Jani Saari

User Oriented UI Development in Games for Elderly People

Bachelor's thesis

Bachelor of Engineering

Information Technology

2020

South-Eastern Finland
University of Applied Sciences

Author (authors)	Degree title	Time
Jani Saari	Bachelor of Engineering	October 2020

Thesis title	
User Oriented UI Development in Health and Wellbeing Games	38 pages

Commissioned by

"ReissuEllu - wellbeing on wheels - a mobile learning environment"-project/South Savo Social and Healthcare Authority ESSOTE

Supervisor

Timo Hynninen

Abstract
The main goal of this thesis was to create a game for the elderly people that would provide the target audience with cognitive stimulus while remaining intuitive and easy to access. The game was a 2D puzzle game created using Unity engine, and it was tested with target audience to achieve iterative development cycle, where the testers were to tell the developers what they want to be added in the game.

The theory part discusses unique challenges in creating a game for this target audience. The thesis reflects the findings of the testing phase on the theoretical foundation.

The testing phase used talk aloud method, where the testers spoke out loud what they were experiencing when playing the game. This phase also included a short feedback form to gather the general opinion of the tester.

The final result of this project was a functioning game that was given to the commissioner of the project. The game combines random puzzle tile elements and a scoring mechanism that rewards the player for picking up additional objects on the road to the goal line, but also removes points for every tile used.

CONTENTS

1 INTRODUCTION

We live in a society where the ratio of elderly population is ever increasing. In the past few decades, it has become more obvious that there are challenges when it comes to the mental wellbeing of this elderly population – be it depression or decreasing cognitive capabilities. This is further accentuated by the limitations age brings along with it to one's mobility.

The bright side in this is that the number of elderly people that have access to smartphones, tablets or computers is also rising. These devices can be used to alleviate the aforementioned cognitive issues. There have been studies indicating that video games can be used to help people with depression (Russoniello et al. 2013). Furthermore, a Finnish study called FINGER demonstrated that cognitive training as part of a multidomain intervention could slow the decline of cognitive functioning in elderly people (Finnish Institute of Health and Welfare 2015).

This thesis explores the challenges of creating a wellbeing game for a specific target audience, the elderly people. The thesis was commissioned by the "ReissuEllu - wellbeing on wheels - a mobile learning environment"-project which seeks to provide the elderly with an access to different services that might be hard for them to reach otherwise – ranging from health professional's assistance to wellbeing games. The purpose is to deliver these services where the people live, especially in smaller villages in the South-Savo region.

The development of the game was a group effort with my fellow student, Justus Juutilainen, with some graphical assistance from Juuso Kolehmainen. This thesis has a focus on the UI development using iterative and agile development cycles based on the feedback from the target audience itself. This is an important aspect of game development because of the large age gap between the developers and the target audience – what feels like intuitive controls for 30-year-old computer-natives might feel utterly alien to a 80-year-old.

2   REISSUELLU GAME

The initial idea for the game came to us in Games by Saimaa Game Jam, organized by a gamer and game development community GameLab Mikkeli in December 2019. This was a three-day long game jam, an event where the participants formed groups and designed a game in the given time. The participants could choose to do whatever they wish or choose to pick the given theme for this event. The theme was the development of a game that would aid in maintaining physical or cognitive capabilities of elderly people, but it should also in some manner incorporate the ReissuEllu car which would be used to deliver both the game and other social or health services to sparsely populated areas of the South-Savo region.

Our team, consisting of myself, Justus Juutilainen and Juuso Kolehmainen opted to go for a game involving the theme, as it had the opportunity for future development for thesis. The first stage was swift brainstorming and planning. We took into account both our skillset, as well as the style of games our target audience might be interested in. This led us to go with a game which would somewhat resemble a popular boardgame, Labyrinth, in which pieces that turn into different directions would have to be placed on the board in order to advance to a desired location. To fit the theme of the car, these pieces would be a road, and the player would be moving a car on this road to a goal line.

The game engine we opted to use was Unity, as we were all familiar with using Unity through either hobbies or university courses. The tight schedule of the game jam style environment lead to a rather bugridden alpha version, but it was still in a presentable shape, as on the last day of the game jam the representatives of ReissuEllu came to evaluate the games. Our team finished first, and as such we agreed to further refine the product as our bachelor's theses. However, it should be noted that it became clear in the evaluation that we would be facing some difficulties in what is perceived intuitive in our own age category, and that of non-gamers in older age categories; our WASD-based controls seemed utterly confusing to the representatives of ReissuEllu.

This age gap is not restricted only to the students of the IT field: in the user survey of stackoverflow.com, a leading question-answer website for programmers, 69.2% of the professionals using Stack Overflow were 34 years old or younger (Stack Overflow Developer Survey 2020). Because of this age gap, the developers are unable to see the game as the target audience sees it. Therefore, we attempted to put heavy focus on tester feedback as driving force for the direction of the development.

2.1 Method of Implementation

After we had agreed to the initial deal to develop the game for ReissuEllu, we had a further meeting with their representatives. In this meeting, we agreed to the basic future features required for the game (randomization, scoring, collecting passengers), and the methods by which we would ensure that we shared the same vision. It also became apparent that the development would continue to use Unity, albeit with one major change – we would be working from different cities.

2.1.1 Using Unity

While the game could have been programmed with any engine, we opted to use Unity due to it being free to use, and our previous familiarity with the engine. The programming language used in Unity is mainly C#, which is the main language the developers are familiar with.

Unity is a game engine for game development and is the current market leader in game engine. Over half of the combined games on mobile, PC and console platforms were developed using Unity. (Unity 2019.) This makes using unity somewhat simple, as assistance to problems can be found both in Unity's own system reference document as well on question-answer sites such as stackoverflow.com.

In order to create a project like ours, one should start with the 2D-template of our project in Unity. Once the project has been created, one can begin to bring

assets into the subfolders of the game. Examples of assets to bring in are sprites, to create graphics for the game. These graphics can then be put on the game scene with by simply dragging and dropping.
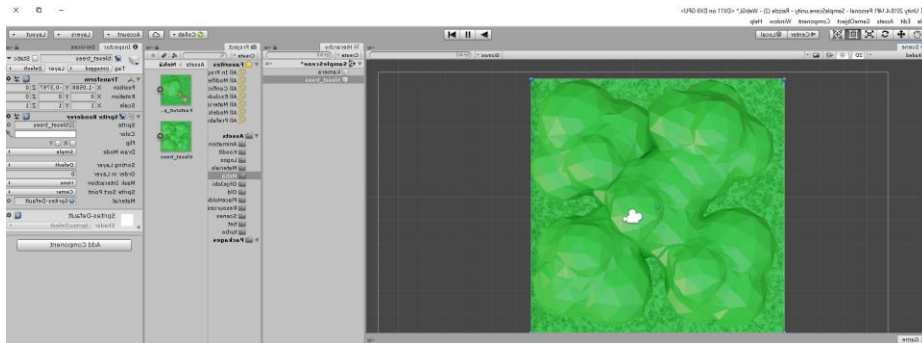
Figure 1. Dragging and dropping sprites

As the graphical components are dropped on the scene, they can be resized, copied and pasted. Eventually one can design the whole graphical appearance of the game in such a manner, although at this point it will still lack any functionality. The functionality for the road tile and the end tile seen in Figure 2 will be added with scripts later in the process.

Figure 2. The basic graphical outlook

Next, for this project, creating a car with which to move is essential. A car object should be brought forth to the scene like the other pictures. It then needs to be given rigidbody, collider and a script to move the car with. These can be seen in Figure 3.
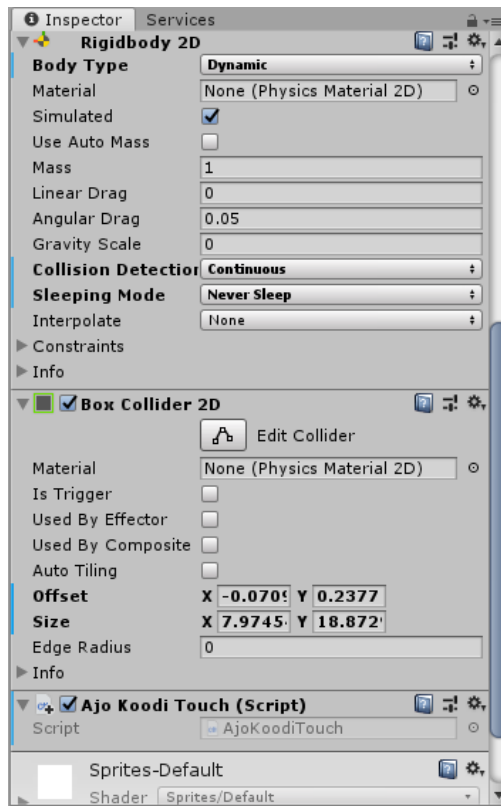
Figure 3. Rigidbody, collider and script

