Faculty of Sciences and Technology

Department of Informatics Engineering

Playsketch

Game Design and Experience Evaluation

Fernando Miguel Cardoso Rocha

Internship Report in the context of the Master in Informatics Engineering, Specialization in Information Systems advised by Prof. Jorge Cardoso and presented to the Faculty of Sciences and Technology / Department of Informatics Engineering.

July 2019

1 2 9 0

UNIVERSIDADE Ð
COIMBRA

This page is intentionally left blank.

Abstract

The experience that a game provides to its players is the focus in the video game development process. Many video games are not successful in the market due to design flaws or playability and usability problems. In this project, an iterative design process of evaluation and correction of playability problems is carried out. The object of study in this process is a work-in-progress top-down 2D mobile racing game designed specifically for children ages six through twelve years old. The evaluations are performed in user-testing scenarios by playtesting sessions. These evaluations resulted in the detection of various playability faults such as the implemented control scheme and the lack of feedback when interacting with some game elements. Four new control schemes (acceleromter-based tilting, steering wheel, analogue joystick and joystick+button) were implemented and tested in a playtesting session. Results show that children prefer the joystick options as they provide better control over the player character. New features were also implemented such as a new game mode and a multiplayer server that can provide better replayability to the game. With the generated knowledge from the interactions with the evaluation participants, a prototype for a new mobile game was developed and shall be evaluated in future work.

Keywords

game design, game evaluation, playability, player experience, video games

This page is intentionally left blank.

Resumo

A experiência que um jogo oferece aos seus jogadores é o foco no processo de desenvolvimento de videogames. Muitos videojogos não são bem sucedidos no mercado devido a falhas de design ou problemas de jogabilidade e/ou usabilidade. Neste projeto, um processo de design iterativo de avaliação e correção de problemas de jogabilidade é realizado. O objeto de estudo neste processo é um jogo de corridas 2D para dispositivos móveis, desenvolvido especificamente para crianças entre seis e doze anos de idade. As avaliações são realizadas em cenários de teste de utilizadores por sessões de jogabilidade. Essas avaliações resultaram na detecção de várias falhas de jogabilidade, como a falta de domínio sobre o controlo existente e a falta de feedback ao interagir com alguns elementos do jogo. Quatro novos controlos (acelerómetro, volante, joystick analógico e joystick + botão) foram implementados e testados em uma sessão de teste. Os resultados mostram que as crianças preferem as opções de joystick, pois fornecem melhor controlo sobre o personagem do jogador. Novas características também foram implementados, como um novo modo de jogo e um servidor multiplayer que podem fornecer aos jogadores a vontade de jogar repetidas vezes. Com o conhecimento gerado a partir das interações com os participantes das avaliações, um protótipo para um novo jogo foi desenvolvido e será avaliado em trabalho futuro.

Palavras-Chave

avaliação de jogos, design de jogos, experiência de jogo, jogabilidade, videojogos

This page is intentionally left blank.

Acknowledgements

I would like to thank Prof. Jorge Cardoso, my department adviser, for helping me during the course of this project, providing me with insightful suggestions and guidance. I would also like to thank Pedro Machado Santa, my Playsketch adviser, for all the suggestions, corrections, and the unshaken availability to make me a better person and a better professional.

I would like to thank all my colleagues from Playsketch and Maria Adelaide for providing me with a fruitful and fun workplace, the opportunity to represent the company in a videogame convention, and for all the water-cooler conversation shenanigans: Luís Lucas Pereira, Filipa Pereira and António Seiça.

I would like to thank all my friends, and battle companions, who helped me to not give up and supported me while I needed the most: Sandra Silva, João Silva, Rui Dias, Eliana Carvalho, Maria Maia, David Sereno and all the members of the "Real República dos Cândidos".

I would also like to thank all the participants, and their parents, who helped me to evaluate the game as well as Prof. Ana Oliveira, Prof. Guida Lacerda and Agrupamento de Escolas da Mealhada to make the arrangements so that I could test the game with as many children as possible.

Last but not least, my family: my grandparents, mother, father, brothers and sister who stood by me at every step of this project and gave me tireless support.

From the bottom of my heart, thank you.

This page is intentionally left blank.

Contents

Acronyms

ECS Entity-Component-System. 43

F2P free to play. 13, 20

GDD Game Design Document. 6

GEQ Game Experience Questionnaire. 16, 36

HCI Human Computer Interaction. 13

HUD Head-Up Display. 18, 34

MDA Mechanics, Dynamics and Aesthetics. 8, 17

MP multiplayer. 17, 35

QA Quality Assurance. 7

RITE Rapid Iterative Testing and Evaluation. 15

UI User Interface. xiii, 14, 17, 18, 20

This page is intentionally left blank.

List of Figures

List of Tables

This page is intentionally left blank.

Chapter 1

Introduction

As of today, gaming, specifically video games, is the biggest form of entertainment worldwide, dethroning television, film and music services (Wijman [2018]). Gaming is an enormous part of our daily lives, either by playing a game ourselves or even by watching someone else play. Ranging from games focused on short game sessions to games that can make players spend hundreds of hours, making use of such elements as immerse storytelling, gorgeous art, beautiful music and challenging gameplay. There is a kind of video game for everyone, making them playable by children and older people alike.

Associated with the increase of video games as a means of entertainment, a report from Ofcom [2017][1] shows that over 66% of children with ages 5 and up play video games for over 7 hours a week. The study also shows that parents are more concerned than ever with the time their children are spending with mobile devices, specifically.

Besides the entertainment value of video games, they are also considered as a method of expression combining many creative disciplines such as music, visual art and narrative writing. With the advances in technology, to create a video game has become a more and more complex task, resulting in big teams with proficiency on many skills ranging from computer programming and animation to dialog and narrative writing.

Even though those technical skills are necessary to develop a video game, they are useless if the most important skill is absent in game creators: imagination. Having the technical capacities, a video game can be created with ease. That does not mean that the game will resonate with people and make them play it.

Imagination is a trait common to everyone, meaning that everyone can be a game designer. Exploring children's creative and imaginative side can have a huge impact on their well-being and development. So, if everyone can be a game designer, we need to explore that characteristic and make the designing of video games available to everyone who does not have the advanced technical skills necessary to create one from scratch.

With the idea that children need to develop their creative and imaginative skills, Playsketch is developing a collection of applications that encourage children to create their own video games.

[1] Ofcom is the regulator for the communication services in the United Kingdom

1.1 Playsketch

Playsketch is a tech-based startup company founded by Pedro Machado Santa and Luís Lucas Pereira and based in Instituto Pedro Nunes, Coimbra. Playsketch's mission consists on creating video games where the end-user is involved in designing and shaping them as their own.

