spel geselecteerd wordt zal de dag erna de weging nul zijn. Dit zorgt ervoor dat dit spel niet gekozen kan worden, mits de lijst van het aantal minigames voldoende is. De daaropvolgende dagen neemt de weging met de opbouw toe tot het maximum is bereikt. Voor reguliere minigames is dit maximum gelijk aan 0.9. Voor spellen die met een hoge regelmatig verplicht toegewezen moeten worden is dit maximum gelijk aan 1.0. De kamer kansen zijn bij elkaar opgeteld 1. Deze zijn toegevoegd omdat het mogelijk kan zijn dat een minigame interessant(er) is voor maar een deel van de kamers. Voor elke minigame worden dus de volgende eigenschappen onthouden: naam, weging, kamerKans1, kamerKans2, kamerKans3, kamerKans4, opbouw, een

---

<sup>4</sup>[https://en.wikipedia.org/wiki/Fisher\OT1\textendashYates\\_shuffle](https://en.wikipedia.org/wiki/Fisher%20Yates_shuffle)

maximum en of de minigame toegewezen is. Hoe het algoritme in detail in zijn werk gaat is te zien in de pseudocode te zien bij Algoritmen 2.

---

**Algorithm 2** Random toewijzen van minigames aan de kamers

---

```
1: Lijst minigames  $\leftarrow$  ingelezen configuratie file
2: string[] kamers  $\leftarrow$  Array ter grootte van het aantal kamers
3: int begin  $\leftarrow$  random getal tussen 0 en het aantal kamers
4: Lijst verplichteGames  $\leftarrow$  getVerplichteGames(minigames)
5: Lijst toeTeWijzenGames  $\leftarrow$  getToeTeWijzenGames(minigames)
6:
7: for int  $i = 0 \rightarrow$  aantal kamers do
8:     kamerNmr  $\leftarrow$  begin +  $i \%$  aantalKamers
9:     if !empty(verplichteGames) then
10:         kamer[kamerNmr] = verplichteGames[0]
11:         markeerGameAlsGekozen(verplichteGames[0])
12:         verplichteGames.remove(0)
13:     else
14:         kiesRandomGame(kamerNmr) ▷ Zie Algoritme 3
15:     end if
16:
17:     updateWegingen(minigames) ▷ Zie Algoritme 4
18: end for
19:
20: Schrijf bijgewerkte minigames naar schijf
```

---

---

**Algorithm 3** Random kiezen van minigames

---

```
1: alle minigames zijn beschikbaar via alleGames
2: kamerNummer  $\leftarrow$  waarde via functie-aanroep
3: somKans  $\leftarrow$  0
4: game  $\leftarrow$  0
5:
6: for game  $\in$  alleGames do
7:   somKans  $\leftarrow$   $\sum$  (weging(game)  $\times$  kans(game))
8:   random  $\leftarrow$  random getal tussen 0 en somKans
9:   x  $\leftarrow$  0
10:  while  $\sum$  (weging(game)  $\times$  kans(game))  $<$  x do
11:    x  $\leftarrow$  x +  $\sum$  (weging(game)  $\times$  kans(game))
12:  end while
13: end for
14:
15: game wordt gemarkeerd als gekozen
16: verwijder game uit alleGames
```

---

---

**Algorithm 4** Bijwerken van de wegenen der minigames

---

```
1: alle minigames beschikbaar via alleGames
2: for all minigame  $\in$  alleGames do
3:   nieuwe weging  $\leftarrow$  oude weging + opbouw
4:
5:   if nieuwe weging  $>$  maxWeging then
6:     nieuwe weging  $\leftarrow$  maxWeging
7:   else if minigame gekozen en opbouw  $<$  1 then
8:     nieuwe weging  $\leftarrow$  0
9:   end if
10: end for
```

---

## 6 Testen

In deze sectie wordt beschreven hoe er is getest. In subsectie 6.1 is beschreven hoe de scripts zijn getest. Vervolgens, in subsectie 6.2, is beschreven hoe gebruikers het spel getest hebben en hoe zij dat ervaren hebben.

### 6.1 Test scripts

Door middel van de test tool UnTested zijn er tests geschreven voor een hoop van de gemaakte scripts in Unity. De Unity Asset Store zegt het volgende over deze test tool: “UnTested makes Asynchronous Testing dead simple! Now you can easily write asynchronous tests and run them from an editor window” [Scopely, 2015]. Er is van alles getest. Bijvoorbeeld bij de minigames is getest of de juiste medaille wordt gegeven en getoond, bepaalde gameobjecten worden aangemaakt, gameobjecten worden verwijderd, dat scores juist worden berekend, dat animators aan gameobjecten worden gekoppeld, of teksten veranderd worden, etc. Een heel belangrijk script dat getest is, heeft te maken met het algoritme dat op basis van bepaalde kansen minigames toewijst aan verschillende kamers. Aangezien niet alles getest kon worden met de UnTested tool, zijn er use cases gemaakt en zijn deze handmatig doorlopen. Deze zijn te vinden in Appendix B.

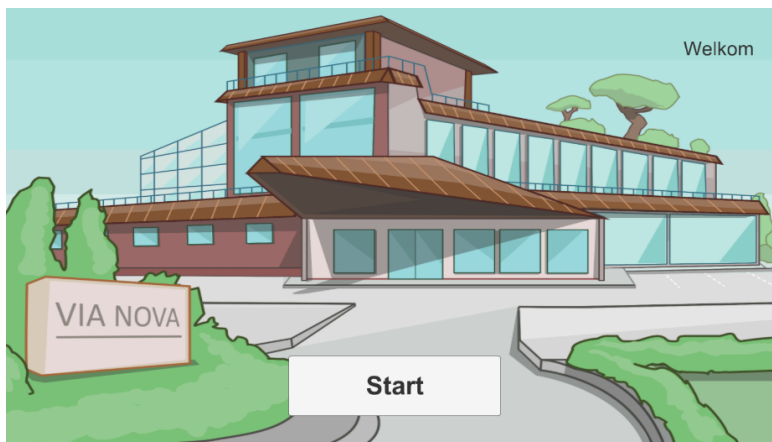
### 6.2 Gebruikerstest

Een gevarieerde groep met een leeftijd tussen 16 en 78 jaar, waarbij de gemiddelde leeftijd rond de 35 jaar lag heeft de gebruiksvriendelijkheid van het spel getest. De gebruiker werd door middel van een tutorial door het spel heen geleid. Hierbij ging de gebruiker in elke kamer een minigame spelen en werd ook de healthbar geïntroduceerd. Door middel van een enquête is de mening van de gebruiker gepeild. Voor de volledige vragen van de enquête zie de Appendix. Uit de resultaten van de enquête kunnen een hoop conclusies worden getrokken. Deze worden meegenomen bij het vervolg van project.

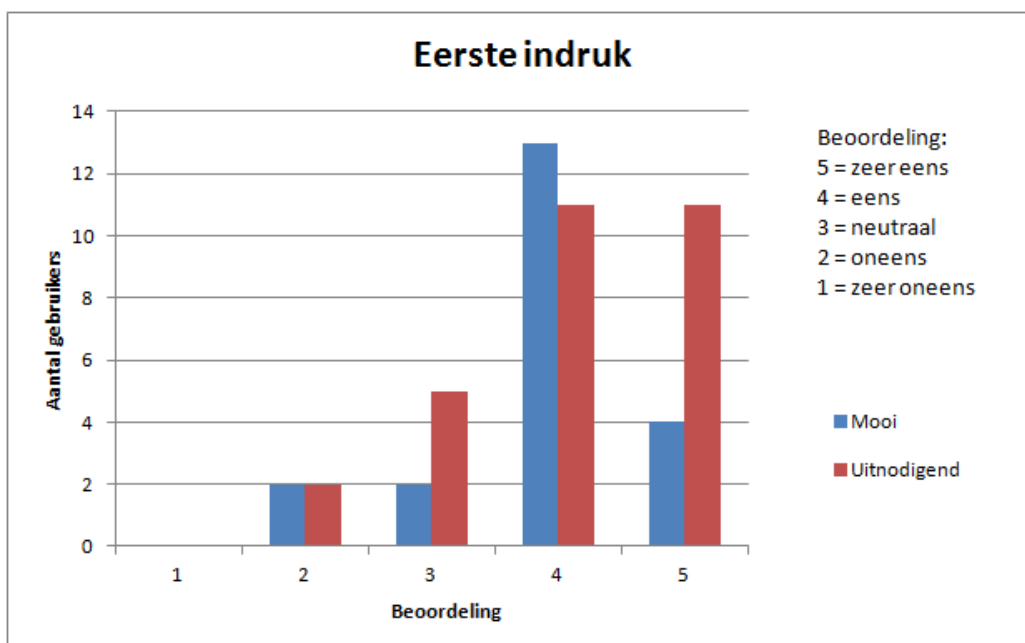
#### 6.2.1 Eerste indruk

Allereerst opende de gebruiker het spel en kreeg het scherm te zien dat zichtbaar is in figuur 14. De gebruiker is gevraagd naar zijn eerste indruk. Het

grootste gedeelte van de gebruikers was ermee eens dat het spel mooi en uitnodigend is, zie figuur 15.