Rigidbody in unity serves the function of allowing the object that has a rigidbody attached to it to use Unity's built-in physics engine. The physics engi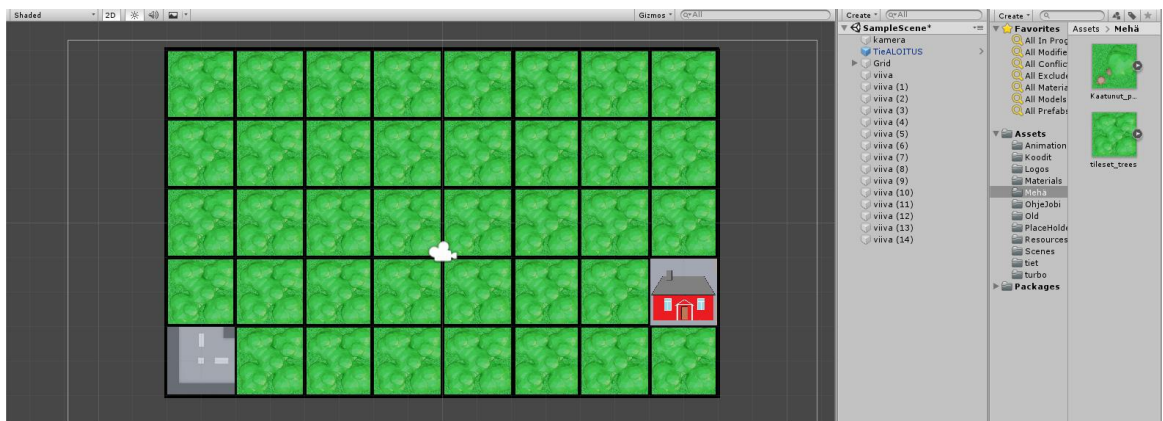ne deals with matters such as gravity, provided the object is given mass. This also allows one to have collision detection in the game provided that the sprites that are colliding use the collider component. (Unity Documentation 2020.)

The final part in making the car movable, functional element of the game is assigning a script to it. This is done is VisualStudio by using C# as the programming language. To determine when the car should move, the script compares the location of the yellow arrows to where the cursor is at the time of pressing left mouse button down. This button pressing automatically translates to touching the screen if the user is using an android device. The script then sets the rotation of the car in the correct direction by setting an angle value to the transform.rotation value of the car object, before using Transfrom.Translate function to move the car in that direction.

```
95   //cursor location
96   Vector3 hiiri = Camera.main.ScreenToWorldPoint(Input.mousePosition);
97   float hiirix = hiiri.x;
98   float hiiriy = hiiri.y;
99   float hiiriz = 0;
100
101   if (hiiriy < -5 && hiiriy > -7) {
102       //up
103       if (Input.GetKey(KeyCode.Mouse0) && (hiirix>-2) && (hiirix<0))
104       {
105           Vector3 eulerRotation = transform.rotation.eulerAngles;
106           transform.rotation = Quaternion.Euler(eulerRotation.x, eulerRotation.y, 0);
107           this.GetComponent<Transform>().Translate(0f, Time.deltaTime * 1.6f, 0f);
108
109       }//down
110
111       if (Input.GetKey(KeyCode.Mouse0) && (hiirix > 0) && (hiirix < 2))
112       {
113           Vector3 eulerRotation = transform.rotation.eulerAngles;
114           transform.rotation = Quaternion.Euler(eulerRotation.x, eulerRotation.y, 180);
115           this.GetComponent<Transform>().Translate(0f, Time.deltaTime * 1.6f, 0f);
116
117
118       }//left
119
120       if (Input.GetKey(KeyCode.Mouse0) && (hiirix > -4) && (hiirix < -2))
121       {
122           Vector3 eulerRotation = transform.rotation.eulerAngles;
123           transform.rotation = Quaternion.Euler(eulerRotation.x, eulerRotation.y, 90);
124           this.GetComponent<Transform>().Translate(0f, Time.deltaTime * 1.6f, 0f);
125
126       }//right
127
128       if (Input.GetKey(KeyCode.Mouse0) && (hiirix > 2) && (hiirix < 4))
129       {
130           Vector3 eulerRotation = transform.rotation.eulerAngles;
131           transform.rotation = Quaternion.Euler(eulerRotation.x, eulerRotation.y, 270);
132           this.GetComponent<Transform>().Translate(0f, Time.deltaTime * 1.6f, 0f);
133
134       }
135
```

Figure 4. Movement code

Once the colliders of the car and movement scripts are in place, colliders can be added to the edges of all possible road tiles as well. This will prevent the car from going through the edges of the road. Colliders are also to be added to the passenger objects to allow one to write a script that picks them up on collision. Colliders are also what detects when the car reaches the goal, which triggers loading the end scene.

The road tiles also have to be made into prefabs in order to make them function in the randomized tile selection on the left side of the screen. The script attached to the randomized tiles calls for a random selection of all available road tiles, then instantiates the said prefab in its allotted slot. This tile, if selected, is then placed on the main playing field should the user wish to do so.
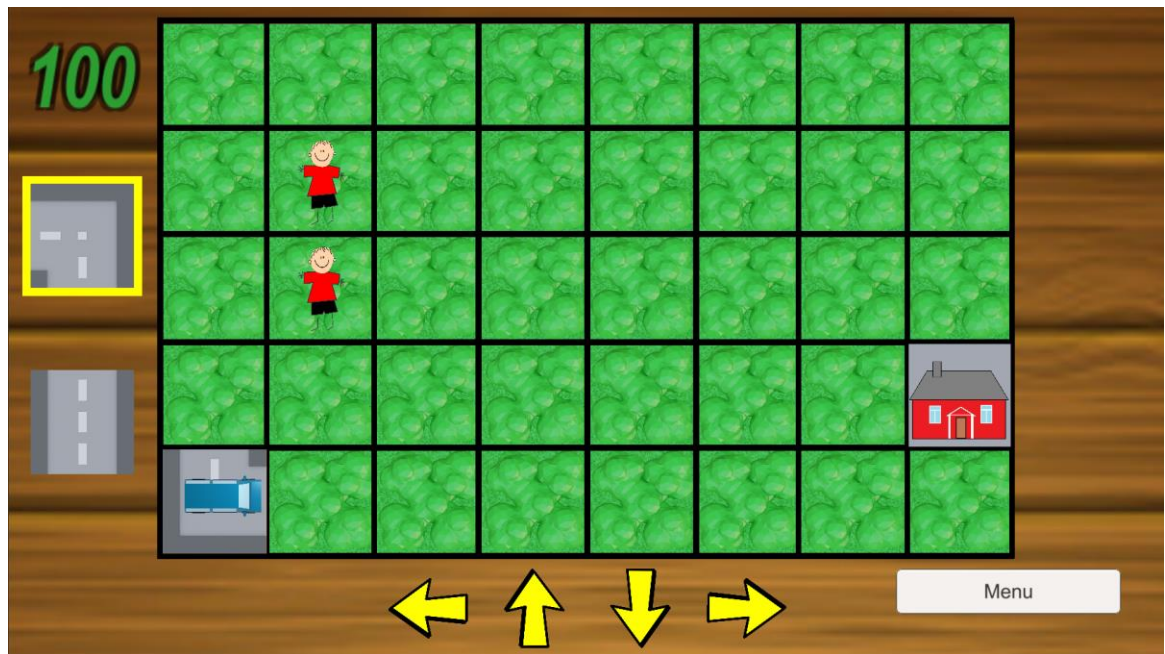
Figure 5. Finished graphical look

Figure 5 above also uses the canvas system of Unity to provide UI elements such as a menu and score. In the Unity editor, canvas and the main game elements are on vastly different scale, so that one does not accidentally confuse them for each other, as can be seen in Figure 6.
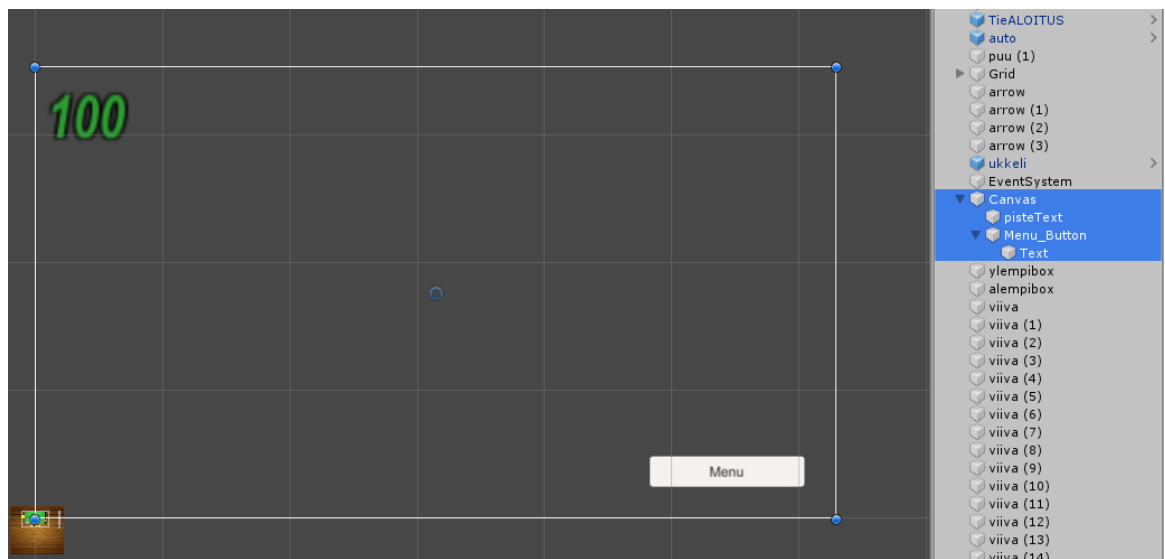
Figure 6. Standard game scale compared to canvas scale

Aside from buttons and text, canvas can also be used for graphical user elements that should always be on predetermined place on top of the gameobjects.