The Playsketch product, also named Playsketch, is a collection of augmented and mixed reality mobile video games where end-user participatory game creation (Tavares and Roque [2007]) is the focus. Designed to appeal primarily to children ages 6 to 12, Playsketch's[2] goal is to encourage everyone to create its own video game.

playsketch
Imagine,
Draw
& Play!
Playsketch will enable everyone to create games by drawing them on paper. Use our mobile apps to capture your drawings and have them translated to playable games in minutes.
Instant fun with friends and family. Anywhere.
www.playsketch.net
playsketch

Figure 1.1: Playsketch Banner

The first video game from this collection is Playsketch Racing, a top-down[3] car racing game where the user is invited to create his own racing circuit by drawing it in a piece of paper (Pereira et al. [2014]).

This application can be divided into two main phases: the creation phase and the gameplay phase. The creation phase is composed by the drawing of the racetrack on a piece of paper, the photo capture of the drawing and, finally, the insertion of game elements into the circuit, as shown in Figure 1.2. Only one element is mandatory to begin the next phase: the finish line. After that, oil spills and acceleration tracks can be included into the level. Finishing

[2] Unless stated otherwise, future references to Playsketch are related to the product and not to the company.

[3] Top-down perspective refers to a camera angle that shows the player character and its surroundings from an aerial point of view

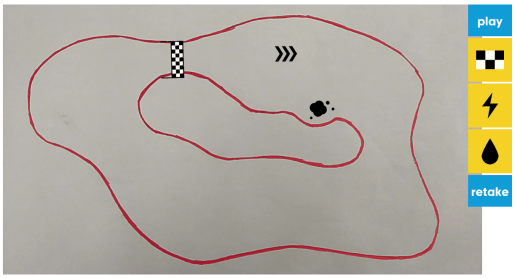


Figure 1.2: Creation Menu

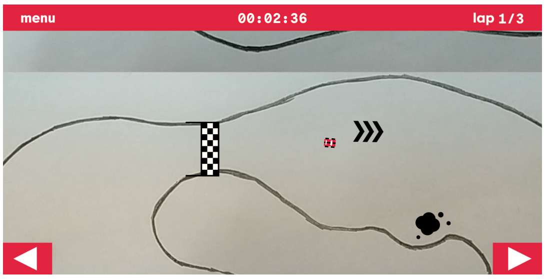


Figure 1.3: Playing the game

this task, the gameplay phase begins. This new phase has only one aim: play. Being a racing game, encourages the player to race the car in the created circuit, making three full laps in the minimum time possible.

1.2 Motivation

The Playsketch Racing video game has already been in contact with potential end-users in different occasions. Observing the interaction between them and the application has already proven to be very important and contributed to the detection of flaws. Some flaws are functional, solved with objective thinking and code analysis. For instance, if the colours used in the circuit's drawing are not dark enough to surpass a fixed threshold, the circuit is not recognised and the collisions with its borders will not be detected.

Despite their importance, these faults are not the most relevant ones. The most important flaws are the ones that cannot be detected using the typical software development processes such as code inspections or black box testing. These are the playability and player experience flaws, the faults that are not directly associated with the functional side of the video game, but instead what the players' experience when playing. For instance, one detected flaw is related with the controls: two directional buttons are used to turn the car left and right. The buttons work according to the specification and, when pressed, the car turns to the desired direction. Some children have difficulty with this control scheme, pressing one button for far too long so that the car is constantly turning the same way, making no progress.

To resolve this issues, playability evaluations must be conducted in order to fully understand the potential problems with the application and to determine possible solutions.

As stated in section 1.1, the game is composed by creation and gameplay phases. Even though there are flaws identified in both phases, this internship will be mainly focused in the latter, addressing problems with the gameplay and playability of the racing game, and not with the creation of the levels by the users.

After the evaluation of this specific video game is iterated and concluded, the generated knowledge will serve as a basis to the development of new game prototypes for the video game collection.

1.3 Objectives

The primary objective of this project is to develop the necessary skills and obtain essential knowledge for a master in Informatics Engineering to successfully integrate the working world. To consolidate and apply concepts learned during the previous years of the degree is also of major importance.

Beyond these primary objectives, the goals to be achieved with this internship are:

- To evaluate Playability and Player Experience of Playsketch Racing, using user testing scenarios;
- To implement improvements to detected problems and new features to Playsketch Racing;
- To continuously evaluate and develop Playsketch Racing
- To develop prototypes for new games to be added to the Playsketch collection.

1.4 Document Structure

The remaining of the document is structured with the following chapters:

- Background and related work - In this chapter is given a presentation about the process and the knowledge generated from investigating previous work related with game design, player experience and its evaluation, the game in study and similar mobile games;
- Approach - This section describes the resorted methodology to reach the internship objectives. It is also shown a list of risks associated with the project, the original work plan for the internship, the actual final work schedule and how it was developed over the period of the internship;
- Preliminary Evaluation - In this chapter, it is presented a user testing evaluation performed to understand the video game state when of the entering of the intern to the project;
- Implementation - Here, the implementation of improvements to the project is documented. A prototype for a new game is also presented;
- Second Evaluation - This chapter contains the description of the user testing evaluation performed during the second semester evaluation regarding the improvements implemented previously;
- Conclusions - This section concludes this document, revealing lessons learned about the project and the processes followed.

This document includes appendices with the informed consent form delivered to the participants of the evaluations carried out as well as the ra data from said playtesting sessions.

Chapter 2

Background and Related work

Beginning the project at hand, a Background and Related work research was conducted with the purpose of answering the following questions:

- What is a video game and what is involved in its development?
- How is playability and player experience characterised and how to evaluate it?
- What is the current state of the top-down racing mobile games genre?
- What is the current state of the Playsketch Racing product?

To answer the first question, some literature on game design and video game development processes was explored. This was very important to understand the techniques available to both designing a new video game as to improve a game that is already mid-production. The second question was tackled by the analysis of articles about playability and player experience. This research was conducted mainly to understand how these subjective terms can be evaluated through testing. On the third question, some games were gathered and analysed to understand what characteristics this kind of games have. Finally, to answer the last question, the product was presented to participants from the target audience in a technology fair[1]. With this informal playtesting "session", it was possible to better understand the objectives of the video game and also how the target audience may interact with it.

2.1 Game Design Overview

Salen et al. [2004] define a game as a *" (...) system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable outcome."*. Through this definition, it is said that a game is a system that has users that interact with it (players), a set of defined limits that are accepted by those users (rules) that end in measurable results and classified with the help of those limits (outcome).