Figuur 14: Startscherm



Figuur 15: Eerste indruk

### 6.2.2 Beoordeling kamers

Vervolgens is de gebruiker alle kamers af gegaan. Het uiterlijk van de kamers is te zien in figuur 16 tot en met figuur 18. De gebruikers waren het er redelijk mee eens dat de kamers mooi, open en vrolijk zijn, zie figuur 19.

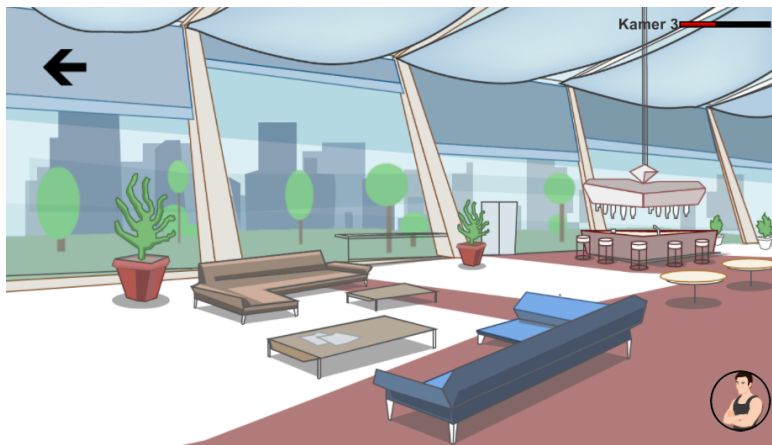


Figuur 16: Kamer 1, kamer gerelateerd aan de module 'slaap'

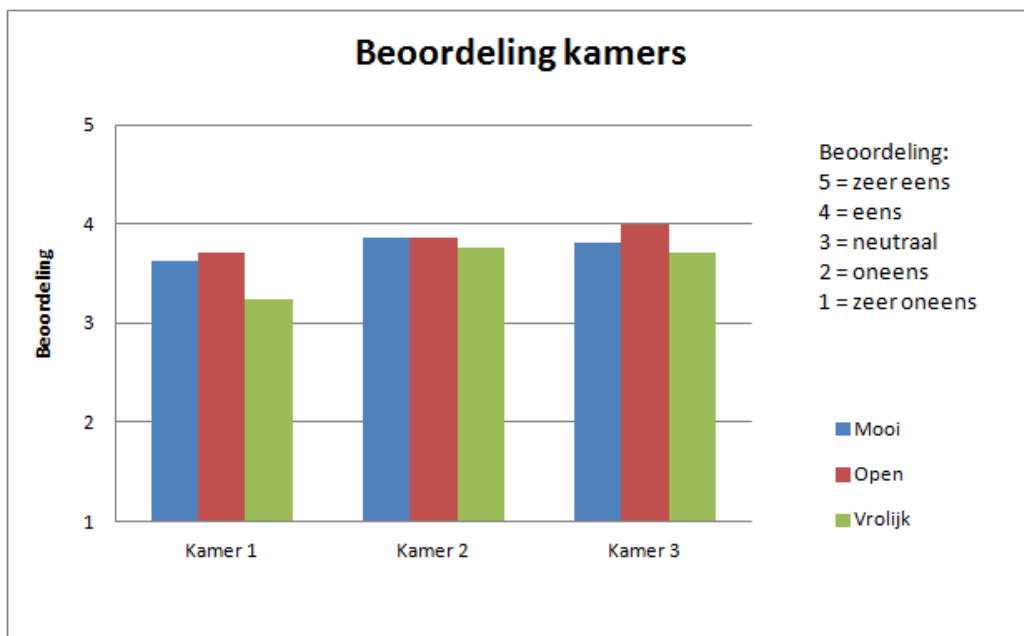


Figuur 17: Kamer 2, kamer gerelateerd aan de module 'leefstijl'





Figuur 18: Kamer 3, kamer gerelateerd aan de module ‘ontspanning’

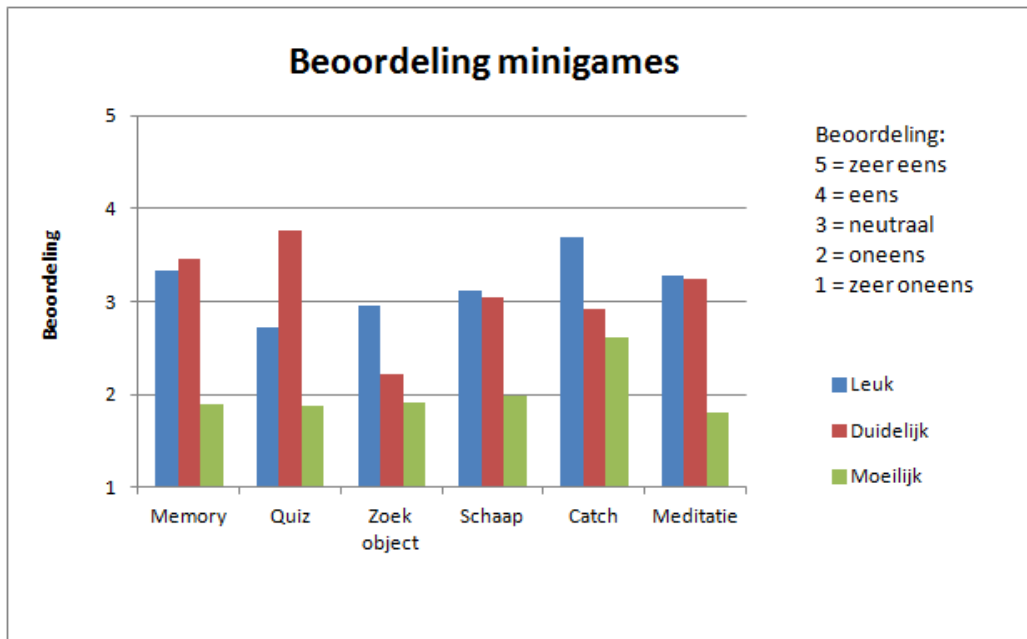


Figuur 19: Beoordeling kamers

### 6.2.3 Beoordeling minigames

Naast de kamers zijn de minigames ook erg belangrijk voor het spel. Uit figuur 20 blijken een aantal dingen. Ten eerste mag de moeilijkheidsgraad

van een hoop spellen omhoog, want de moeilijkheid heeft gemiddeld een score van 2 op een schaal van 5 gekregen. Ten tweede blijkt dat veel gebruikers de minigame Zoek object niet erg duidelijk vonden. Dit komt doordat veel gebruikers niet snapt hoe de besturing werkt van deze minigame. Dit zou eenvoudig opgelost kunnen worden door de besturing uit te leggen in het scherm dat getoond wordt voordat het spel begint. Ten tweede blijkt dat het spel Catch het goed doet bij de gebruikers. Dit komt doordat het spel een hoog entertainment gehalte heeft. Als laatste blijkt dat de Quiz het minder goed doet. Gemiddeld krijgt het zelfs een score van 2 op een schaal van 5. Verklaringen die gebruikers geven voor dit lage cijfer was dat het te simpel was en soms hadden gebruikers het gevoel dat ze niet serieus genomen werden.



Figuur 20: Beoordeling minigames

#### 6.2.4 Algemene ondervindingen

Er zijn een aantal dingen die de gebruiker niet duidelijk of enigszins lastig vond. De volgende opmerkingen zijn het meest belangrijk en deze zullen in het vervolg van het project meegenomen worden.