Once all of the scenes, objects and scripts required for the game are finished, the project needs to be built on the platform one desires. This can be found in the build settings section of Unity. Here the developer can also choose which scenes should be included in the game, as well as several other build related settings such as splash intro pictures and icons.
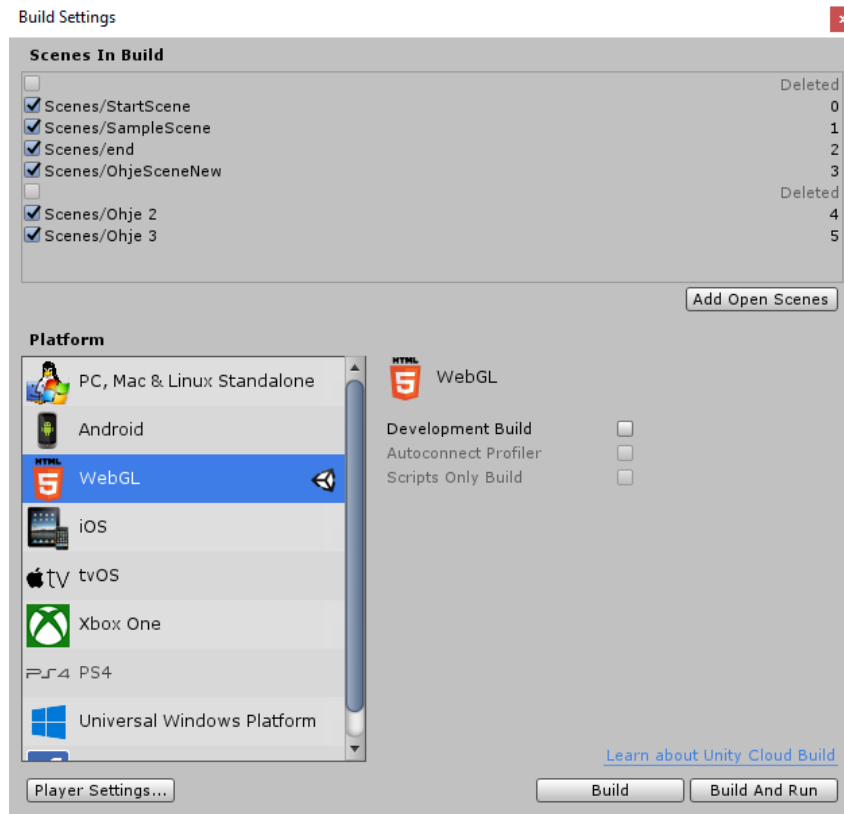
Figure 7. Build settings

2.1.2 Coordinating Team Development

One of the challenges that we faced in our development was that we as developers were living in different towns. Furthermore, the commissioner of the product was a person outside our development team. As such, we had to implement multiple means of keeping in touch with each other.

First, we had to keep the vision of the commissioner of the product in mind. Towards that end, we had roughly monthly meetings on how the product is progressing, what kind of new features should be implemented, et cetera. Since the commissioner lived in the same town as I do, it was easy to make these meetings function – the other developer would then attend the meetings via a

conference call. This proved to be an effective means of communication. Later, when the coronavirus caused a shutdown of all meetings, we opted to do the meetings entirely through Microsoft Teams. It is a business oriented program that allows video conferencing, file sharing and screen sharing, among other features that assist when working from home (Microsoft Teams: more ways to be a team 2020).

Secondary difficulty was the communication between the members of the development team. Since we were unable to see the content the other person was producing in real time, we had to figure out some type of way to organize the workflow and ensure that what we're implementing will not cause issues with the piece of code the other person is working on. We opted for a two-pronged approach in defeating these issues.

The first approach was to use a decent tool to organize the specific features, and assign them to one person, when it was being worked on by them. The service we used for this was Trello, a web based collaboration tool that organizes projects into boards and "tells you what's being worked on, who's working on what, and where something is in a process" (What is Trello? 2017).

The secondary way to ensure nonproblematic development was the usage of Unity's Collaborate tool. This tool, which is part of Unity Team, allows a small team to download and upload changes to and from the cloud, as well as synchronize the version of the project they are working on with the most recent version found in the Teams cloud service (Unity Manual: Unity Collaborate, 2020).

This two-pronged approach accompanied with a steady communication and picture sharing through an instant messaging service made the project relatively painless to manage. Further assistance to the innate difficulties of work-at-home development came from the monthly meetings which provided us with strict deadlines that we'd have to meet. At the end of the deadline we would push out a

new WebGL build of the program to demonstrate via a hosting service, so that everyone would be able to open the game, if they so wanted.

Without the use of these methods, a project can quickly devolve into a coding nightmare, especially if one foregoes good commenting practices in the code. Furthermore, as only one member of the development team was available in person for the testing sessions, it is vital that the findings of the testing sessions are shared with the rest of the team and the developmental tasks distributed based on discussions on how to react to these findings.

3   GAMES AND ELDERLY PEOPLE

This chapter will concentrate on the prevalence of gaming among elderly people, as well as the studies made on the preferences of features in games in this age category. It will further explore the challenges of this segment of population, as well as how to arrange user oriented development with them.

3.1   Prevalence of Gaming among Elderly People

While many in the younger segments of population might have the impression that next to no one in the 50+ age group plays video games, this is far from the truth. A survey that was performed in the United States in 2016 suggests that 38% percent play video games, with three quarters of the gamers playing weekly. The prevalence of more consistent gamers seems more common in the older age categories, with 37% of the gamers in the 50-59-year-old category reporting daily gaming, as opposed to 43% in the 60+ category. (Anderson 2016.)

3.2   Unique Challenges

An aging body and mind brings along conditions that present several challenges unique for that segment of players. According to WHO, "common conditions in older age include hearing loss, cataracts and refractive errors, back and neck pain and osteoarthritis, chronic obstructive pulmonary disease, diabetes, depression, and dementia" (World Health Organization 2018). These conditions

are rarely taken fully into account in game development (Gerling & Masuch 2011). When designing user interface of games targeted for older population, the following conditions should be taken into consideration.

3.2.1 Visual Impairments

A lot like many other other senses, visual senses deteriorate over time. By the age of 65 approximately one third of people have a condition that impairs their vision (Ganley & Roberts, 1983). Furthermore, other ailments can cause non-standard sight as well: for instance people with dementia slowly lose the ability perceive white color. The ability to see blue, turquoise and violet diminishes as people get older, while red, yellow and orange remains largely unaffected. (Sievänen et al. 2007, 22.)

These are important factors to take into consideration in the development of games, and UIs thereof. First thing to keep in mind should be that the details should not be too small: graphics should be clear and large. The same applies to any text given on the screen. There should be concise contrasts used with dark and bright to help the players separate objects within the game from one another.

Furthermore, the coloring schemes should be thought of from the angle of the user: While for the developer the differentiations between blues and violets might be easy to understand, the end user does not necessarily see a difference. This leads to frustration in gameplay, and lowers the motivation to continue playing. As such, important objects in the game (such as the end goal) should be using colors from the higher end of visible spectrum: yellow, orange and red.

3.2.2 Hearing loss

As with visual sense, the sense of hearing is also affected by the ravages of time. While this hearing loss can be a combination of age-related hearing loss and noise-induced hearing loss, it affects a large number of people. In the US, one in three people in the 65−74 age category have hearing loss, and this percentage increases to nearly half for the 75+ age category. (NIDCD 2018.) Furthermore,

the range of frequencies heard is lowered as the age of the person increases (Rodriguez Valiente et al. 2014).

While games are often audiovisual combinations, the prevalence of hearing loss among the elderly brings to question whether it should be used as a necessary part of the game. Rather, it should be used as a complimentary tool to assist the visual side: For instance, when a passenger is collected in a game by car, a 'hurrah' sound effect could play, reinforcing the visual clue of picking up a passenger with an audio clue. The sound effects should fall into the middle frequencies heard by a human to lower the chances that the player does not hear it due to age-related conditions.

3.2.3 Tremors

Another common condition among elderly population is shaking of hands, also known as tremors. This condition tends to correlate highly with age: over 20% of people in their seventies qualify for having tremors, and in people in their eighties the percentage climbs to 30%. In the 90+ category the percentage is over 50. (Deuschl et al, 2015.)