Game Design is the art of defining all the above-stated characteristics of a game or, in other words, *"the act of deciding what a game should be."*(Schell [2014]). This means that, when designing a game, one must acknowledge the players that will interact with it and what are the rules that define both that interaction and the possible outcomes.

[1] Festival da Ciência, Oliveira do Bairro `https://feciob.wixsite.com/feciob2018`

If the player interaction with a game is made through a digital user interface, we say that said game is a video game. Figure 2.1 shows the complex process that is the interactivity between player and computer.

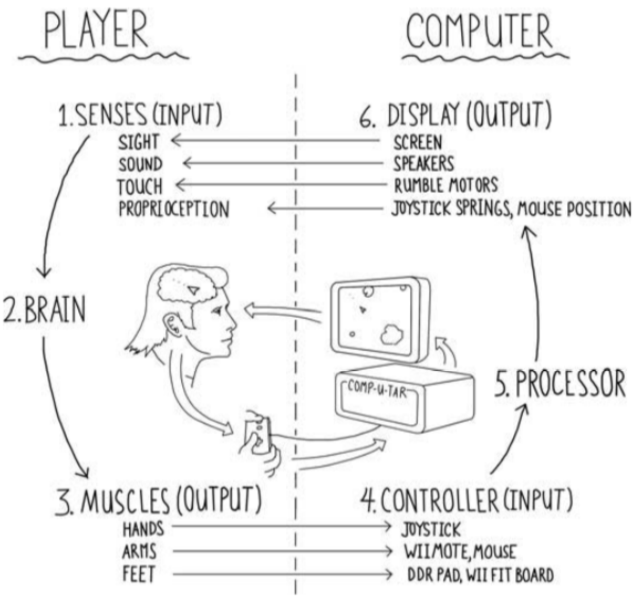


Figure 2.1: Interactivity between player and computer (Swink [2008])

The development of video games, much like any other kind of software development, is composed by multiple phases, following a waterfall-like process. These phases are Concept, Pre-Production, Production and Post-production (Martinho et al. [2014]), described in section 2.1.1.

2.1.1 The Making of Video Games

The Concept phase begins with the conceptualisation of an idea for a video game. The aim is to define the high-level characteristics of the game by developing a Game Design Document (GDD) that describes the innovative aspects of the game and how the player should interact with it. An early prototype showcasing what is described in the GDD should also be developed. These two artifacts, GDD and prototype, should be used as means to convince someone with decision-making power to invest in the game. In later years, video games also have been funded through crowdfunding activities. With this approach, it is possible to ascertain if people are really interested and willing to fund the development of the video game. With the concept approved, the game moves into the Pre-Production phase.

In the Pre-Production phase, the objective is to fully develop the GDD and the planning for the entire development cycle. It's in this phase that the design of the game should be completely detailed. Details like mechanics, characters, story, challenges, rewards, among others must be defined in this phase, even though it might be necessary to rethink some of those in the following phases. In this phase, there is the need to bring other skills into the development team such as programmers, writers and artists, besides the original game designer. So, it is necessary to define the game's art style and development platform. The development platform should be selected with the intended target platforms for the game in mind. As an outcome of this phase, it is expected that a fully detailed GDD, a technical design document, work plan and budget document and a more detailed prototype to be delivered, entering the Production phase.

The Production phase is the most time-consuming phase of the development cycle, it is the phase where the game is going to be fully developed. The process for this phase should be iterative and focused at the development of a playable prototype at every iteration cycle. At every cycle, the prototype should be developed, tested, evaluated and a new iteration takes place. This method is called Iterative Design, and it is explained in more detail in section 3.1. This phase has three milestones, with each one corresponding to a game version:

1. **Alpha:** at this point the game should be playable from start to finish. Some elements like art assets might not be final and some bugs might be present, but the engine and user interface should be complete. Instead of building the game, the alpha stage is about polishing the game. This version must be tested intensely in order to check if the gameplay is perceived as intended in the Concept phase.

2. **Beta:** this version of the game should be faced as the last version of the game. All programming bugs should be corrected and visuals should be final. This version should be polished enough that it only needs to tune playability problems. Usually, this version is tested in two stages: privately, with a small and controlled group of players, or publicly, with a bigger and less controlled group of players.

3. **Gold**: the game is considered as final and should be ready to launch and the Post-Production phase begins.

In the Post-Production phase, the game is ready to launch but some details might still be missing. This involves passing the target platforms quality control, usually with game consoles. The game must be registered in a video game content rating system. The game should also pass a step of localisation, where the contents are translated to other languages and the adaptation of other details in case of cultural differences. Post-launch, the game can also receive updates due to reviews.

2.1.2 Testing Video Games

As seen in the previous section, the testing and evaluation of the video game are present in all the development phases, making them top priority specially during the Production phase. Schell [2014] distinguishes four types of testing: Focus Groups, Quality Assurance (QA) Testing, Usability Testing and Playtesting.

Focus Groups refers to the interviewing of potential players about their game preferences, in order to determine if a game idea is worth of being developed. This strategy is used early in the development cycle, usually at the Concept phase.

QA Testing is usually done by a group of specialised members of the game development team in order to encounter as many bugs as possible, just like any other type of software QA department. Following Nacke [2010] Game Usability Model, QA *"refers to numerical evaluations of technological functionality"*.

Usability Testing has the objective to determine if the user interface of the video game is intuitive and easy to use. According to ISO 9421-11:2018, usability is defined as the "extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use". Although it is indispensable for an enjoyable game, usability alone is not enough.

Playtesting refers to the gathering of testers to play the game so to see if their experience is the same as envisioned by the game designer. Playtests are the main concern during the production of a video game, as it is the primary means of evaluating the state of the game and if the experience designed matches the experience the players are having.

Schell [2014] states that the game, either its physical or digital object, is not the final goal of the game design activity but the experience that said game gives to a player instead.

2.2 Video Games as Artifacts to Experiences

A game designer designs games but, more importantly, focuses on designing experiences. So, we can conclude that the games are mere artifacts to achieve the greater objective of the design process. This objective is called the Player Experience.

Martinho et al. [2014] define player experience as the *"set of sensations that the game provides (to the player) during its usage, that involves the realisation of activities, and the memories that stay after the utilisation"*. The experience can never be forced into the player, as it varies from the individual regarding their knowledge, motivations and preferences. So, the experience is an interpretation, made by the player, regarding the playing of a game.

2.2.1 Player Experience Models

With such as a subjective term as player experience considered of major importance in the game design area, some authors proposed models as to characterise it and to guide game designers to designing better experiences.