1. Eenmaal aangekomen in het avatarscherm had niet iedereen (vooral oudere gebruikers) door dat je een poppetje van jezelf moest maken. Ook was niet duidelijk dat je door op de pijltjes te klikken haar en kleding kon veranderen. De button om geslacht te veranderen was niet duidelijk zichtbaar.
2. In het algemeen hadden de wat oudere gebruikers moeite met Engelse termen zoals 'healthbar', 'minigame', 'avatar', 'go' en 'tutorial'. De teksten dienen in het vervolg volledig in het Nederlands te zijn, maar voor de wat jongere gebruikers zou dit niet nodig zijn.
3. Ook konden de gebruikers die wel wisten wat een healthbar was, deze niet altijd meteen vinden. Dit is eenvoudig op te lossen door de bar iets groter te maken en/of met een pijl naar de bar te wijzen.
4. De gebruiker wil graag meer feedback wanneer de muis op een object wordt gehouden dat naar een andere locatie leidt of dat een ander scherm opent. Een oplossing zou zijn om de desbetreffende objecten te laten oplichten zodra de muis erop wordt geplaatst.
5. Niet iedere gebruiker had meteen door hoe de muis gebruikt moest worden voor elk minigame. Dit zou voordat een minigame begint vertelt moeten worden. Daarnaast zou ook van te voren vertelt moeten worden hoe een spel wordt beoordeeld (bijvoorbeeld dat een spel op tijd gebaseerd is). Dan kan de gebruiker hier zich op voorbereiden.
6. Bij memory had gebruiker het idee dat er iets zou gebeuren als een set was gevonden. Nu lagen de gevonden sets gewoon open, maar ze zouden ook kunnen verdwijnen naar een stapel.
7. Sommige gebruikers gingen bij de minigame Schaaap de schapen tellen in plaats van deze aan te klikken.
8. Sommige gebruikers missen de toevoeging van geluid.

## 7 Proces

In deze sectie wordt het procesverloop beschreven. In subsectie 7.1 wordt beschreven hoe de communicatie verliep en welke afspraken hierover waren gemaakt, hierbij wordt onderscheid gemaakt tussen de interne en de externe communicatie. Subsectie 7.3 gaat in op de verdeling der taken en hoe dit werd bijgehouden. Als laatste gaan subsectie 7.4 over het gebruik van het versie controle systeem Git en subsectie 7.5 geeft een beeld over hoe het proces verlopen is.

### 7.1 Communicatie

De communicatie verliep primair via twee aangewezen PhD-studenten aan Leiden Universiteit, te weten drs. Meriem Manaï en drs. Roel Hogervorst. Zij zijn nauw betrokken bij het project en daarom ons eerste aanspreekpunt. Manaï en Hogervorst communiceren vervolgens met de daadwerkelijke cliënt, Prof. Dr. Andrea Evers. Manaï, Hogervorst en Evers komen wekelijks op donderdag bijeen (samen met nog enkele andere leden van de onderzoeksgroep) om het project te bespreken. Vervolgens koppelen Manaï en Hogervorst de feedback uit deze sessies naar ons terug. De communicatie verloopt normaliter per e-mail. Vanwege het gebruik van de SCRUM methodologie en om het communiceren te vergemakkelijken is besloten om elke maandag met Manaï en Hogervorst samen te komen (in beginsel in Delft). Zo kan de progressie van de week voorafgaand aan de meeting worden bespreken.

Binnen het projectteam is één iemand aangewezen als verantwoordelijke voor de communicatie; Sander. Hier is voor gekozen om het voor alle partijen duidelijk te maken met wie ze contact op moeten nemen. Sander is veel bereikbaar en kan hierdoor snel de verdere interne dan wel externe communicatie opzetten. Er is afgesproken dat er binnen een dag wordt gereageerd op email (waar mogelijk).

#### 7.1.1 Interne communicatie

Voor de interne communicatie wordt gebruik gemaakt van een WhatsApp<sup>5</sup>-groep, een Skype<sup>6</sup>-groep en de e-mail. Voor het bijhouden van de openstaande taken, en de verdeling van de taken, werd gebruik gemaakt van

---

<sup>5</sup>**WhatsApp** (mobiele messaging client) <http://www.whatsapp.com>

<sup>6</sup>**Skype** (audio/video conferenties en messaging client) <http://www.skype.com>

Trello<sup>7</sup> en later van GitHub issues. Naast deze tools is afgesproken om meerdere malen bijeen op een centrale locatie te werken. Dit is in principe de bibliotheek van de TU Delft. Meetings (met locatie) worden in de gedeelde Google-kalender<sup>8</sup> gezet.

### 7.1.2 Extern communicatie

De externe communicatie verloopt in principe allemaal per e-mail. Hierdoor staat altijd zwart op wit wat er is afgesproken / gezegd en kan er eventueel makkelijk iets worden teruggezocht. Om het proces te vergemakkelijken zijn er mailgroepen opgesteld waarin direct alle relevante mensen worden toegevoegd als ontvangers. Externe communicatie vindt onder andere plaats tussen naar Manai (et. al.), Bidarra, het design team, de project coördinatoren, enz.

## 7.2 SCRUM

Tijdens het project is gebruik gemaakt van de SCRUM methodologie<sup>9</sup>. De sprint retrospective, sprint planning en sprint review vinden allen plaats op maandag, tijdens en na de meeting met de cliënt (en Product Owner) uit Leiden. Dit omdat de cliënt op maandag in de gelegenheid was om naar Delft te komen. De Daily SCRUM wordt dagelijks uitgevoerd via de directe communicatie kanalen. De sprint review vindt aan het eind van de week plaats.

De SCRUM master zorgt voor het goede verloop van het SCRUM proces, deze houdt Trello (later Git) op orde en houdt toezicht op het algemene proces. Dion is in onze groep benoemd tot SCRUM master. Hij is ook benoemd tot Quality of Assurance en moet daardoor zorgen dat de gemaakte game het juiste doel bereikt. Alle communicatie verloopt via Sander en hij zal notuleren bij meetings. Daarnaast gaat Shirley over de algemene code-structuur van het spel en tenslotte is Chantal verantwoordelijk voor het testen.

De Definition of Done beschrijft wanneer een taak afgerond is. Een taak is afgerond, zodra de volledige functionaliteit is geïmplementeerd, getest en

---

<sup>7</sup>**Trello** (online organisatie tool) <http://www.trello.com>

<sup>8</sup>**Google Calendar** (online kalender tool) <http://calendar.google.com>

<sup>9</sup>**SCRUM** (Software ontwikkel methodologie) <http://www.scrum.nl/>

geïntegreerd. Na de integratie dient opnieuw getest te worden dat overige functionaliteiten nog steeds correct functioneren.

### 7.3 Taakbeheer

Voor het beheer van de (openstaande) taken wordt gebruik gemaakt van het eerder genoemde Trello. Hierin kunnen mensen aan taken (kaarten) worden gekoppeld, en de taken kunnen per SCRUM-sprint ingedeeld worden. Trello wordt bijgehouden door Dion, hij zorgt ervoor dat Trello up-to-date is en dat Trello overzichtelijk en bruikbaar blijft. Per sprint wordt bijgehouden wat er gedaan moet worden (de backlog, TO-DO), wat er op dat moment gedaan wordt (Doing) en wat er is afgerond (Done). Halverwege het project is besloten over te stappen op het issue-systeem van GitHub. Dit systeem zit direct geïntegreerd in het versiebeheersysteem (zie subsectie 7.4) en kon makkelijker links leggen tussen taken en de doorgevoerde veranderingen op het versiebeheersysteem.

### 7.4 Versiebeheersysteem

Voor het versiebeheer is een repository opgezet op GitHub<sup>10</sup>. De repository is te vinden via <https://github.com/sandervdo/BEP/>, deze is wel afgeschermd. Dit is bewust gedaan omdat dit project nog uitgevoerd moet gaan worden. Als tooling is gebruik gemaakt van SourceTree<sup>11</sup> met daarin geïntegreerd Git Flow<sup>12</sup>.

### 7.5 Verloop van het proces

In het begin werden problemen ondervonden met de communicatie. Het ontwikkelteam overlegde op maandag (het beginmoment van de SCRUM-sprint) met Manaï en Hogervorst. Zij overlegden vervolgens op donderdag met Evers (et. al.). Hierdoor kwam er een vertraging van enkele dagen in de feedback op de overlegde ideeën. Dit is opgelost door de meetings van Evers (et. al.) te verplaatsen naar de dinsdag, waardoor het ontwikkelteam na een dag in principe al feedback kon ontvangen.

---

<sup>10</sup>**GitHub** (online versiebeheer systeem) <http://www.github.com>

<sup>11</sup>**SourceTree** (Git client) <https://www.sourcetreeapp.com/>

<sup>12</sup>**Git Flow** (model) <http://nvie.com/posts/a-successful-git-branching-model/>

Na enkele weken werken heeft het ontwikkelteam besloten om Trello te vervangen door de geïntegreerde functionaliteit van Git. Het nadeel was dat we niet meer meerdere mensen aan één taak konden koppelen in dit systeem. Dit is opgelost door de overige mensen een notificatie te sturen over de taak. Het voordeel van de GitHub issues was de integratie in het versiebeheer systeem. We konden de taken (issues) koppelen aan specifieke commits in het versiebeheer systeem en zo dus eenvoudig terugzien welke taak met welke commit was afgerond. Ook vond het team het nuttig om ervaring op te doen met GitHub Issues.