Tremors are likely to cause problems with games that require precise movements, especially with the mouse. Furthermore, they can lead to accidental double tapping of touchscreen. It stands to reason that the UI for the games should be developed so that the preciseness of a touch or a click is not a key factor. This means larger objects, which ties in well with the graphical side of the design. One should also develop a means to prevent accidental doubletapping, for instance, by introducing a timer that activates, when the user taps the screen and prevents tapping for the following 0.5 seconds.

The prevalence of tremors also plays a part in choosing the device for which the game is developed, when considering touch devices. With these users a tablet is most likely more useful a gaming device than a smartphone, and as such the UI optimization should be done with the common table dimensions in mind.

3.2.4 Reaction Speed

Reaction time and attention is yet another challenge in elderly gaming. It has been shown that there is a strong correlation between age and latency of response – depending on the difficulty of the task being performed, the latency increase between age groups of 32−44 years old and those of 65−74 year old can be 10 to 30 per cent (Tun & Lachman 2008). Its effect on gaming is reinforced by the perceived feeling of elderly players playing off-the-shelf games, where the players mention the game being too fast (Sopanen,2015, 17).

To avoid frustration caused by games that are too fast for their users, this knowledge should be brought forth on the design table of the game. Even if the game seems too slow for the developers, this is not necessarily the case for the target audience. One way to avert this problem entirely is to remove the time dimension from the game by choosing a turn-based approach. This approach foregoes both the difficulty of reacting in time as well as the pressure caused by timer mechanics sometimes used in games. This approach does require one to develop a different scoring mechanism though.

3.2.5 Digital Immigrants

The term Digital Immigrant was coined by Mark Prensky in 2001, and it refers to people who grew up before 1985 – which is considered to be the start of digital age (Hayes 2019). The terms "digital native" and "digital immigrant" are still in use, however the age itself does not seem to be the determining factor, rather than the previous usage of digital devices (Helsper & Eynon 2010). However, older adults tend to use digital devices much less per week than younger adults (Olson et al. 2011).

To reach as wide an audience in the age group as possible, one should consider the development of a multiplatform game. This is the approach our team opted to use with the use of WebGL. These kinds of games can be opened from the same link with android devices meeting the necessary android version requirements, as

well as with laptops and desktop computers: one only needs to program control mechanisms for both of these devices in that one version of the game.

Lesser familiarity with the devices can also lead to hesitation and doubt when trying something new. For these reasons the tutorial should be clear and show the player how to play the game, leaving little up for interpretation. This became obvious to us during our testing phase, described in chapter 4.

3.3   User-Oriented Development

To first get started with development for a specific audience, one needs to know what the audience wishes for the product. In regards to this thesis, this means knowing the general preferences of players in our age group. These can vary greatly between different age segments.

While with younger population violent games gain popularity, with 55+ year-old people it is the least wished for feature. Instead, the three most sought after features in a game are competition for high score, single-player only, and intellectual stimulation. (Blocker et al 2014.) These features will serve as the backbone in designing and developing the initial versions of the game under development.

To fully understand the wishes of the target audience, however, one has to use the iterative and incremental development cycle where the product and its features are tested on the users and feedback is collected. This feedback will then be used for the next iteration of the game. This development cycle should be continued until the final product is ready and can be given to the customer. (Farcic 2014.) This developmental cycle can be seen in process in the next chapter.

4   END-USER INTERFACE TESTING AND DEVELOPMENT

Once the initial game had been finished to the point that it was feasible to test it on our target audience, we entered the end-user testing phase. The testing was

slightly delayed due to the coronavirus epidemic, and for these reasons we had to opt for testing mainly in outdoor environment. The different weather conditions therefore added variation to the visibility, making the results not directly comparable with each other. It should be noted though that the testing sessions were intended to be an opportunity for the testers to see the game and provide opinions on what they would like to see changed in the game, and what features they would like added in it – in a sense, for them to become part of the game design team through their ideas and feedback.

Willing volunteers were searched to partake as testers by contacting local arrangers of recreational activities for elderly people. Fliers were also left in Omatori, a meeting place of citizens of more venerable age. Four individuals wished to take part in the development process: two women and two men. Both genders had one participant of age category 61−70, and one from the 71−80 category. One of the women was only able to partake in one of the testing sessions due to health reasons.

Test Session 1		Test Session 2		Test Session 3	
Gender	Age Group	Gender	Age Group	Gender	Age Group
Woman	61-70	Woman	71-80	Woman	61-70
		Man	71-80	Man	71-80
		Man	61-70	Man	61-70

Figure 8. Participants in each session

The testing was performed on a tablet using the "think aloud" method. In this method, the user is given a task, and then asked to think out loud their thoughts as they are performing the task. The testers are also notified that they are not the ones being tested, but the program itself. The observer of the test takes notes or video of the test and encourages the testers to keep talking. Help should be given only when the test cannot continue without it, either because of an inability to continue the test or the user quitting. (Lewis & Rieman 1993, 83−85.)

The test was performed three to four times in each session, starting with no explanation of the mechanics in the first round, and increasing the amount of

mechanics explained on each of the following rounds. This was done in order to test the entirety of the game in one test session, as well as to gather data for the creation of the tutorial section to assist first-time users understand the mechanics and the goal of the game.

After the testing was done, the testers were asked to give free feedback, followed by a questionnaire that asked the following questions: age, gender, devices they use and how much they use them. These were followed by game-specific questions: understanding the purpose of the game, the perceived fun factor of the game, ease of placing road tiles and driving the car, as well as whether they would play the game again in the future.

4.1   The First Test Round

Our first tester was a woman in the 61−70 years age category with no previous experience with mobile games. This test was performed on a tablet in a harbor cafe, in order to minimize the risks of the coronavirus. The testing equipment also included hand sanitizers.

The tester played the game four times. On the initial attempt with no instructions at all, there were difficulties in understanding the purpose of the game. Indeed, the tester thought the roadblocks were not roadblocks at all, but instead plus and minus signs. The purpose of the game was unclear.

On the second attempt the 'instructions' button in the main menu was pointed out to her. While she was able to read the text, it was difficult as the font size was so small. This highlights one of the challenges of creating games for elderly people, the deterioration of vision. Simple text instructions were not enough to alleviate the issues of confusion. While on the second test attempt the tester realized she would have to build a road, it was unclear how to select different pieces and how to put them on the grid.

On the third attempt, I gave the tester a brief verbal explanation on what should be done. This seemed to have a greater impact, and the tester was able to pick

pieces and place them in a pattern resembling a road. However, here we ran into another issue – the edges of the road pieces were not clear enough, thus causing the tester to consistently make errors that would trap her advancement. In addition, the tester did not initially realize the car was even a car – she thought it a fence.

Finally in the fourth test, I presented the tester with a pre-made road to test the driving system. This proved very intuitive system for the tester, and she was able to navigate the road and collect passengers with ease.

The tester felt strongly that the game was fun, but also felt strongly that the placement of the road was hard. Further, she felt that the purpose of the game wasn't easy to figure out. The tester agreed to test the next iteration of the game.

The biggest takeaway from this test round was that the car needs to stand out more, with clearer and more vibrant colors. The color of the car in this version of the game was pale blue, and as such it falls in the spectrum of colors that are harder for the elderly to see, as discussed earlier in this thesis. Moreover, the instruction screen should come before the main menu, so that users would know they should read it. The instructions themselves need to be clearer, perhaps through the means of animation and audio instructions. Finally, the edges of roadblocks must be more distinct, so that players get the feeling that they are impassable objects.

It should also be noted that the weather might have been a factor in these test results. The test happened at midday of a sunny summer day, which caused the visibility of the screen to be less than optimal. Regardless of all the feedback gained from this test round would be taken to development phase for the next iteration of the game.

4.2   Changes between the First and Second Test Rounds

Between the two first test rounds, we had but a small period for development. As such, our focus was on a) starting the development of tutorial-style help section for the game and b) graphical changes to improve the visibility of different components of the game.

While the development cycle of the tutorial scene would end up taking longer than the period between the two tests, some important minor changes were included between the tests one and two; namely contrast enhancing at the roadblock tiles to improve the visibility of road edges, and improved quality of the car graphics so that the testers would understand that it is indeed a car. The car was also changed to use darker graphics to stand out more from the roads. Finally, the buttons on the initial menu were greatly increased in sight for the sake of visibility, and the UI on that screen was made to scale based on the resolution of the target device.

4.3   The Second Test Round

All of the tests in the second round were performed in a lounge of an apartment building for senior citizens. Due to the coronavirus situation, the tablet used for testing was disinfected between each test session. Furthermore, each tester used hand disinfectant and only one test subject was allowed in the space at a time.

4.3.1   Tester 1

The first tester in our second-round tests was a woman in the 71−80 age category with little experience in mobile games. However, she was familiar with tablets and smartphones, using them 2−3 hours a day.

The initial attempt with the game was much like what was observed with the tester on the first test round – the instructions were deemed unclear. The first attempt at playing the game could be described as random placement of tiles and figuring out how it works. This could be interpreted as lack of familiarity with gaming in general, one of the challenges posed by being digital immigrant and not being exposed to gaming through the entirety of her life. The second and third attempts were showing increasing proficiency with the game, and the tester showed understanding that the placement of the tiles need not always be next to each other.