Mechanics, Dynamics and Aesthetics

Mechanics, Dynamics and Aesthetics (MDA) is a framework proposed by Hunicke et al. [2004] that formalises games and their consumption by dividing them with distinct components: Rules, System and "Fun" and establishing their design counterparts: Mechanics, Dynamics and Aesthetics, as shown in Figure 2.2.

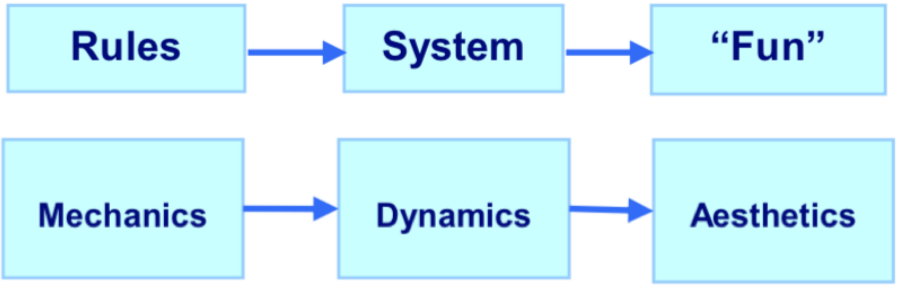


Figure 2.2: The components of game consumption and their design counterparts

Mechanics represent the "*particular components of the game, at the level of data representation and algorithms* that combined with Dynamics, *"the run-time behavior of the mechanics acting on the player inputs and each others outputs"* to evoke *"the desirable emotional responses"*, represented by Aesthetics.

This framework considers both the game designer and the player perspectives, shown in Figure 2.3. From the designer's perspective, the mechanics lead to dynamic system behaviour that subsequently leads to particular aesthetic experiences. While from the player's perspective, aesthetics set the tone, leading to observable dynamics and eventually, to operable mechanics. It is most useful to consider both perspectives when developing a game as it can show us how the player dynamics and aesthetics can change through changes on the mechanics.

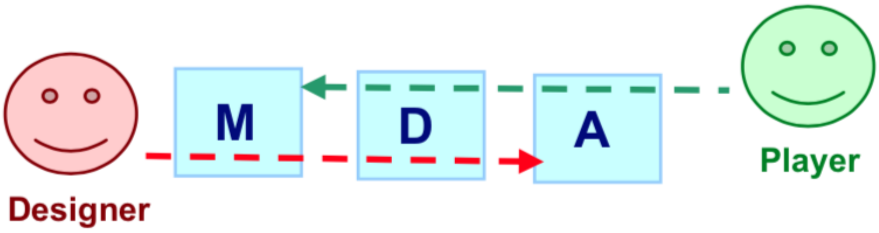


Figure 2.3: Designer and Player perspectives

The authors also list eight types of aesthetics which they feel depict some of the components of "fun":

- Sensation, game as sense-pleasure;
- Fantasy, game as make-believe;
- Narrative, game as drama;
- Challenge, game as obstacle course;
- Fellowship, game as social framework;
- Discovery, game as uncharted territory;
- Expression, game as self-discovery;
- Submission, game as pastime.

GameFlow

GameFlow is a model proposed by Sweetser and Wyeth [2005] with the objective of evaluating player enjoyment in games based on Csikszentmihalyi and Csikszentmihalyi [1975] theory of Flow. Flow describes the state of mind characterised by loss of self-consciousness, distorted perception of time and a host to pleasurable sensations that exists between the sensations of boredom and anxiety. As shown in Figure 2.4, the Flow state occurs when both the person's skills and the challenges of the activity are balanced. Harder challenges for a non-skilled person will provoke worry and anxiety while easier challenges on a skilful person will raise sensations of boredom.

By adapting the elements of flow, Sweetser and Wyeth [2005] created eight elements to the GameFlow model: Concentration, games should require concentration and the player should be able to concentrate on the game; Challenge, games should be sufficiently challenging and match the player's skill level; Player Skills, games must support player skill development and mastery; Control, players should feel a sense of control over their actions

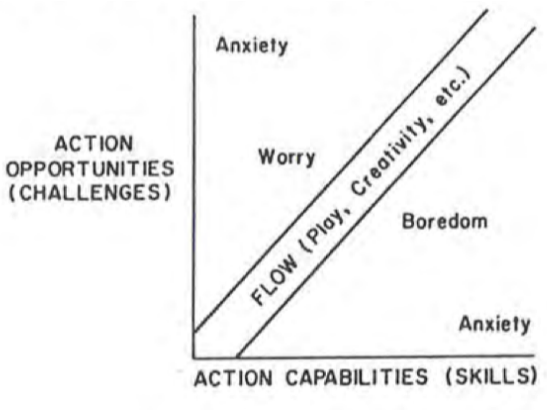


Figure 2.4: Csikszentmihalyi and Csikszentmihalyi [1975] Theory of Flow

in the game; Clear Goals, games should provide the player with clear goals at appropriate times, Feedback, players must receive appropriate feedback at appropriate times; Immersion, players should experience deep but effortless involvement in the game; and Social Interaction, games should support and create opportunities for social interaction.

Each element includes a set of central criteria that can be used to design and evaluate games regarding player enjoyment such as *"game interfaces and mechanics should be easy to learn and use"* and *"games should provide a lot of stimuli from different sources"*.

Participation-centered Game Experience Design

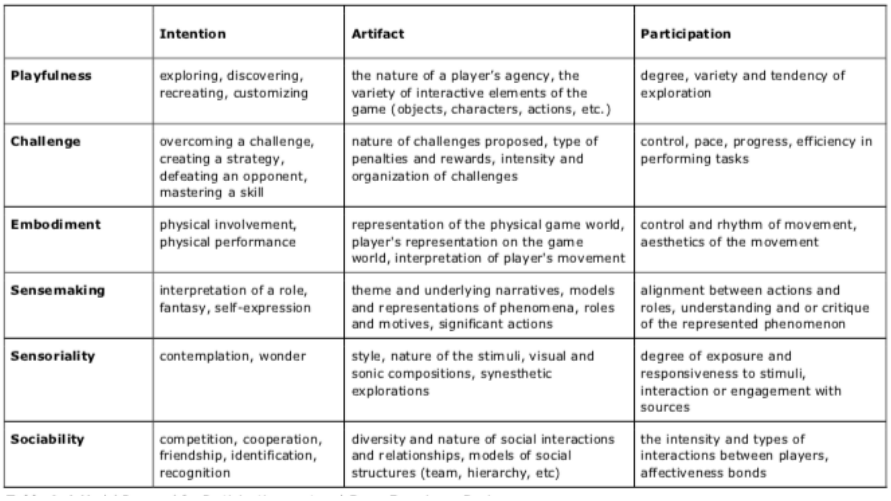
Pereira and Roque [2012] proposed a model for game design and experience evaluation based on the concept of *"Play is experienced through participation. When a player interacts with a game, the formal system is manifest through experiential effects."* (Salen et al. [2004]). This directly relates with the idea that video games could be designed with player participation in mind.