Halverwege het project is ook besloten om het Git Flow model te gaan hanteren. Dit model schrijft een specifieke manier voor omtrent het gebruik van Git. Dit vereenvoudigt het proces / versiebeheer aanzienlijk. Elke nieuwe functionaliteit wordt ontwikkeld in een aparte aftakking. Deze branches zijn herkenbaar aan de naamgeving **feature/naam-van-de-feature**. Er wordt nooit direct een nieuwe feature toegevoegd aan de **master-branch** met de daadwerkelijke releases. Een feature komt altijd als eerste in de **development-branch**, waarin alle functionaliteiten worden getest. Hier wordt ook gecontroleerd dat er niet ergens anders in het systeem problemen zijn ontstaan door deze nieuwe feature. Aan het eind van de sprint wordt vanaf de **development-branch** een **release/naam-release-branch** gemaakt. Hierin wordt een werkende release klaargemaakt. Eventuele problemen worden nog verholpen en als er een werkend sprint-resultaat is dan wordt deze doorgevoerd op de **master-branch** (en de eventuele bug-fixes worden ook ‘development’ doorgevoerd).

## 8 Discussie en Aanbevelingen

Nadat het bachelor eindproject is afgerond is het product nog niet klaar om meteen gebruikt te worden voor onderzoek van de Universiteit Leiden. Allereerst moeten een aantal nieuwe functionaliteiten nog geïmplementeerd worden, dit wordt besproken in subsectie 8.1. De verdere ontwikkeling van de minigames die nog ontwikkeld moeten worden zijn genoemd in subsectie 8.2. Daarnaast is het design nog niet helemaal af en ook nieuwe functionaliteiten zullen designs nodig hebben, waarbij voornamelijk de minigames en de uitjes nog een hoop tijd gaan kosten. Dit wordt besproken in 8.3 Ten slotte zijn er nog een aantal context gerelateerd zaken die voor rekening van de klant zijn, deze zijn genoemd in sectie 8.4.

### 8.1 Functionaliteiten

De volgende functionaliteiten moeten nog worden geïmplementeerd:

1. Telkens als de algemene gezondheidsbalk gevuld is wordt het tijd voor een uitje. De avatar bevindt zich dan in een andere omgeving en speelt daar ook de minigames. Dit kan zijn in een dierentuin, bij een kermis, in een bioscoop of bij een feest.
2. De fontein in de tuin is pas na een aantal dagen beschikbaar. Als je op de fontein kan je alle minigames die je wel eens gedaan hebt opnieuw spelen om je highscores te verbeteren. De eerste paar dagen zal de fontein nog niet beschikbaar zijn omdat hij zogenaamd kapot is, maar daarna wordt hij geïntroduceerd door de coach.
3. Toevoegen van meer game elementen die ervoor zorgen dat de gebruiker vaker het spel gaat spelen. De volgende game-elementen doen dat al: de highscores, de medailles, de objecten voor de kamer en de uitjes.
4. Elke week moeten de minigames vanuit een andere ‘pool’ worden getrokken. Op deze manier zullen de willekeurig bepaalde minigames niet te vaak achter elkaar voorkomen.

### 8.2 Nieuwe minigames

Er zijn nu negen minigames geïmplementeerd, maar dit is te weinig om de gebruiker vier maanden ongeveer een kwartier per dag te vermaken. Het idee



is dat er minimaal 23 minigames bij komen, die elk uit drie levels bestaat (easy, medium, hard). Ook deze minigames moeten te maken hebben met de modules. Het gaat nog een hoop tijd kosten om deze minigames te maken. Er is wel al nagedacht over welke minigames mogelijk worden toegevoegd. Deze zijn te vinden in de Appendix C. Verder zullen er varianten op de minigames komen door bijvoorbeeld het design aan te passen.

### 8.3 Design

De huidige versie van het design is nog niet voldoende voor de eindversie van het spel die vier maanden duurt. De volgende elementen zijn nog nodig.

1. Meer outfits voor de avatar. Omdat de gebruiker wat ouder is moeten er nog meer outfits komen met lange mouwen en/of vesten. Ook moeten er meer haarkleuren voor de avatar komen zodat de gebruiker zich beter identificeert met de avatar.
2. Vier achtergronden voor elk van de uitjes. Telkens als de algemene gezondheidsbalk volledig gevuld is gaat de speler op een uitje. Dit is een nieuwe omgeving waar minigames gespeeld kunnen worden. Voorbeelden van nieuwe omgevingen zijn: dierentuin, kermis, bioscoop en een feest.
3. De vierde kamer wordt de immuuntraining kamer of toekomstkamer genoemd. De klant weet echter nog niet wat daar de bedoeling van is/wordt. Mogelijk wordt het een zwembad, sauna, een ruimte met jacuzzi of een futuristisch restaurant.
4. Van de andere drie kamers is het idee uitgewerkt, maar er moeten nog elementen toegevoegd worden die de kamers huiselijker en vrolijker maken.
5. Er staan al meerdere objecten in de kamers, maar uiteindelijk moeten er minstens 12 objecten aan en uit te zetten zijn in de kamer. De gebruiker kan objecten verdienen door minigames te spelen.

### 8.4 Context

Een aantal context gerelateerde zaken is voor rekening van de klant, aangezien zij verstand hebben van het psychologisch aspect van het spel. Er staan

nu zelfbedachte teksten in het spel, maar deze dienen vervangen te worden. Dit omvat de volgende teksten.

1. Introductieverhaal van het spel.
2. Aanmoedigende teksten na het spelen van het spel voor bronzen, gouden en zilveren medailles. Er zijn meerdere teksten per medaille nodig, zodat ze nog geshuffled kunnen worden.
3. Teksten van de tutorial. Bij het tutorial worden een tour gegeven naar de tuin, de kamers, naar de 4 minigames van de dag en worden verschillende functionaliteiten van het spel behandeld.

## 9 Conclusie

Het project is goed verlopen en er is aan bijna alle Must-haves en Should-Haves van het MoSCoW-model voldaan. Afgezien van de dilemma's, audio en speelbaarheid van twee maanden, maar dit is in overleg gegaan met de client. Deze zullen in het vervolgproject naast het maken van meer minigames aan de orde komen. Uit de gebruikerstest is gebleken dat de gebruiker het spel mooi en leuk vindt, maar dat de moeilijkheidsgraad van een hoop spellen nog te laag ligt. Gelukkig is de moeilijkheidsgraad eenvoudig aan te passen. Het gemaakte beloningssysteem zorgt ervoor dat de gebruiker minder snel verveelt zal raken en het spelen van het spel als leuk ervaart. De code van het spel is zo opgebouwd dat het goed onderhoudbaar is en dat dient ook zo te blijven bij het vervolgproject.

Het doel van het spel is om gedragsverandering te stimuleren en het motiveren tot het doen van oefeningen, waarbij het belangrijk is dat het spel als leuk wordt ervaren. Of dit doel behaald zal worden is pas over twee jaar bekend wanneer het psychologisch onderzoek van de Universiteit Leiden afgerond is.

## Referenties

- [Brown et al., 1997] Brown, S. J., Lieberman, D. A., Gemeny, B., Fan, Y. C., Wilson, D., and Pasta, D. (1997). Educational video game for juvenile diabetes: results of a controlled trial. *Informatics for Health and Social Care*, 22(1):77–89.
- [Bunchball, 2010] Bunchball (2010). Gamification 101: An introduction to the use of game dynamics to influence behavior.
- [Conrad et al., 2010] Conrad, M., Neale, J., and Charles, A. (2010). This is my body: the uses and effects of the avatar in the virtual world.
- [Fogg, 2009] Fogg, B. J. (2009). A behavior model for persuasive design. In *Proceedings of the 4th international Conference on Persuasive Technology*, page 40. ACM.
- [Griffiths et al., 2004] Griffiths, M. D., Davies, M. N., and Chappell, D. (2004). Online computer gaming: a comparison of adolescent and adult gamers. *Journal of adolescence*, 27(1):87–96.
- [Hofmann et al., 2010] Hofmann, W., De Houwer, J., Perugini, M., and Baeyens, F. (2010). Evaluative conditioning in humans: A meta-analysis.
- [HopeLab, 2015] HopeLab (2015). [http://www.hopelab.org/wp-content/uploads/2008/10/h1\\_rm\\_outcomesstudy\\_200808142.pdf](http://www.hopelab.org/wp-content/uploads/2008/10/h1_rm_outcomesstudy_200808142.pdf). Geraadpleegd 26 april.
- [iso.org, 2011] iso.org (2011). Iso 25010. [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=35733](http://www.iso.org/iso/catalogue_detail.htm?csnumber=35733). Geraadpleegd op 23 juni 2015.
- [Lieberman, 2010a] Lieberman, D. (2010a). Can playing digital games improve our health? <http://tedxtalks.ted.com/video/TEDxAmericanRiviera-Debra-Liebe;search%3Alieberman>. Geraadpleegd op 26 april.
- [Lieberman, 2010b] Lieberman, D. (2010b). Serious games: Mechanisms and effects. Hoofdstuk 8.
- [Lieberman, 2001] Lieberman, D. A. (2001). Management of chronic pediatric diseases with interactive health games: Theory and research findings. *The Journal of ambulatory care management*, 24(1):26–38.