The fourth and final attempt concentrated on testing the driving aspect on a pre-made road. This proved to be the easiest part of the game for her, and driving was confident and flawless, which can be interpreted as the driving system being intuitive to use. While picking up a person from the road, however, she remarked that one "should not drive over people", though in the context of the game it was understood that it was simply to pick up people.

The post-test interview and feedback phase of the test session revealed that the instructions for the game were unclear. Further, the road roadblocks were initially hard to understand. However, the edges of the road once they were pointed out to her were easily visible. The tester found the purpose of the game clear, and strongly felt that the game was fun, that steering of the car was easy, and that she would try the game again. The placement of the road was considered of average difficulty. She agreed to partake in the test sessions of future iterations.

4.3.2 Tester 2

The second tester in our second-round tests was a man in the 71-80 age category. He did not own any device with a touch screen. However, he was familiar with tabletop computers.

The initial attempt, as seen in the previous tests, was difficult. It appeared unclear to him how the placement of the tiles on the map functioned. He often tried to use the arrow keys, intended to steer the car, to try to put the road tiles on the map.

During the second attempt, there was progress in understanding the placement mechanics. However, he found it was not clear that one needs to have a road under the car to advance on the playing field. He also often tried to place one tile on top of a tile that has already been placed. These difficulties, much like with the previous tester, could be attributed to lack of previous knowledge of games, or in other words, digital immigration.

The third attempt showed further improvement in making the road, however he encountered a bug that caused a tile different from that which was visually shown to the player to be put on the playing field. This caused further confusion.

For the fourth test attempt, I placed a road for him to drive on. As I did so, the tester mentioned that simply seeing how the roadblocks ought to be put on the field really helped him understand how the game should be played. He used the preplaced road for riding mechanics tests, and as before, this went flawlessly.

In the post-test feedback segment, he mentioned that the instructions in the game were unclear. He also said that it was not clearly stated that one cannot place a tile on top of another. For the tutorial section he suggested that something akin to a video of someone playing the game could be added. The tester thought the game was fun, and that driving the car was easy. The purpose of the game was unclear to him. Placing the roadblocks on the play-area was considered of average difficulty, and he was neutral regarding playing the game again. The tester did agree to further testing, however.

4.3.3 Tester 3

The third tester in our second-round tests was a man in the 61-70 age category. He had never used a touch-screen device, but did own a tabletop computer that he used 1−2 hours a day.

He started the initial test by reading the instruction screen. He found the instructions confusing, and way too small to read properly. The challenge of age-related decrease in eyesight affecting gaming was therefore observed. Upon first trying the game, he thought that the fallen trees in the background, intended to be simply deviation from the standard backgrounds of bushes, were obstacles. Further, the arrows were confusing, since they appeared to do nothing when pressed – this was due to the car not being on the road.

The second and third attempts with the game showed vast improvement in understanding the game, and from the logical pathbuilding that was done one could deduce that the tester had grasped the purpose of the game. Sadly, the bug of wrong piece being placed on the field struck again and foiled his attempts at successfully completing the game. In the fourth attempt of testing the riding mechanics on a pre-built road, the tester had no difficulty in steering the car.

In the post-test interview, the tester suggested that perhaps there should already be a crossroads piece under the car at the start of the game. Further, he said the same thing as the other testers: the instructions of the game were unclear, too small, and a video-style tutorial of the game would help him understand the core concept of the game easier. The tester said that the purpose of the game was very clear. The fun-factor, steering the car, and whether he'd like to play the game received a neutral answer. The placement of the road on the playing field was considered slightly difficult. This tester, like the others, said he'd be willing to try the game again in its next iteration.

4.3.4   Test Round Two Conclusions

The biggest takeaway from the test round two was that there needs to be a video/animation tutorial for playing the game. This will let the players know straightaway the purpose of the game as well as the game mechanics. The background should be simplified somewhat to avoid confusing the players with fallen trees that could be considered obstacles. Fixing the bug that causes wrong tiles to be placed on the field should also become a priority, for it tends to confuse the players as well as lower their motivation for playing the game.

The changes between test one and test two proved efficient in helping the people see the different objects in the game: What is a car, where are the edges of the road, et cetera? One must take into consideration, though, that this testing was done inside unlike the first one – as such, the differences in lighting might play a part in the better results.

4.4   Changes between the Second and Third Test Rounds

The changes between the second and third test rounds began with an improvement of the tutorial system. The changes were a shift away from small text-based approach into a system where the user sees an animation with a hand playing the game. In this first iteration of this system, the mechanic only showed the placement of the road tiles – and this was further explained with large font on the right side of the screen. While this approach left the rest of the game unexplained, it helped us get a clearer picture of whether this style of an approach is wanted.

Other changes that were made to the game were bugfixes to correct erroneous behavior that caused wrong tiles to be placed on the playing field, which had been plaguing the game since the initial build. We also introduced a popup screen that told a player that the tile is already taken, if the player tried to place a road tile on top of a pre-existing road.

4.5   The Third Test Round

The third test round followed the same pattern as before – the locations were chosen to minimize the risks involved with human contact in the times of a global pandemic, and disinfection was performed every time the person playing the game changed.

4.5.1   Tester 1

The first tester was a man in the 61−70 age category. He had already participated in one test before (see the second test round, Tester 3). It should be

noted that unlike the first time, on the second testing session the person did not use his glasses.

As the tester felt he knew how to play the game, he opted to skip the instructions. There were some immediate problems he faced when playing the game. First, he felt that he could place a tile on top of another, already placed tile. However, the new popup screen helped clarify the situation for him. Another difficulty was that it was not intuitive for him that unwanted pieces could be placed anywhere on the playing field – this caused him to run into deadends quite fast, if the tiles did not fit properly.

During the gameplay, the tester suggested on changing the background so that there are no fallen trees apparent – this is something that came up during the first test round, but we did not manage to implement for the new round of testing. Simplifying the game's graphical elements could help to counter the age-related visual impairments. The tester also made accidental doubletaps on the screen which he himself said were user mistakes, but ones he makes often. This lead to placing tiles in wrong spots. This was seen by the arranger of the test to be caused by tremors, another challenge of the target age group.

Further feedback was given on the size of the font in the ingame menu. The tester found it so small that it was hard to read. There was also a bug related to the ingame menu where one was able to click on the background instead of the menu button, as well as a bug after restarting the game from the ingame menu which caused the initial road tile placed not being the one that was shown. The latter bug caused quite a bit of confusion.

The tester also provided feedback on the tutorial screen, stating that it would be a good idea to notify the player that unwanted pieces can be put anywhere on the tilemap, instead of right next to the road as the tester tried to do initially.

The post-test questionnaire showed the following results on different areas of the game on the scale of 1 to 5, where 5 is the most positive result:

- Understanding the purpose of the game: 3 (down two points)
- How fun the game was: 2 (down one point)
- How easily understandable road placement was: 2 (up one point)
- How easy it was to move the car: 2 (down one point)
- Would you play the game again: 3 (no change)

4.5.2 Tester 2

The second tester was a man in the 71−80 age category. He had already participated in one test before (see the second test round, Tester 2).

This tester opted to look at the instructions first. This clarified the tile placement mechanics. At the beginning, he also tried to place a tile on top of another tile – and the popup-screen quickly helped him find out that it is not possible. The tester quickly grasped the basic mechanics of the game, but was unsure about whether to avoid the characters on the screen or to pick them up.

One of the difficulties for this tester was that there's no initial piece under the car. This often lead to him wondering why the car was not moving. The tester was able to successfully clear the stage after five tries.

The post-test questionnaire results were as follows:

- Understanding the purpose of the game: 4 (up two points)
- How fun the game was: 3 (down one point)
- How easily understandable road placement was: 2 (down one point)
- How easy it was to move the car: 4 (no change)
- Would you play the game again: 4 (up one point)

4.5.3 Tester 3

The third tester was a woman in the 61−70 age category. She had already participated in one test before (see the first test round).

Out of the people trying the game again in the third test round, she seemed to come to terms with how the game functioned the fastest. While she too had difficulties with the placement of the tiles, she quickly learned how to place them. There was only one attempt to place a tile on top of another tile, and after the popup appeared, it clarified the situation for her.

The difficulties for her were choosing the pieces on the left side of the screen and trying to see, if placing them would cause an obstacle. She further clarified that the tiles were very clear to understand, but somehow that didn't translate into how they were placed on the field.

Further, like some previous testers, she said the ingame menu buttons were too small. This again highlights the issues of deterioration of visual senses in older people. She also felt that the tutorial lacked instructions for driving the car, and suggested that one should likely place text ingame in front of the arrows to state what the arrows are for. She also suggested adding mention of placing unwanted tile pieces anywhere on the tile in the tutorial.

The post-test questionnaire results were as follows:

- Understanding the purpose of the game: 4 (up two points)
- How fun the game was: 5 (no change)
- How easily understandable road placement was: 4 (up three points)
- How easy it was to move the car: 2 (down three points)
- Would you play the game again: 5 (no change)

4.5.4 Test Round Three Conclusions

While most of the bugs that appeared in the second test version of the game were removed coming into the third test round, we did not see increase in how fun the testers found the game. Instead, we saw a decline in how fun the game was found. This could be due to the fact that there was little to no visual changes

since the original test. It could also be mere frustration at not being able to successfully finish the game, as the fun-factor seemed to heavily correlate with how well the person did in the game.