This model consists on three operational levels which work as three key points of the design process: Intention, the proposed forms of participation by the game designer and the kind of experiences that are enabled; Artifact, the way that the video game supports the intent of the game designer; and Participation, the characteristics of the actual player activity and their relation with the initial forms of participation proposed.

As to characterise players' participation in each one of those levels, Pereira and Roque [2012] enunciate six perspectives in participation, as seen in Figure 2.5: Playfulness, participation as a free-form, spontaneous, child-like type of play; Challenge, participation as assessed by the performance of the player in overcoming challenges, within the purpose of the game; Embodiment, participation as a physical relationship between the player and the video game; Sociability, participation as a basis on the establishment of relationships between players; Sensemaking, participation as expression and creation of meaning; and Sensoriality, participation as basis on the engagement of perception, filtering, acceptance or reproduction of stimuli.

Playability Model

The player experience is greatly influenced by the playability of a game. As though playability does not have a standardised definition like usability has, Sánchez et al. [2009] under-



	Intention	Artifact	Participation
Playfulness	exploring, discovering, recreating, customizing	the nature of a player's agency, the variety of interactive elements of the game (objects, characters, actions, etc.)	degree, variety and tendency of exploration
Challenge	overcoming a challenge, creating a strategy, defeating an opponent, mastering a skill	nature of challenges proposed, type of penalties and rewards, intensity and organization of challenges	control, pace, progress, efficiency in performing tasks
Embodiment	physical involvement, physical performance	representation of the physical game world, player's representation on the game world, interpretation of player's movement	control and rhythm of movement, aesthetics of the movement
Sensemaking	interpretation of a role, fantasy, self-expression	theme and underlying narratives, models and representations of phenomena, roles and motives, significant actions	alignment between actions and roles, understanding and or critique of the represented phenomenon
Sensoriality	contemplation, wonder	style, nature of the stimuli, visual and sonic compositions, synesthetic explorations	degree of exposure and responsiveness to stimuli, interaction or engagement with sources
Sociability	competition, cooperation, friendship, identification, recognition	diversity and nature of social interactions and relationships, models of social structures (team, hierarchy, etc)	the intensity and types of interactions between players, affectiveness bonds

Figure 2.5: Pereira and Roque [2012] Participation-centered Model

stand it as *"the degree to which specified users can achieve specified goals with effectiveness, efficiency and specially satisfaction and fun in a playable context of use"*. This definition is close from the one used to describe usability, as seen in the previous section, but with a major difference, playability involves a fun attribute in a context of play.

For Sánchez et al. [2009], playability is characterised regarding 7 attributes: Satisfaction, gratification or pleasure derived from playing; Learnability, player's capacity to understand and master the mechanics; Effectiveness, time and resources necessary to offer players a fun experience; Immersion, capacity of the video game contents to be believable; Motivation, characteristics that prompt a player to realise specific actions; Emotion, player's involuntary impulse in response to the video game; and Socialization, game attributes that promote the social dimension. Every one of these attributes is then characterised by a set of properties, depicted in Figure 2.6.

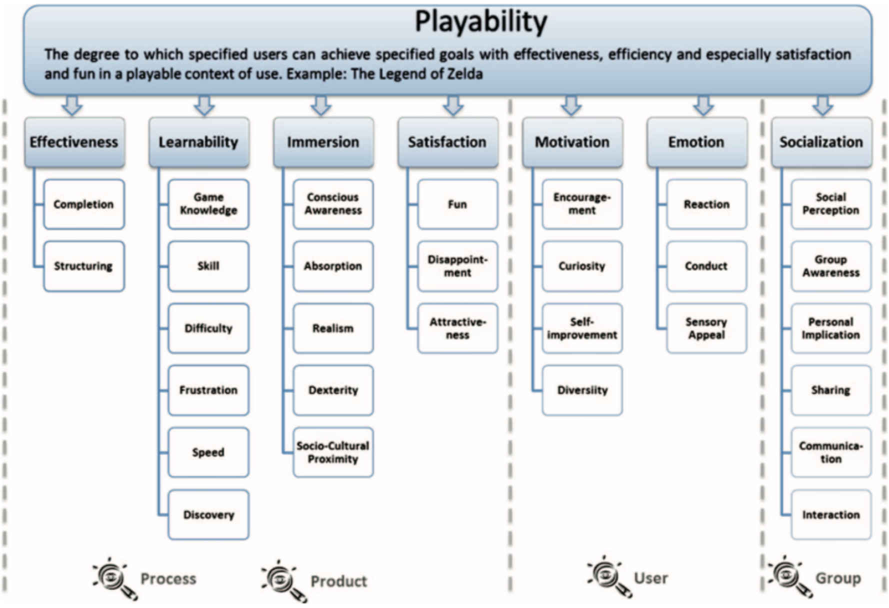


Figure 2.6: Sánchez et al. [2012] Playability Model

Sánchez et al. [2012] propose a classification model based on different perspectives of playability called "Facets of Playability". Each facet allows the identification of the attributes and proprieties of playability that are influenced by the player-game interaction process. The six facets of playability are:

1. **Intrinsic Playability:** represents the nature of the video game itself and how it is presented to the player. This is related to the game core: rules, challenges, rewards and goals.

2. **Mechanical Playability:** it is related to the quality of the video game as a software system. It is associated with the game engine: fluency of graphics, rendering, artificial intelligence and software routines.

3. **Artistic Playability:** it is related with the quality of the artistic and aesthetic rendering of the game elements like visual graphics, music and story

4. **Interactive Playability:** this facet is strongly connected with the game interface and associated with the video game user interface and player interaction.

5. **Intrapersonal Playability:** also called Personal Playability, represents the individual perceptions and feelings that the video game evokes in the players.

6. **Interpersonal Playability:** or Social Playability, this refers to feelings and perceptions of the players, when playing in a group context.