- [Lim and Byron, 2009] Lim, S. and Byron, R. (2009). Being in the game: Effects of avatar choice and point of view on psychophysiological responses during play.
- [NVPI, 2008] NVPI (2008). Onderzoek maakt einde aan misvattingen over spelers van computergames. <http://www.nvpi.nl/nieuws/onderzoek-maakt-einde-aan-misvattingen-over-spelers-van-computergames>. Geraadpleegd op 21 april.
- [Reumafonds, 2015] Reumafonds (2015). Wat is reuma? <http://www.reumafonds.nl>. Geraadpleegd op 26 april.
- [Scopely, 2015] Scopely (2015). <https://www.assetstore.unity3d.com/en/#!/content/12325>. Geraadpleegd 2 juni.
- [Toril et al., 2014] Toril, P., Reales, J. M., and Ballesteros, S. (2014). Video game training enhances cognition of older adults: A meta-analytic study. *Psychology and aging*, 29(3):706.
- [Watson et al., 2012] Watson, P., de Wit, S., Cousijn, J., Hommel, B., and Wiers, R. (2012). Motivational mechanisms underlying the approach bias to cigarettes.
- [Watson et al., 2013] Watson, P., de Wit, S., Cousijn, J., and Hommel, B. W. (2013). Motivational mechanisms underlying the approach bias to cigarettes.
- [Wiers et al., 2009] Wiers, R., Rinck, M., Kordts, R., Houben, K., and Strack, F. (2009). Retraining automatic action-tendencies to approach alcohol in hazardous drinkers.
- [Yee, 2009] Yee, N. (2009). The proteus effect.
- [Yoon and Vargas, 2014] Yoon, G. and Vargas, P. (2014). Know thy avatar: The unintended effect of virtual-self representation on behavior.

# Appendices

## A SIG Feedback

Deze appendix bevat de evaluaties van de Software Improvement Group (SIG). Deze zijn meegenomen in de verdere ontwikkeling van het product.

### A.1 Eerste evaluatie

<b>Feedback door DENNIS BIJLSMA ontvangen op 29 MEI 2015</b>
--

De code van het systeem scoort net vijf sterren op ons onderhoudbaarheidsmodel, wat betekent dat de code bovengemiddeld onderhoudbaar is. Om deze score te behouden is het goed om te letten op de Unit Size en de Unit Complexity.

Voor Unit Size wordt er gekeken naar het percentage code dat bovengemiddeld lang is. Het opsplitsen van dit soort methodes in kleinere stukken zorgt ervoor dat elk onderdeel makkelijker te begrijpen, te testen en daardoor eenvoudiger te onderhouden wordt. Binnen de langere methodes in dit systeem, zoals bijvoorbeeld de ‘nieuwSchaap’-methode, zijn aparte stukken functionaliteit te vinden welke ge-refactored kunnen worden naar aparte methodes. Commentaarregels zoals bijvoorbeeld ‘//kies random welk dier te voorschijn komt’ of de verschillende if-blokken zijn een goede indicatie dat er een autonoom stuk functionaliteit te ontdekken is. Het is aan te raden om de lengte van methodes goed in de gaten te houden en deze waar mogelijk op te splitsen.

Voor Unit Complexity wordt er gekeken naar het percentage code dat bovengemiddeld complex is. Ook hier geldt dat het opsplitsen van dit soort methodes in kleinere stukken ervoor zorgt dat elk onderdeel makkelijker te begrijpen, makkelijker te testen en daardoor eenvoudiger te onderhouden wordt. In dit geval komen de meest complexe methoden ook naar voren als de langste methoden, waardoor het het in de gaten houden van de lengte ook de complexiteit zal reduceren.

Over het algemeen scoort de code bovengemiddeld, hopelijk lukt het om dit niveau te behouden tijdens de rest van de ontwikkelfase. De aanwezigheid van test-code is in ieder geval veelbelovend, hopelijk zal het volume van de test-code ook groeien op het moment dat er nieuwe functionaliteit toegevoegd wordt. Het valt trouwens wel op dat de test-code library ook aanwezig is in de code-base. Om verwarring te voorkomen zou het goed zijn om deze code duidelijk te markeren als externe code.

## A.2 Tweede evaluatie

**Feedback door DENNIS BIJLSMA ontvangen op 17 JUNI 2015**

In de tweede upload zien we dat het codevolume sterk is gegroeid, terwijl de score voor onderhoudbaarheid is gedaald van 5 naar 4 sterren. Dat is vrij normaal, op het moment dat een systeem gaat groeien is het moeilijk om zo'n hoge score vast te houden.

Op het niveau van de deelscores zien we een kleine verbetering voor Unit Size en Unit Complexity, die tijdens de analyse van de eerste upload als verbeterpunt werden aangemerkt. Een grote(re) stijging lag ook niet in de lijn der verwachting, aangezien jullie zoals gezegd al vrij hoog scoren.

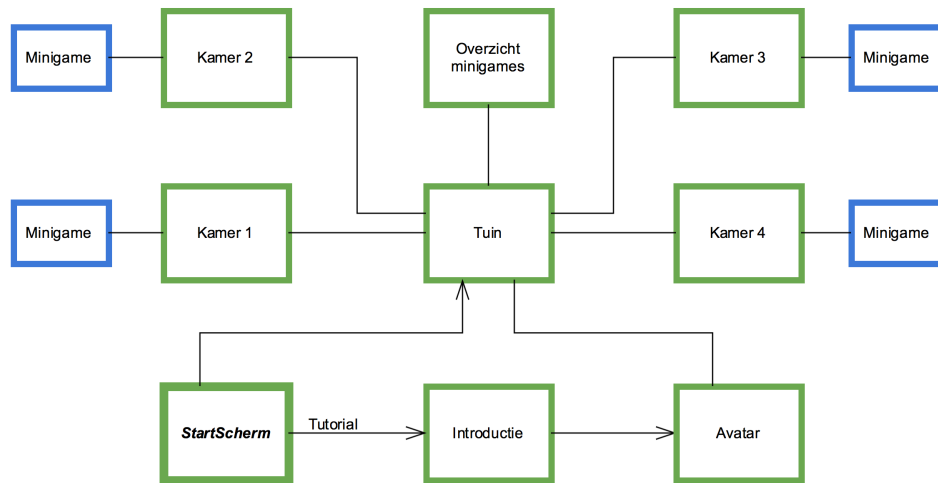
Het is goed om te zien dat jullie naast productiecode ook testcode hebben toegevoegd. Het valt wel op dat de verhouding productiecode/testcode sinds de eerste upload wat is afgezwakt, probeer hier voor de toekomst op te letten.

Uit deze observaties kunnen we concluderen dat de aanbevelingen van de vorige evaluatie zijn meegenomen in het ontwikkeltraject.



## B Testcases

In deze appendix zijn use cases uitgebreid besproken. Ze definiëren de beoogde werking van het product. In figuur 21 is te vinden hoe de routing verloopt.



Figuur 21: Routing door het spel

### B.1 Tutorial

In de tutorial wordt de gebruiker gedwongen een specifieke route, te nemen door het spel. Bepaalde knoppen horen dan uitgeschakeld te zijn en dit zal ook naar voren komen in de use cases. Aangezien het lastig is om click events te simuleren in een Unity test case (met behulp van de library UnTested) wordt elke van deze test cases met de hand getest.

Tabel 5: Volgende scene na START

Title	Volgende scene na START
Pre-Conditions	- Huidige scene is de START scene - Nog geen tutorial gedaan
Test Steps	Klik op <b>Start</b>
Expected Results	INTRO scene wordt geladen

Tabel 6: Volgende scene na INTRO scene

Title	Volgende scene na INTRO scene
Pre-Conditions	- De huidige scene is de INTRO scene - Nog geen tutorial gedaan
Test Steps	Klik op <b>Volgende</b>
Expected Results	AVATAR scene wordt geladen

Tabel 7: Volgende scene na AVATAR

Title	Volgende scene na AVATAR
Pre-Conditions	- De huidige scene is de AVATAR scene - Nog geen tutorial gedaan
Test Steps	Klik op <b>Verder</b>
Expected Results	TUIN scene wordt geladen.