On average, understanding the purpose of the game and road placement improved. However, since explaining the steering mechanics was dropped from the tutorial, there was a decline in understanding how to drive the game.

Further, some of the complaints from the first test round still existed – fallen trees on the map, no tile under the car originally, et cetera. These ought to take priority on the development list, as the testers feel they're important. Further, the tutorial needs to have separate pages to explain that one can place tiles anywhere on the map, not only next to pre-existing road. The movement mechanic also needs be a part of the tutorial. While our original approach was to merely show the bare minimum needed to play to avoid flood of information, the better strategy based on this test round seems to be including the basics of everything.

To avoid doubletapping, due to incorrect technique of using the tablet or shaky hands, a cooldown timer of Boolean value should also be implemented to prevent two too quick taps.

5   FINAL PRODUCT AND FUTURE DEVELOPMENT PROSPECTS

5.1   User Feedback Results

Based on the different testing rounds we can draw some conclusions on the changes made into the game and the effects of the changes on the user experience. A good indicator for this would be the post-test feedback results. In all of the following diagrams we use a scale from one to five, in which one is a negative result and five is a positive one.

First, let's compare the results of the first time the testers played the game (the top diagram) to the second time they played (the bottom diagram) in the question "How clear was the purpose of the game".
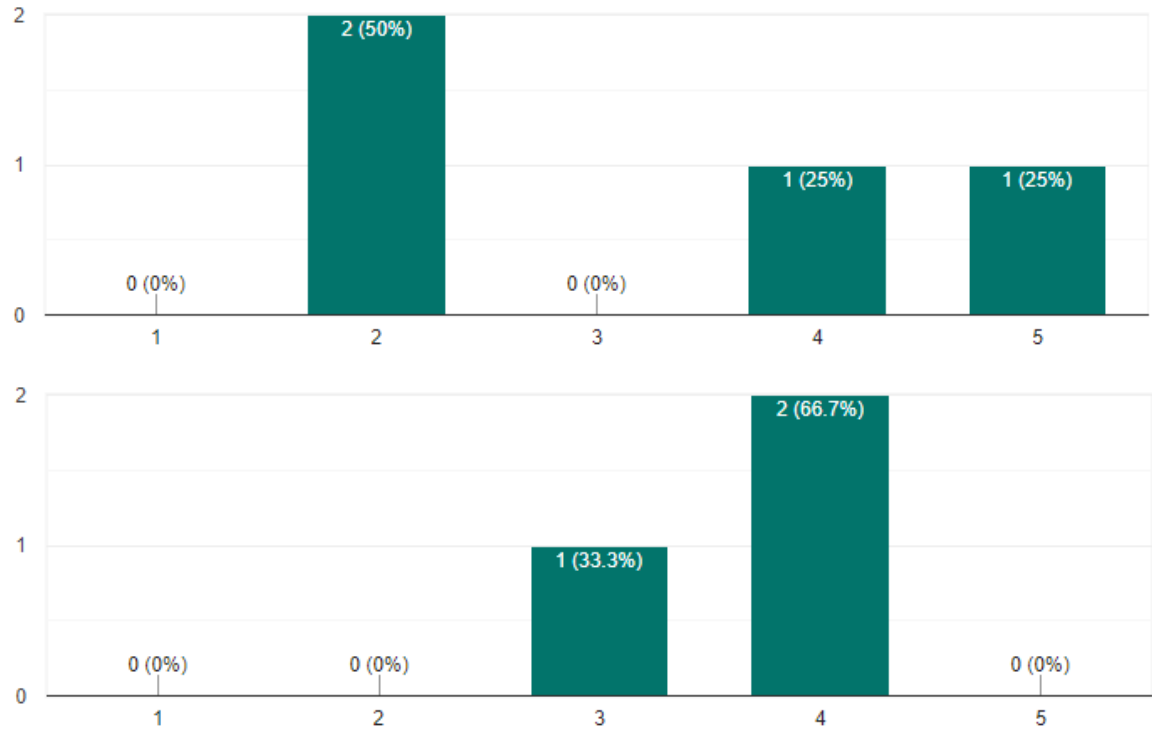

Figure 9. Understanding the purpose of the game. First time results (top) compared to second time (bottom)

The first test shows that for some of the players the purpose was clear, but at the same time others were struggling. Two of the testers reported that the purpose of the game was below average to understand. When we compare this to the results from the second time the testers played the game, we see the responses range from average to good. There has thus been some increase in understanding the purpose of the game.

While some of this increase could be attributed to playing the game for the second time, it should be taken into account that the tests were performed nearly a month apart, and the testers were unable to play the game in the meantime. It would stand to reason then, that the increased contrast based on user feedback and addition of a visual tutorial to the game would be contributing factors for the increase.

Next we move on to look at the results of the second question, "How fun do you find the game".
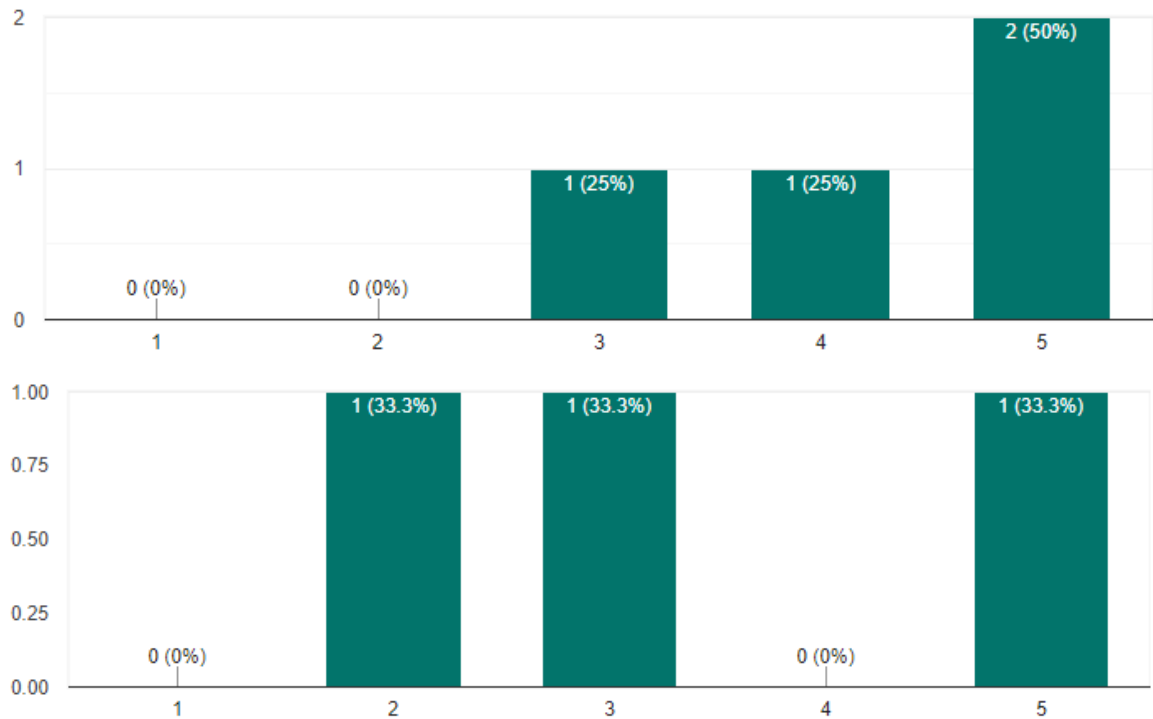

Figure 10. How fun the game was found. First time results (top) compared to second time (bottom)

In general it could be said that the fun factor of the game was trending downwards. While part of this can be explained by the fact that one of the testers that originally liked the game had to drop out of the testing, it does not explain everything. One possible reason was the usage of a negatively stylized road sign popup to let the user know when they are attempting to place a tile on top of another tile, as confusion as to why it can't be done is met with a frustrating popup.

Some of the testers also reported frustration on the difficulty level of the game. This type of loss of fun could possibly be alleviated by positive reinforcement, as well as a type of "hint" button which would suggest a tile placement for the user, if they are out of ideas.