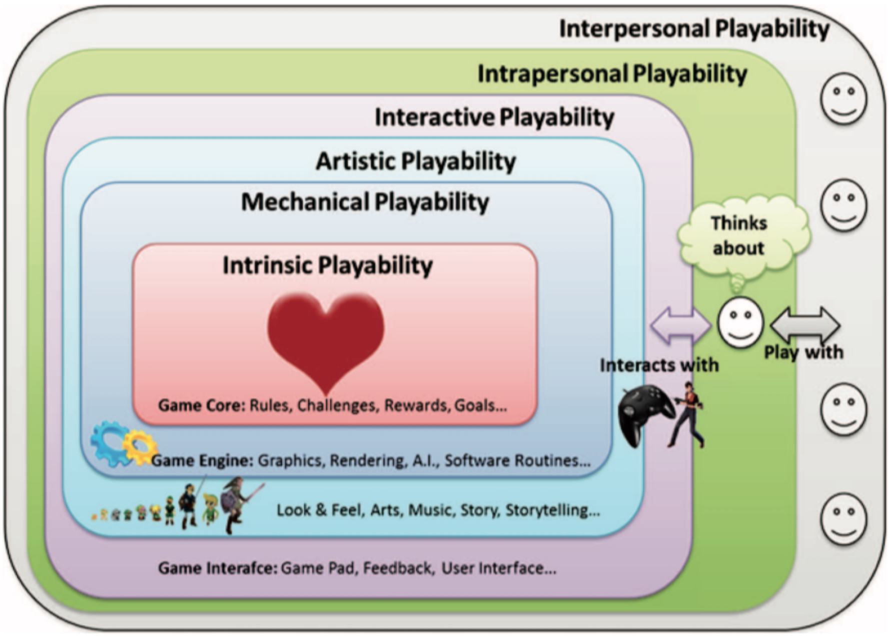

Interpersonal Playability
Intrapersonal Playability
Interactive Playability
Artistic Playability
Mechanical Playability
Intrinsic Playability
Game Core: Rules, Challenges, Rewards, Goals...
Game Engine: Graphics, Rendering, A.I., Software Routines...
Look & Feel, Arts, Music, Story, Storytelling...
Game Interafce: Game Pad, Feedback, User Interface...
Thinks about
Interacts with
Play with

Figure 2.7: Sánchez et al. [2012] Facets of Playability

Using the Facets of Playability and the Playability models, it is possible to elaborate playable requirements where each attribute of playability and its related game element is analyzed in order to obtain a positive experience. For instance, the playable requirement "Tutorials in the initial level" is associated with rules and mechanics and related with the Intrinsic Playability facet and the Learnability attribute of Playability (Sánchez et al. [2012]).

2.2.2 Playability and Player Experience Evaluation

Using the previous models, evaluation of playability and player experience in video games can be quantified when the data is available or through theoretical inspections of the artifact. But, in a practical environment, data must be collected in order to model both of those characteristics. The collection of the necessary data can be fulfilled using methods like Playability Heuristics Evaluation or Playtesting (Desurvire and El-Nasr [2013]).

Playability Heuristics

Heuristic evaluation is an usability inspection method used by practitioners and researchers of Human Computer Interaction (HCI) and consists in experts inspecting the target system using a set of guidelines to govern the evaluation. This technique was then adapted for game user research in order to evaluate not only the usability of a video game but also it's playability.

Playability heuristics first appeared when Federoff [2002] conducted a case study in a game company and defined heuristics considered as the first heuristic set for evaluating video games. In the following years, many playability heuristic sets have been developed concerning various types and attributes of video games (Desurvire et al. [2004], Desurvire and Wiberg [2009], Pinelle et al. [2008, 2009]).

Korhonen and Koivisto [2006] firstly developed a set of playability heuristics that was later expanded to cover different aspects of the game (Korhonen [2016]). The two core modules of that set reflect what the author thinks are the most important aspects of playability: game usability and gameplay, shown in Table 2.1. These heuristics are supposed to be common to all games and can be used to evaluate any type of game. Other modules have been added regarding multiplayer, mobile, context-aware and free to play (F2P) games.

Code	Gameplay Heuristics
GP1	the game provides clear goals or supports player-created goals
GP2	the player sees the progress in the game and can compare the results
GP3	the players are rewarded and the rewards are meaningful
GP4	the player is in control
GP5	challenge, strategy, and pace are in balance
GP6	the first-time experience is encouraging
GP7	the game story, if any, supports the gameplay and is meaningful
GP8	there are no repetitive or boring tasks
GP9	the players can express themselves
GP10	the game supports different playing styles
GP11	the game does not stagnate
GP12	the game is consistent
GP13	the game uses orthogonal unit differentiation
GP14	the player does not lose any hard-won possessions

Table 2.1: Gameplay Heuristics (Korhonen [2016])

The heuristic evaluation procedure can be divided into five phases (Paavilainen et al. [2018]): Preparation, Individual evaluation, Debrief with inspector team, Report findings and Aftermath.

The Preparation phase consists in the choice of inspectors and the choice of the heuristics set to use. Multiple inspectors should be included so as to compare their findings and to achieve the best outcome possible from the evaluation. Optimally, these inspectors should be both experts on gaming and the method of heuristics evaluation. With the lack of double experts, the options are to select a mix of experts on gaming and experts on the method or to select a large group of novice inspectors with interest on gaming and the evaluation method. The chosen heuristics should cover multiple aspects of the video game, with larger focus on the usability and gameplay core aspects.

The next phase consists in the inspectors individual evaluation of the game with the help of the chosen heuristics. The inspectors play the game, observing the design and taking notes on aspects of the game that might cause playability problems by violating the heuristics. In the case that problems not related with the heuristics arise, they should be reported as well. The evaluation can be divided into three rounds: the first round is dedicated to explore the interface elements external to the actual gameplay such as menu, configuration and settings screens. The second round concentrates on the gameplay, focusing on whether the game is understandable and behaves as expected. In the final round, the inspectors focus on the User Interface (UI) and how wit supports the gameplay, by providing accurate and sufficient information for the player. Besides playability problems, positive findings should be included in the outcome report to motivate developers and to prevent the alteration of positive factors in order to fix the problems encountered.

In the Debrief with inspector team phase, the inspectors work together to merge a master list of playability problems based on the individual findings. Every item on the list should include a description of the problem, the location of it in the game, why it is considered a problem and what heuristic was violated, if any, and a proposed solution.

Next, all the findings are reported to the development team and further discussed. This discussion consists to help the developers understand the problems and also to discern if the heuristics are violated on purpose, regarding the game design.

The final phase, Aftermath, consists on the inspectors reviewing the carried out process, analysing how well the evaluation covered the necessary aspects if met the evaluation objectives.

An important note is that the inspectors rarely represent the target audience of the game, both by not representing the target demographic and by playing the game with the finality to evaluate instead of enjoying it as a regular player would. Focusing on evaluating the game can produce false positive findings, which might not be actual playability problems for the players. With this in mind, heuristic evaluation should not be considered an alternative to playtesting, but as a complement during the entire development process.