Tabel 8: Tutorial TUIN 1A

Title	Tutorial TUIN 1a
Pre-Conditions	- Huidige scene is de TUIN scene - Tutorial is nog niet afgerond - Nog geen kamer gezien
Test Steps	- Klik op de linker draaideur - Klik op de rechter draaideur - Klik op het rechter bordje
Expected Results	Er gebeurt niks

Tabel 9: Tutorial TUIN 1B

Title	Tutorial TUIN 1b
Pre-Conditions	- Huidige scene is de TUIN scene - Tutorial is nog niet afgerond - Nog geen kamer gezien
Test Steps	Klik op het linker bordje
Expected Results	KAMER1 scene wordt geladen

Tabel 10: Tutorial KAMER1 A

Title	Tutorial KAMER1A
Pre-Conditions	<ul style="list-style-type: none"> <li>- Huidige scene is de KAMER1 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- Nog niet in andere kamers geweest</li> <li>- Nog geen minigame in KAMER1 gespeeld</li> </ul>
Test Steps	Klik op het TUIN-symbool
Expected Results	Er gebeurt niks

Tabel 11: Tutorial KAMER1 B

Title	Tutorial KAMER1B
Pre-Conditions	<ul style="list-style-type: none"> <li>- Huidige scene is de KAMER1 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- Nog niet in andere kamers geweest</li> <li>- Nog geen minigame in KAMER1 gespeeld</li> </ul>
Test Steps	<ul style="list-style-type: none"> <li>- Klik op het bed</li> <li>- Klik op het kruisje</li> </ul>
Expected Results	<ul style="list-style-type: none"> <li>- Panel van de minigame verschijnt.</li> <li>- Op het kruisje klikken doet niks</li> </ul>

Tabel 12: Tutorial KAMER1 C

Title	Tutorial KAMER1C
Pre-Conditions	<ul style="list-style-type: none"> <li>- Huidige scene is de KAMER1 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- Nog niet in andere kamers geweest</li> <li>- Net een minigame in KAMER1 gespeeld</li> </ul>
Test Steps	Klik op het TUIN symbool
Expected Results	TUIN scene wordt geladen

Tabel 13: Tutorial TUIN 2A

Title	Tutorial TUIN 2A
Pre-Conditions	<ul style="list-style-type: none"> <li>- Huidige scene is de TUIN scene</li> <li>- Tutorial is nog niet afgerond</li> <li>- Alleen in KAMER1 een minigame gespeeld</li> </ul>
Test Steps	<ul style="list-style-type: none"> <li>- Klik op het linker bordje</li> <li>- Klik op het rechter bordje</li> <li>- Klik op de rechter draaideur</li> </ul>
Expected Results	Er gebeurt niks

Tabel 14: Tutorial TUIN 2B

Title	Tutorial TUIN 2B
Pre-Conditions	- De huidige scene is de TUIN scene - Tutorial is nog niet afgerond - Alleen in KAMER1 een minigame gespeeld
Test Steps	- Klik op de linker draaideur
Expected Results	KAMER2 wordt geladen

Tabel 15: Tutorial KAMER2 A

Title	Tutorial KAMER2 A
Pre-Conditions	- De huidige scene is de KAMER2 scene - Tutorial nog niet afgerond - KAMER1, KAMER3 en KAMER4 nog niet - Nog geen minigame in KAMER2 gespeeld
Test Steps	- Klik op het TUIN symbool
Expected Results	Er gebeurt niks

Tabel 16: Tutorial KAMER2 B

Title	Tutorial KAMER2 B
Pre-Conditions	- De huidige scene is de KAMER2 scene - Tutorial nog niet afgerond - KAMER1 bezocht, KAMER3 en KAMER4 nog niet - Nog geen minigame in KAMER2 gespeeld
Test Steps	- Klik op het Ying-Yang logo - Klik op het kruisje
Expected Results	- Panel van de minigame verschijnt. - Op het kruisje klikken doet niks

Tabel 17: Tutorial KAMER2 C

Title	Tutorial KAMER2 C
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de KAMER2 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- KAMER1 bezocht, KAMER3 en KAMER4 nog niet</li> <li>- Net een minigame in KAMER2 gespeeld</li> </ul>
Test Steps	- Klik op het TUIN symbool
Expected Results	TUIN scene wordt geladen

Tabel 18: Tutorial TUIN 3A

Title	Tutorial TUIN 3A
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de TUIN scene</li> <li>- Tutorial is nog niet afgerond</li> <li>- In KAMER1 en KAMER2 is een minigame gespeeld</li> </ul>
Test Steps	<ul style="list-style-type: none"> <li>- Klik op het linker bordje</li> <li>- Klik op het rechter bordje</li> <li>- Klik op de linker draaideur</li> </ul>
Expected Results	Er gebeurt niks

Tabel 19: Tutorial TUIN 3B

Title	Tutorial TUIN 3B
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de TUIN scene</li> <li>- Tutorial is nog niet afgerond</li> <li>- In KAMER1 en KAMER2 is een minigame gespeeld</li> </ul>
Test Steps	- Klik op de rechter draaideur
Expected Results	KAMER3 wordt geladen

Tabel 20: Tutorial KAMER3 A

Title	Tutorial KAMER3 A
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de KAMER3 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- KAMER1 en KAMER2 bezocht, KAMER4 nog niet</li> <li>- Nog geen minigame in KAMER3 gespeeld</li> </ul>
Test Steps	Klik op het TUIN symbool
Expected Results	Er gebeurt niks

Tabel 21: Tutorial KAMER3 B

Title	Tutorial KAMER3 B
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de KAMER3 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- KAMER1 en KAMER2 bezocht, KAMER4 nog niet</li> <li>- Nog geen minigame in KAMER3 gespeeld</li> </ul>
Test Steps	<ul style="list-style-type: none"> <li>- Klik op de koelkast</li> <li>- Klik op het kruisje</li> </ul>
Expected Results	<ul style="list-style-type: none"> <li>- Panel van de minigame verschijnt.</li> <li>- Op het kruisje klikken doet niks</li> </ul>

Tabel 22: Tutorial KAMER3 C

Title	Tutorial KAMER3 c
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de KAMER3 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- KAMER1 en KAMER2 bezocht, KAMER4 nog niet</li> <li>- Nog geen minigame in KAMER3 gespeeld</li> </ul>
Test Steps	Klik op het TUIN symbool
Expected Results	TUIN scene wordt geladen

Tabel 23: Tutorial TUIN 4A

Title	Tutorial TUIN 4A
Pre-Conditions	- De huidige scene is de TUIN scene - Tutorial is nog niet afgerond - Minigame gespeeld in KAMER1, KAMER2, KAMER3
Test Steps	- Klik op het linker bordje - Klik op de linker draaideur - Klik op de rechter draaideur
Expected Results	Er gebeurt niks

Tabel 24: Tutorial TUIN 4B

Title	Tutorial TUIN 4B
Pre-Conditions	- De huidige scene is de TUIN scene - Tutorial is nog niet afgerond - Minigame gespeeld in KAMER1, KAMER2, KAMER3
Test Steps	Klik op het rechterbordje
Expected Results	KAMER4 wordt geladen

Tabel 25: Tutorial KAMER4 A

Title	Tutorial KAMER4 A
Pre-Conditions	- De huidige scene is de KAMER4 scene - Tutorial nog niet afgerond - KAMER1, KAMER2 en KAMER3 bezocht - Nog geen minigame in KAMER4 gespeeld
Test Steps	Klik op het TUIN symbool
Expected Results	Er gebeurt niks



Tabel 26: Tutorial KAMER4 B

Title	Tutorial KAMER4 B
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de KAMER4 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- KAMER1, KAMER2 en KAMER3 bezocht</li> <li>- Nog geen minigame in KAMER4 gespeeld</li> </ul>
Test Steps	<ul style="list-style-type: none"> <li>- Klik op de tv</li> <li>- Klik op het kruisje</li> </ul>
Expected Results	<ul style="list-style-type: none"> <li>- Panel van de minigame verschijnt.</li> <li>- Op het kruisje klikken doet niks</li> </ul>

Tabel 27: Tutorial KAMER4 C

Title	Tutorial KAMER4 C
Pre-Conditions	<ul style="list-style-type: none"> <li>- De huidige scene is de KAMER4 scene</li> <li>- Tutorial nog niet afgerond</li> <li>- KAMER1, KAMER2 en KAMER3 bezocht</li> <li>- Net een minigame in KAMER4 gespeeld.</li> </ul>
Test Steps	Klik op het TUIN symbool
Expected Results	TUIN scene wordt geladen

## B.2 Linked scenes

De volgende use cases gaan ervan uit dat de gebruiker de tutorial al heeft doorlopen. Bij deze use cases wordt gekeken of de juiste scene worden geladen na een bepaalde handeling.