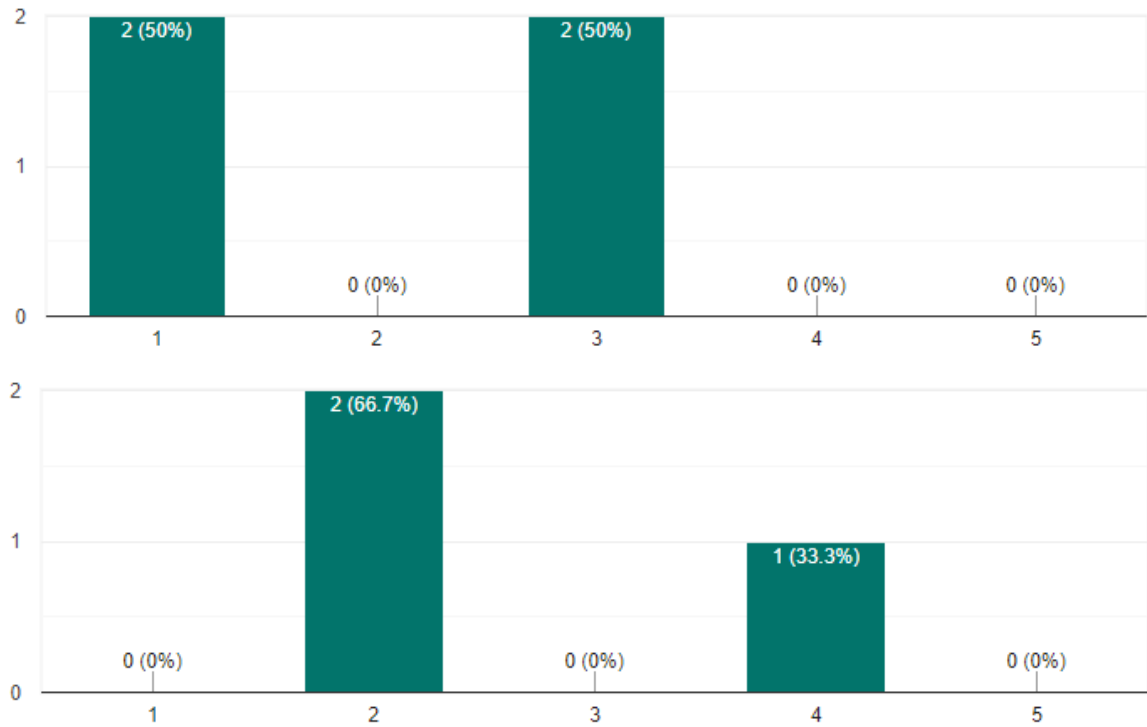The third question in our feedback form was "How easy was it to understand the roadplacing mechanics"?

Figure 11. The road placement mechanics. First time results (top) compared to second time (bottom)

On the first round half of the testers said the placement of road tiles was very hard to understand. That lead us to concentrate heavily on the development of a more tutorial-like structure with emphasis on the road placement. While the second-round test results showed that some of the players still considered them hard, we can see increase in this part of the game. The notable factor is that while the entire tutorial was not complete by the second test round, its tile placement portion was. In the tutorial, an animated finger showed the tap-to-select and tap-to-drop mechanics of the tile placement, and during observation in the testing the participants realized this very quickly. In this regard a visual animation tutorial seems an efficient tool for teaching how to play a game.

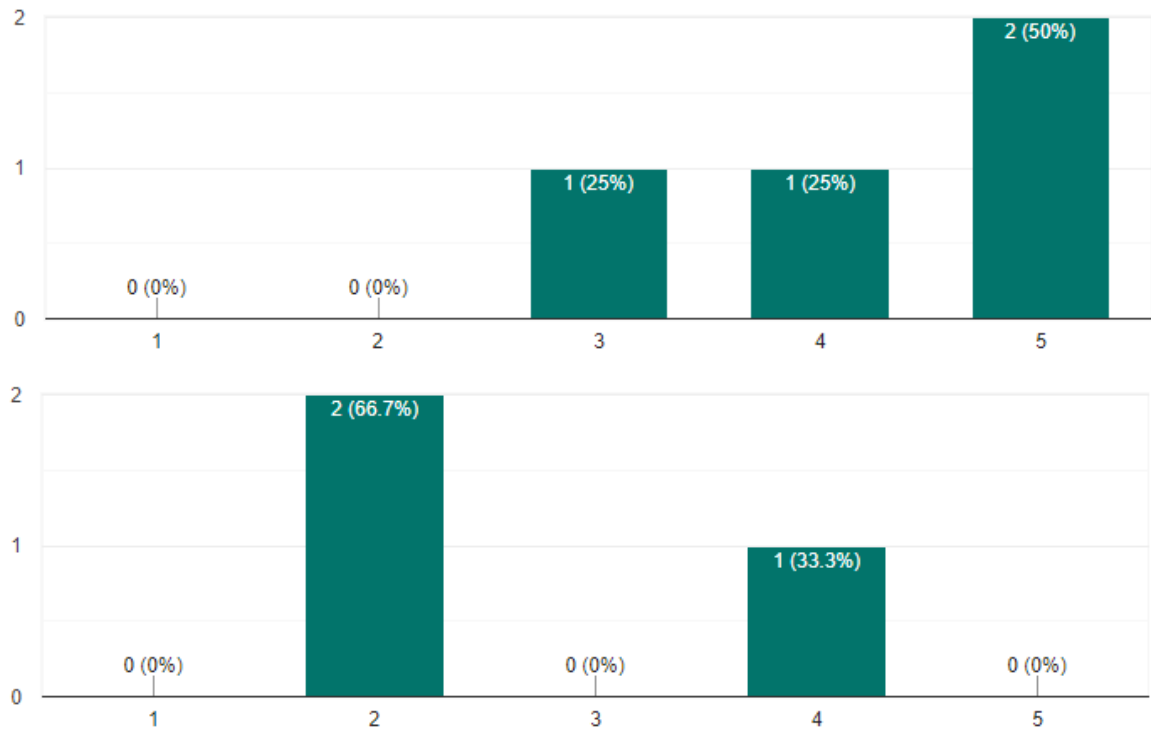The fourth question was titled "How easy it was to move the car"?

Figure 12. Car movement mechanics. First time results (top) compared to second time (bottom)

Here we see yet another significant drop in user experience. Whereas in the primary test round all users reported the car movement to be average, good or very good, on the second round the results ranged from bad to good. This is a significant drop in a core mechanic of the game. An explaining factor can be that the change of the tutorial from a text-based one to an animation-based one was not complete by the time the tests took place – whereas in the text-based tutorial it was explained that you steer the car by using the yellow arrows in the game, in the second round testing there was neither text nor animation based tutorial to explain these mechanics. As such the users were left to their own devices in trying to understand the mechanics. The take-in from this result is that one should never assume something is inherently intuitive when developing a game for an audience different from your own background.

The final question in the feedback form was "Would you play the game again".
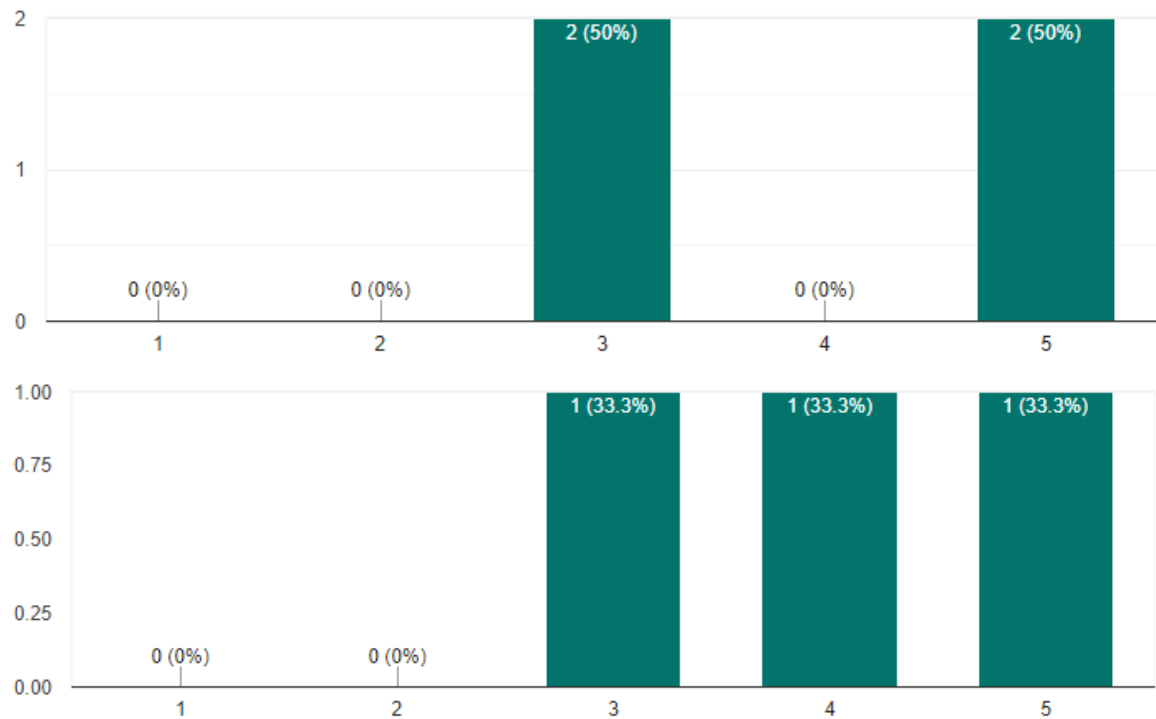

Figure 13. Would you play it again? First time results (top) compared to second time (bottom)

Here, the results stayed relatively the same, with all scores being maybe or above both at first time testing the game as well as the second time. It would appear that even though the fun factor was reported to be lower, the testers still felt the game was interesting enough to play again. This is an important value to take into consideration when thinking about further iterations of the game, and whether it shows promise in the minds of the testers. One could say that the premise of the game is good, but the execution in some of the test versions of the game was lacking.