Playtesting

Playtesting is the most used game testing method regarding game companies (Rajanen and Nissinen [2015], Rajanen and Tapani [2018]). Playtesting involves the gathering of a large group of testers with the objective to play the game, gaining useful feedback from players to improve the overall experience of the game. Playtesting should accompany the process of game development as soon as possible, as player feedback might be tremendously helpful to avoid bad core design. Later on, playtesting might help to understand when the game is considered ready to launch.

Fullerton [2014] presents a series of playtesting tips to help game designers improve the process. First, the recruiting of playtesters varies according to the game state. If the game is in its early stages with only a crude prototype developed, self-testing is a valuable method to understand how the game fundamental concepts work. If the development team is composed by over one person, each one of them will self-test both as a group and as individuals. Self-testing will continue throughout the whole project but, as it evolves, it is necessary to bring outside testers to give an accurate understanding of the project. Afterward, the people that might test the game are relatives and friends of the developers. These people bring new perspectives to the project as they are seeing it for the first time and will discover things that the developers might have not considered. The prototype should already be playable and the UI should be intuitive enough so that the developers do not have to explain to the testers what they have to do. This type of testers are good in early stages but have reserves about giving the developers the criticism they might need to make a better game. In result of that, it is necessary to test the game using people with no relationships with both the project and the development team.

Playtesting can be adapted to better fulfill the objectives, for instance, Rapid Iterative Testing and Evaluation (RITE) is a method developed by Medlock et al. [2002] during the testing of the first Halo game. This method consists in testing the game using a think-aloud methodology and every time a problem is discovered, it is immediately fixed with testing resuming after at one fix.

A typical playtesting session is composed by four main stages (Fullerton [2014]), excluding the final wrap-up phase:

1. **Introduction:** this stage begins the process of playtesting. It serves the opportunity for the responsible to introduce himself, thank the participants and explain to them how their contribution will help make a better game. If necessary, it is in this phase that the responsible must warn the participants they are being recorded, either video or audio, and deliver consent forms.

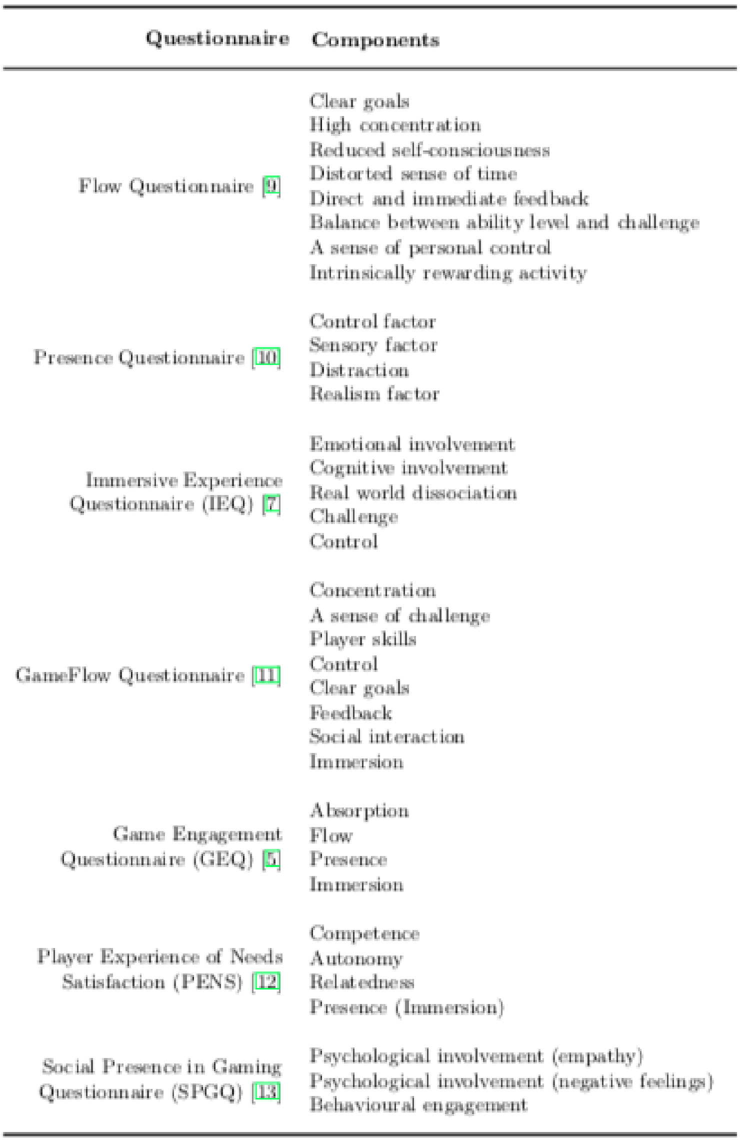
2. **Warm-up Discussion:** this phase serves to better understand the kind of players the participants are, specially if they are costumers to games related to the one being evaluated.

3. **Play Session:** in this stage is where the actual action of play occurs. The intention is to leave the participants playing the game without the responsibles‘ help. Observers might be present in the room or observing through a one way glass. Either way, their job is to take notes, guarantee that the players are involved in the act of playing and to remind them to keep thinking out loud about their choices while playing.

4. **Discussion of Game Experience:** finishing the play session, it is necessary to inquiry the participants in order to understand their perception of the game. The most used instruments to collect this data are one-to-one or group interviews and individual questionnaires.

Questionnaires

Relatively to collecting data from playtesters, there are many proposals of questionnaires to evaluate the experience of playing video games. Nordin et al. [2014], in their article titled *"Too Many Questionnaires: Measuring Player Experience Whilst Playing Digital Games"*, analyse the available questionnaires developed at the time and categorise every one using attributes that influence game enjoyment. Those questionnaires and respective categories can be seen in Figure 2.8, illustrating the table originally presented in Nordin et al. [2014] work.