Tabel 28: START naar TUIN

Title	START naar TUIN
Pre-Conditions	Huidige scene is de START scene
Test Steps	Klik op het TUIN symbool
Expected Results	De TUIN wordt geladen

Tabel 29: Tutorial naar KAMER1

Title	TUIN naar KAMER1
Pre-Conditions	Huidige scene is de TUIN scene
Test Steps	Klik op het linker bordje
Expected Results	KAMER1 wordt geladen

Tabel 30: Tutorial naar KAMER2

Title	Tuin naar KAMER2
Pre-Conditions	Huidige scene is de TUIN scene
Test Steps	Klik op het linker bordje
Expected Results	KAMER2 wordt geladen

Tabel 31: Tutorial naar KAMER3

Title	TUIN naar KAMER3
Pre-Conditions	Huidige scene is de TUIN scene
Test Steps	Klik op het linker bordje
Expected Results	KAMER3 wordt geladen

Tabel 32: Tutorial naar KAMER4

Title	TUIN naar KAMER4
Pre-Conditions	Huidige scene is de TUIN scene
Test Steps	Klik op het linker bordje
Expected Results	KAMER4 wordt geladen

Tabel 33: KAMER1 naar TUIN

Title	KAMER1 naar TUIN
Pre-Conditions	Huidige scene is scene KAMER1
Test Steps	Klik op het TUIN symbool
Expected Results	KAMER1 wordt geladen

Tabel 34: KAMER2 naar TUIN

Title	KAMER2 naar TUIN
Pre-Conditions	Huidige scene is scene KAMER2
Test Steps	Klik op het TUIN symbool
Expected Results	KAMER2 wordt geladen

Tabel 35: KAMER3 naar TUIN

Title	KAMER3 naar TUIN
Pre-Conditions	Huidige scene is scene KAMER3
Test Steps	Klik op het TUIN symbool
Expected Results	KAMER3 wordt geladen

Tabel 36: KAMER4 naar TUIN

Title	KAMER4 naar TUIN
Pre-Conditions	Huidige scene is scene KAMER4
Test Steps	Klik op het TUIN symbool
Expected Results	KAMER4 wordt geladen

### B.3 Minigames

De volgende use cases gaan over de minigame AAT. Omdat het lastig is om clicks event te simuleren is deze minigame alleen handmatig getest door de volgende use cases.

Tabel 37: AAT - score 0%

Title	AAT - score 0%
Pre-Conditions	- <code>maxAfb</code> staat op 3 ingesteld in het AAT script - De huidige scene is de AAT scene
Test Steps	1. Klik op het kruis. 2. Als een portret afbeelding verschijnt schuif deze afbeelding dan omhoog, schuif andere afbeeldingen omlaag 3. Voer stap 2 driemaal uit
Expected Results	Score panel wordt geladen en de score is 0%

Tabel 38: AAT - score 33%

Title	AAT - score 33%
Pre-Conditions	- <code>maxAfb</code> staat op 3 ingesteld in het AAT script - De huidige scene is de AAT scene
Test Steps	1. Klik op het kruis. 2. Als een portret afbeelding verschijnt schuif deze afbeelding dan omlaag en schuif andere afbeeldingen omhoog 3. Als een portret afbeelding verschijnt schuif deze afbeelding dan omhoog, schuif andere afbeeldingen omlaag 4. Voer stap 3 tweemaal uit
Expected Results	Score panel wordt geladen en de score is 33%

Tabel 39: AAT - score 66%

Title	AAT - score 66%
Pre-Conditions	- maxAfb staat op 3 ingesteld in het AAT script - De huidige scene is de AAT scene
Test Steps	1. Klik op het kruis. 2. Als een portret afbeelding verschijnt schuif deze afbeelding dan omlaag , schuif andere afbeeldingen omhoog 3. Voer stap 1 en 2 tweemaal uit 4. Als een portret afbeelding verschijnt schuif deze afbeelding dan omhoog, schuif andere afbeeldingen omlaag
Expected Results	Score panel wordt geladen en de score is 66%

Tabel 40: AAT - score 100%

Title	AAT - score 100%
Pre-Conditions	- maxAfb staat op 3 ingesteld in het AAT script - De huidige scene is de AAT scene
Test Steps	1. Klik op het kruis 2. Als een portret afbeelding verschijnt schuif deze afbeelding dan omlaag, Als een landscape afbeelding verschijnt schuif deze afbeelding dan omhoog 3. Voer stap 1 en 2 driemaal uit
Expected Results	Score panel wordt geladen en de score is 100%

## C Toekomstige minigames

Hieronder bevindt zich een lijst met ideeën voor toekomstige minigames. Dit zijn concepten die nog verder uitgedacht moeten worden.

**Kamer inrichten** Bij deze minigame is de gebruiker in staat om een lege / basiskamer in te richten. Dit kan door vooraf gedefinieerde objecten aan de kamer toe te voegen. Denk bijvoorbeeld aan het toevoegen van een specifieke bank / stoel, een tafeltje en meer decoratieve zaken zoals kleden, gordijnen, etc.

**Maaltijd samenstellen** Bij deze minigame kan de gebruiker een maaltijd samenstellen voor een fictief persoon. Hierbij is het doel om deze maaltijd zo gezond mogelijk te maken. Het idee hierachter is de gebruiker bewust te maken van gezonde eetpatronen.

**Visspel** Bij dit minigame moet de gebruiker voorwerpen uit een water vissen. Dit kunnen vissen of badeendjes zijn wat ontspannend werkt. Of juist slechte voorwerpen omdat deze niet in het water horen.

**Broodjes maken** Bij deze minigame moet de gebruiker gezonden broodjes maken. Hierbij kan de uitdaging zijn om veel broodjes te maken, bijvoorbeeld voor andere avatars, binnen een bepaalde tijd.

**Platformer** Binnen de platformer games (denk aan Mario) zijn er meerdere mogelijkheden. Zo kan er bijvoorbeeld een minigame zijn door een droomwereld.

**Koorddanser** In deze minigame is het de bedoeling dat de avatar balanceerd op een koord. De gebruiker moet aan de hand van de pijltjes toetsen zorgen dat de avatar netjes in balans blijft.

**Kudde schappen** Bij deze minigame is het de bedoeling dat de gebruiker een kudde schappen verplaatst van een ruimte naar de volgende.

**Wolf in schaapskleren** Hier is er binnen een kudde schapen een wolfs in schaapskleren. De gebruiker moet deze wolf zo snel mogelijk proberen te vinden.

**Voorkom stress** Deze minigame focust op het voorkomen van stress. Er moeten nog nader te bepalen handelingen worden uitgevoerd, waarbij bij onjuist uitvoeren de stress verhoogd.

**Afleidingen tijdens/voor slapen** Tijdens deze minigame zal de gebruiker worden geconfronteerd met slechte eigenschappen en afleidingen die een goede nachtrust versturen. Denk hierbij bijvoorbeeld aan vlak voor het slapen gaan eten en aan mobiele telefoons.

**Pong spel** Bij deze minigame moet er met een bal voorwerpen aangetikt worden. Deze bal wordt aan de hand van een platform bediend.

**Bal hooghouden** Het doel van deze minigame is een bal hooghouden, de bal mag hierbij niet de onderkant van het scherm raken.

**Tetris** Hierbij is de bedoeling een variant te maken op het al bestaande tetris. Bijvoorbeeld door het gebruik van fruit of yoga houdingen.

**Balans op een bal** Bij deze minigame is het de bedoeling dat de avatar in balans blijft. Onbalans wordt veroorzaakt door slechte voorwerpen, die de gebruiker mocht wegklikken of slepen.

**Dieren verzorgen** Bij deze minigame is het de bedoeling dat de avatar dieren moet verzorgen. Veel mensen hebben een positieve associatie met dieren. Verder is verzorging ook erg belangrijk. Aangezien het niet gepast is om de avatar bijvoorbeeld te laten douche, kan dit op deze manier opgelost worden.

**Mastermind met gezonde items** Deze minigame is gebaseerd op het al bestaande spel Mastermind, maar dan met gezonde items in plaats van kleurtjes. Hierbij wordt de gebruiker gestimuleerd na te denken en wordt er een associatie gemaakt met positieve items.

**Zoek spel in een krant** Deze minigame laat de gebruiker een item zoeken in een krant.

**Pooltafel** Een minigame met een pooltafel roept bij de gebruiker een associatie met ontspanning en gezelligheid.

**Solitaire** Bij deze minigame geldt, net als bij de pooltafel, dat solitaire vaak door mensen (vooral ouderen) wordt gespeeld als zijnde ontspanning.

**Doolhof** Bij deze minigame moet de avatar uit een doolhof zien te komen. Dit zou een standaard doolhof kunnen zijn, of bijvoorbeeld een winkel waar je items moet verzamelen.

**Gezonde koelkast** Bij deze minigame is het de bedoeling dat de gebruiker de binnenkant van een koelkast bekijkt en alle slechte voorwerpen uit de koelkast haalt.

**Voorwerpen scannen** De bedoeling bij deze minigame is om voorwerpen uit bijvoorbeeld een winkelmandje of winkelkarretje te halen en deze te scannen. Hierbij moeten de goede voorwerpen wel gescant worden en de slechte niet.