5.2   User Accessibility for the Game

During development, the build that was used had been a WebGL one which functioned with both newer Android devices as well as computers. This build will be handed over to the commissioner of the product and will be put up on their servers to guarantee constant uptime for users as opposed to uncertain uptime using free hosting services during development. The end result of this is that the commissioner will then be able to provide a link to the game directly to the end users, who can use the product from their own home with their own devices

rather than being solely at the mercy of outside equipment to get access to the game.

In order to further increase the number of users that can access the game, one could make native builds for iOS and Android respectively, increasing the list of supported devices especially in the lower end devices oft found on older population. This can be easily done, as Unity as a game engine allows one to build the same project on several platforms. The app versions would also have the added benefit of being available even when there's no internet connection available.

Unity also supports building the project on consoles, such as Play Station 4. However adapting the project to be used by a console controller would require some work, and combining that with the low availability of consoles in the households of elderly people, this is not necessarily the way to go.

At the moment the game is largely to be used only in the South Savo region. There is however no reason why the game could not be used in the whole of Finland, or even globally. The primary concept of the game is likely to be rather universal, and cultural differences ought not be a large concern when making regional versions of the game.

5.3   Suggestions for Commercial Scale Development Goals

While the game developed by us is a functioning product, and can be adequately used by the commissioner for the task it was ordered for, it lacks the polish of a commercial scale product. Several key areas of the game could be improved with proper funding. These can be divided into visual outlook, audio, controls and accessibility.

On the visual front, one should take into consideration that the main development team did not have access to a fulltime graphical designer. We did get assistance in the form of assets from a graphical designer, however such expertise should be an essential part of the game development in other parts as well, such as

menus. Furthermore, the graphical look of the game could be further improved by changing some of the coloration themes to further support the altered vision of the elderly people.

The same limitations affected us on the audio side of things. Commercial scale development would allow funding to purchase sound effects and possibly music for the game. These sound effects would provide further feedback for the user in the game, when visual clues are not enough on their own.

On the control front most pieces are already in place. Should there be ports to consoles, these would require their own control systems. During our test phases we learned that touchscreen as a method to control a game seems very easy to learn even for users who had previously not used tablets. This should be the primary control type for commercial products.

Accessibility is covered for Android and PC products with the WebGL implementation. In Finland these devices are the major players on the market. However, if one were to think a commercial scale implementation of the product, there would be need for native support – especially for iOS systems. Furthermore, the current version only supports Finnish as a language. Unity offers great localization tools which could be used to translate the program to different languages, and as such allow wider reach for players.

6   CONCLUSIONS

The initial goal of this thesis was to create a functioning program for elderly people that would provide them with cognitive challenges and therefore help to alleviate the age-induced decline of mental capabilities. To achieve this we created a puzzle game with randomized roadblocks of different shapes that have to be connected together to form a path to the goal tile and drive the car there, while trying to collect passenger and also trying to keep the number of placed tiles to minimum. Specific focus was put to achieving this in a co-operative fashion with the target audience to help bridge the gap between digital native developers and the target audience, who are digital immigrants.

During the project it became clear that the considerations that should be taken into account when creating a game for the elderly population are numerous, ranging from what is considered intuitive for the target audience to general decline in sensory input due to age-related conditions. Much of this was learned from reading related literature, which could then be tied to the results we witnessed in the testing. It would have been preferable to have a few more testing rounds with the volunteers to take a deeper dive into the subject, but the coronavirus epidemic severely restricted the opportunities for this.

Overall, it could be said that the project was somewhat successful in achieving its original goal of being both easy to reach and play, while still providing mental stimulation for the player and giving them a reason to keep trying again. It should be noted however that if such a product were to be made for national or worldwide distribution, it would require a paid team of professionals to create a more polished product.

These types of technologies should be used more often in the future as the number of elderly people continues to increase – but so too increases the percentage of people in this group of people to whom technology is not an alien concept. The initial investment in the development of such an application is likely to pay itself back by offering a method to support the battle against regression of cognitive capabilities.

REFERENCES

Anderson, G. Oscar. Video Games: Attitudes and Habits of Adults Age 50-Plus. Washington, DC: AARP Research, June 2016. https://doi.org/10.26419/res.00125.001

Blocker, K., Wright, T. & Boot, W. 2014. Gaming preferences of aging generations. *Gerontechnology*, 12(3), pp. 174-184.

Deuschl, G., Petersen, I., Lorenz, D. & Christensen, K. 2015. Tremor in the elderly: Essential and aging-related tremor. *Movement Disorders*, 30(10), pp. 1327-1334. doi:10.1002/mds.26265

Farcic, V. 2014. Software Development Models: Iterative and Incremental Development WWW document. Available at: https://technologyconversations.com/2014/01/21/software-development-models-iterative-and-incremental-development/ [Accessed 14 September 2020].

Finnish Institute of Health and Welfare. 2015. *FINGER-tutkimuksen tuloksia*. WWW document. Available at: https://thl.fi/fi/tutkimus-ja-kehittaminen/tutkimukset-ja-hankkeet/finger-tutkimushanke/finger-tutkimuksen-tuloksia [Accessed 19 October 2020].

Ganley, J. P. & Roberts, J. 1983. Eye conditions and related need for medical care among persons 1-74 years of age: United States, 1971-72. *Vital Health and Statistics, series 11 no. 228.* Washington: U.S. Government Printing Office.

Gerling, K., & Masuch, M. 2011. When gaming is not suitable for everyone: Playtesting wii games with frail elderly. University of Duisburg-Essen.

Hayes, A. 2019. Digital Immigrant. WWW document. Available at: https://www.investopedia.com/terms/d/digital-immigrant.asp [Accessed 21 September 2020].

Helsper, E. J. & Eynon, R. 2010. Digital natives: Where is the evidence? *British Educational Research Journal*, 36(3), pp. 503-520. doi:10.1080/01411920902989227

Lewis, C. & Rieman, J. 1993.Task-Centered User Interface Design: A Practical Introduction. Shareware document. Available at: http://hcibib.org/tcuid/tcuid.pdf [Accessed 3 October 2020].

Microsoft Teams: More ways to be a team. 2020. WWW document. Available at: https://www.microsoft.com/en-us/microsoft-365/microsoft-teams/group-chat-software [Accessed 16 September 2020].

NIDCD (National Institute on Deafness and other Hearing Disorders). *Age-related hearing loss.* 2018. WWW document. Available at: https://www.nidcd.nih.gov/health/age-related-hearing-loss [Accessed 19 September 2020]

Rodríguez Valiente, A., Trinidad, A., García Berrocal, J. R., Górriz, C. & Ramírez Camacho, R. 2014. Extended high-frequency (9-20 kHz) audiometry reference thresholds in 645 healthy subjects. *International Journal of Audiology, 53*(8), pp. 531-545. doi:10.3109/14992027.2014.893375

Olson, K., O'Brien, M., Rogers, W. & Charness, N. 2011. Diffusion of Technology: Frequency of use for Younger and Older Adults. *Ageing International*, 36(1), pp. 123-145. doi:10.1007/s12126-010-9077-9

Russoniello, C. V., Fish, M. & O'Brien, K. 2013. The efficacy of casual videogame play in reducing clinical depression: a randomized controlled study. *Games for Health Journal* 6. 341-346.
Sievänen, L., Sievänen, M., Välikangas, K., Eloniemi-Sulkava, U. 2007. Opas ikääntyneen muistioireisen kodin muutostöihin. Helsinki: Ympäristöministeriö.

Sopanen, J. 2015. Elderly clients' experiences of playing a videogame. Bachelor's Thesis. Laurea University of Applied Sciences.

Stack Overflow Developer Survey 2020. 2020. WWW document. Available at: https://insights.stackoverflow.com/survey/2020#developer-profile-age-professional-developers5 [Accessed 13 September 2020].

Tun, P. A. & Lachman, M. E. 2008. Age Differences in Reaction Time and Attention in a National Telephone Sample of Adults: Education, Sex, and Task Complexity Matter. *Developmental Psychology*, 44(5), pp. 1421-1429. doi:10.1037/a0012845

Unity. 2019. *Key facts*. WWW document. Available at: https://unity.com/our-company#key-facts [Accessed 28 September 2020].

Unity Documentation. 2020. *Unity – Scripting API: Rigidbody2D.* WWW document. Available at: https://docs.unity3d.com/ScriptReference/Rigidbody2D.html [Accessed 28 September 2020].

Unity Manual: Unity Collaborate. 2020. WWW document. Available at: https://docs.unity3d.com/Manual/UnityCollaborate.html [Accessed 16 September 2020].

What is Trello? 2017. WWW document. Available at: https://help.trello.com/article/708-what-is-trello [Accessed 16 September 2020].

World Health Organization. *Ageing and health.* 2018. WWW document. Available at: https://www.who.int/news-room/fact-sheets/detail/ageing-and-health [Accessed 19 September 2020].