**Space invaders** Bij deze minigame is het de bedoeling dat de slechte voorwerpen worden weggeschoten.

**Fruit crush** Deze minigame is gebaseerd op Candy crush, alleen zijn de candy voorwerpen fruit voorwerpen.

**Moestuin** Bij deze minigame is het de bedoeling dat de gebruiker een moestuin bij houdt. Deze minigame kan, zodra deze ontgrendeld is, in de tuin gespeeld worden.



## D Infosheet

### DELFT UNIVERSITY OF TECHNOLOGY BACHELOR PROJECT INFOSHEET

---

## Via Nova

Game project voor het onderzoek Expect Health van de Universiteit Leiden

---

### *TU Coach*

Dr. ir. R. BIDARRA

### *Opdrachtgever*

Prof. dr. A. EVERS

### *Teamleden*

Chantal ECKHARDT

Dion DE HOOG

Sander VAN DEN OEVER

Shirley DE WIT

---

## Omschrijving

LEIDEN UNIVERSITEIT (LU) doet onderzoek naar de behandeling van chronisch zieke patiënten. Een van de onderzoeken kijkt naar de effecten van een game op de beleving van de behandeling voor reuma-patiënten. De game moet vier maanden lang, elke dag, gespeeld worden. De game mag dus niet snel vervelen en moet de gebruikers langere tijd bezig houden. De game moet een bepaalde leefstijl aanleren, de elementen hieruit moeten naar voren komen in de game. Tijdens de research fase kwam naar voren dat er veel aspecten zijn die afhankelijk zijn van de interpretatie van de gebruiker. Vormen, kleuren en andere elementen spelen een belangrijke rol bij deze interpretatie. Het proces verliep soepel, de taken waren goed verdeeld en alle voortgang werd secuur bijgehouden. Moeilijkheden werden ondervonden bij de communicatie naar de cliënt. Door een herorganisatie van deze communicatie is dit euvel verholpen. Het ontwikkelde product is een game, met daarin allerlei minigames waarin de gebruiker subtiel en op een leuke manier in aanraking komt met een 'ideale' leefstijl. Het product is een prototype voor het daadwerkelijke product. LU zal met een team van developers, waaronder het oorspronkelijke team, dit spel verder ontwikkelen en gebruiken bij een onderzoek op echt personen.

## Teamleden

Het team regelde als geheel de taakverdeling en planning. Afspraken werden gezamenlijk besproken en beslissingen werden ook met het hele team gemaakt. De verantwoordelijkheid van de overige taken wisselde en lag meestal ook niet bij één persoon. Een uitzondering hierbij is de communicatie. Sander was verantwoordelijk voor alle (externe) communicatie. De interesses van de teamleden varieerden flink. **Chantal** was bijvoorbeeld geïnteresseerd in de gebruiksvriendelijkheids-aspecten, **Dion** in de gaming aspecten en het coderen, **Sander** in het aansturen en coderen en **Shirley** in algoritmen en onderhoudbaarheid.

---

**Contactpersoon** Sander van den Oever <S.Y.vandenOever@student.tudelft.nl>

**Client** A. Evers (Leiden Universiteit, Faculteit der Sociale Wetenschappen)

**Coach** R. Bidarra (TU Delft, Faculteit Elektrotechniek, Wiskunde en Informatica, Intelligent Systems groep)

**Verslag** Het eindverslag van dit project kan gevonden worden via <http://repository.tudelft.nl>

**Presentatie** Het product wordt op 1 juli 2015 om 16:15 gepresenteerd te EWI, TU Delft.

## E Oorspronkelijke opdrachtomschrijving

Opdrachtomschrijving verkregen vanaf BEPSYS (<http://bepsys.herokuapp.com>) d.d. 26 mei 2015.

### OPDRACHTBESCHRIJVING

#### BACHELOREINDPROJECT TECHNISCHE INFORMATICA

Gaming voor een optimale behandeling bij chronische lichamelijke aandoeningen

### E.1 Opdrachtgever

Namens de sectie Gezondheids-, Medische-, en Neuropsychologie (GMN) aan de Universiteit Leiden wordt deze opdracht aangeboden. De contactpersoon is Prof. Dr. Andrea Evers (hoofd van de sectie GMN: [a.evers@fsw.leidenuniv.nl](mailto:a.evers@fsw.leidenuniv.nl)).

### E.2 Context

Wanneer iemand lijdt aan een chronische lichamelijke aandoening, zoals bijvoorbeeld reumatoïde artritis, met bijvoorbeeld klachten van pijn, bewegingsbeperkingen of vermoeidheid, brengt dit veel nadelen in het dagelijks leven met zich mee. Een chronische aandoening kan tevens veel invloed hebben op het psychologisch welbevinden en de sociale omgeving van een persoon. Zo is bijvoorbeeld het nemen van medicijnen, iedere dag weer, niet voor iedereen eenvoudig. Regelmatig krijgen patiënten met een chronische aandoening psychologische ondersteuning bij het omgaan met de ziekte. Onderdeel van deze behandeling is ook om mensen te leren een zo positieve mogelijke houding te ontwikkelen ten opzichte van hun behandeling en hun gezondheid. Met een positieve houding houden patiënten hun therapie makkelijker en langer vol. Dit heeft wederom een positieve invloed op de gezondheid wat weer kan leiden tot een betere levenskwaliteit of tot lagere ziektekosten. Ook is uit recent onderzoek gebleken dat een positieve houding ten opzichte van de behandeling en de daaraan gekoppelde gezondheid niet alleen een directe invloed kan hebben op de ervaren kwaliteit van leven van patiënten, maar ook op het beloop van een aandoening.

We willen mensen een training aanbieden waarin we hun ondersteunen een positieve houding ten opzichte van hun behandeling te ondersteunen en te versterken. Door middel van conditionering leert iemand door bepaald gedrag (bijvoorbeeld het nemen van medicatie) te koppelen aan een verbeterde ziekte-uitkomst: de behandeling wordt geassocieerd met gezondheid, welzijn, zelfstandigheid, mobiliteit, etc. Na voldoende koppelingen (bijvoorbeeld door een toepassing met gaming) kan zelfs de context van de behandeling een automatische lichamelijke reactie opwekken (bijvoorbeeld door het zien van de pil die dagelijks wordt ingenomen), die wederom de ziekte-uitkomst gunstig beïnvloedt.

### **E.3 Doel**

Ons einddoel is om een training te ontwikkelen waarbij mensen met een chronische lichamelijke aandoening leren om een zo positief mogelijke houding te ontwikkelen ten aanzien van hun behandeling en daarmee mogelijk het ziektebeloop gunstig te beïnvloeden. Hiervoor zijn wij op zoek naar een mobiele game (voor telefoon of tablet) waarin op een aansprekende, interactieve manier associaties worden gevormd omtrent de behandeling van de gebruiker, zoals b.v. het effect van de reguliere behandelingen op de gezondheid.

### **E.4 Doelgroep**

Gebruikers met chronische lichamelijke aandoeningen die via psychologische trainingen hun behandelresultaten willen optimaliseren en versterken. Aangezien de doelgroep waarschijnlijk weinig ervaring heeft met gaming, is het belangrijk om de game op een aansprekende manier vorm te geven, waarbij eventueel gebruik wordt gemaakt van oefeningen of minigames (bijvoorbeeld door middel van visualisaties).

### **E.5 Opdracht**

Ontwerp en creëer een game voor een mobiele telefoon of tablet, die gebruikers op een interactieve, aansprekende en eenvoudige manier informeert over de positieve eigenschappen van hun behandeling. Company description Gezondheids- en Medische psychologie richt zich op het bestuderen, voorkomen en beïnvloeden van psychologische aspecten van lichamelijke gezondheidsproblemen. Er wordt aandacht geschonken aan de relatie tussen

biopsychosociale processen (bijv. cognities, emoties en gedrag; psychofysiologische stress-systemen) en lichamelijke gezondheid. Hierbij gaat het zowel om gezondheidspreventie en -bevordering als om diagnostiek en behandeling bij mensen met (chronisch) somatische aandoeningen. Neuropsychologie is de (klinische) wetenschap die zich bezighoudt met de relaties tussen hersenen en gedrag en de stoornissen die zich hierin kunnen voordoen.

## **E.6 Extra informatie**

**Bedrijf** Universiteit Leiden, Sectie Gezondheids-, Medische en Neuropsychologie

**Client** Andrea Evers <A.Evers@fsw.leidenuniv.nl> (+31 (0)71 - 527 3739)

**TU Delft Coach** Rafael Bidarra <R.Bidarra@tudelft.nl>

**Presentatie** 1 juli 2015 om 16:15 te EWI, TU Delft.