Questionnaire	Components
Flow Questionnaire [9]	Clear goals High concentration Reduced self-consciousness Distorted sense of time Direct and immediate feedback Balance between ability level and challenge A sense of personal control Intrinsically rewarding activity
Presence Questionnaire [10]	Control factor Sensory factor Distraction Realism factor
Immersive Experience Questionnaire (IEQ) [7]	Emotional involvement Cognitive involvement Real world dissociation Challenge Control
GameFlow Questionnaire [11]	Concentration A sense of challenge Player skills Control Clear goals Feedback Social interaction Immersion
Game Engagement Questionnaire (GEQ) [5]	Absorption Flow Presence Immersion
Player Experience of Needs Satisfaction (PENS) [12]	Competence Autonomy Relatedness Presence (Immersion)
Social Presence in Gaming Questionnaire (SPGQ) [13]	Psychological involvement (empathy) Psychological involvement (negative feelings) Behavioural engagement

Figure 2.8: Questionnaires measuring player experience

Nordin et al. [2014] state that many of the existing questionnaires poses many challenges to new researchers, as the theory behind them relies on specific details, for instance, Control in the Immersive Experience Questionnaire might not be defined as in the GameFlow Questionnaire. Also, many of these questionnaires are not publicly available or are not considered reliable or trustworthy due to missing statistical validation.

One of the questionnaires not addressed in that study is the Game Experience Questionnaire (GEQ) (IJsselsteijn et al. [2013]).

The GEQ consists of a total of 64 both positively and negatively worded questions answered on a 5-point Likert scale and is divided into three modules, the first two try to understand the player's feelings and thoughts while playing the game while the last assesses how players felt after they had stopped playing. These modules are:

1. **The core questionnaire:** this module contains the principal part of the GEQ and assesses game experience as scores on seven components: Immersion, Flow, Competence, Positive and Negative Affect, Tension and Challenge. Even though

the components are not defined by the authors, the related questions give us a fair understanding of what each one means. For instance, some questions regarding the Tension component are *"I felt annoyed"* and *"I felt irritable"*, giving the impression that Tension in this context is considered a negative element of the experience.

2. **The Social Presence Module:** this module investigates psychological and behavioural involvement of the player with other social entities, be they virtual, as artificial intelligence players, mediate, as online players, or co-located, as present players.

3. **The Post-game module:** this module evaluates how players felt after they stopped playing. This is important to understand why players started a play session and why did they ended it.

2.3 Playsketch Racing

Playsketch Racing is the first of a collection of mixed reality video games developed by Playsketch. It is a top-down racing game targeted to mobile platforms with the intention of making the players design their own video game by drawing the game world on a piece of paper. Following Hunicke et al. [2004] MDA model, Playsketch Racing Aesthetics is mainly focused in the Discovery and Expression elements, derived from the usage of the game as a creational tool with which the players design and play something completely new every time. Elements of Challenge are also present, as players might tend to create bigger worlds with more obstacles to feel the enjoyment of mastering both the skills of designing as of controlling the player character, as well as Fellowship, with the game having a multiplayer (MP) option.

Figure 2.9: Playsketch Racing Main Menu

The game consists in two main phases, the creation phase and the gameplay phase. In the creation phase, the player takes a photograph of their drawing in order to the game to turn it into a racing track with solid objects, and includes game elements such as acceleration tracks and oil spills.

The drawing is then captured and a binarization algorithm is applied to it. The new binarized image is then transformed into a texture with the pixels below the defined threshold becoming solid objects to the player character to collide with. Both the original and binarized images are inserted into the scene overlapped, so the player sees his creation and also how to game is interpreting it.

The insertion of game elements is done with a UI menu, depicted in Figure 2.10, that spawns the objects into the centre of the scene which can then be dragged by the player into their desired places.

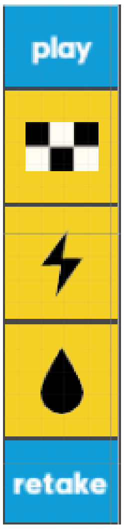


Figure 2.10: Insertion of game elements UI

The top button "play" finishes the creation and customisation processes and begins the gameplay phase of the video game. The top yellow button represents a finish line, the element that states both the starting place of the race and the end after the completion of three laps around the track. The next button spawns an acceleration track object for the player to interact with. This element gives the player a speed boost of two times the normal speed for every second that the player is interacting with it. Following that, the next button spawns an oil spill object that reduces the car velocity and also increases the turning speed. Finally, the "retake" button can be used to take another picture.

The gameplay phase begins when the player finishes customising his newly created world. In this phase, the player is invited to make three laps around his racetrack. Each lap begins and ends where the finish line is positioned. The UI in this phase consists of two main components: the game inputs and the Head-Up Display (HUD).

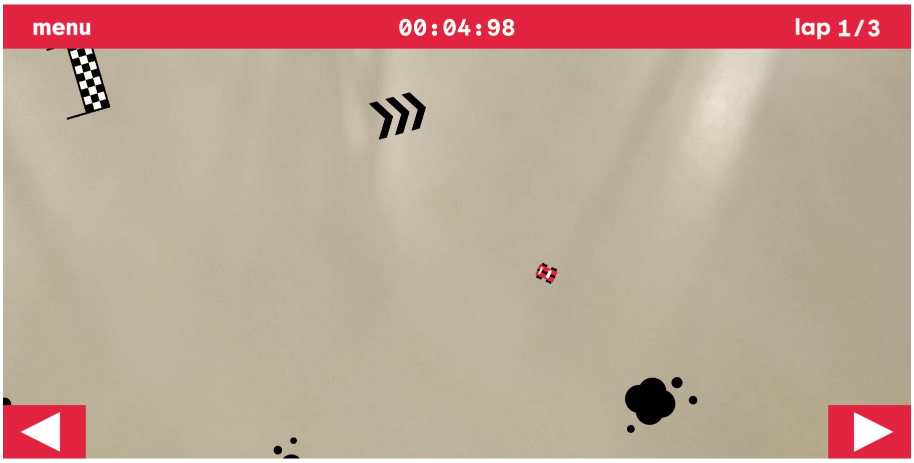


Figure 2.11: Playsketch Racing Game UI

The HUD, the top section of Figure 2.11, is composed by three elements: a "menu" button that, when pressed, returns the game to its main menu, as shown in Figure 2.9; a timer that gives feedback on the player related to the time he spent racing the racetrack; and a lap counter that gives the player information about his progress regarding the racing objective.

The game input, in the bottom section of Figure 2.11, is composed by two directional buttons that turn the player character left and right, respectively. These are the means of player interaction with the game world.

Festival da Ciência

In October of 2018, the Playsketch team was invited to participate in *"Festival da Ciência“* science and technology fair where Playsketch Racing would be in contact with visitors coming from the area primary schools. This contact was observed by the intern as a means of understanding how do potential players interact with Playsketch Racing in it's current state. Around 40 children aged 8 to 9 interacted with the game, starting by the drawing of their racetracks, their capture into the game and further customization and ending in a play session.

The company setup, as depicted in Figure 2.12, was conceived with three children playing simultaneously at any given time. As the visitors came in field studies from schools, the majority of the interactions were realised by ten participants at a time.

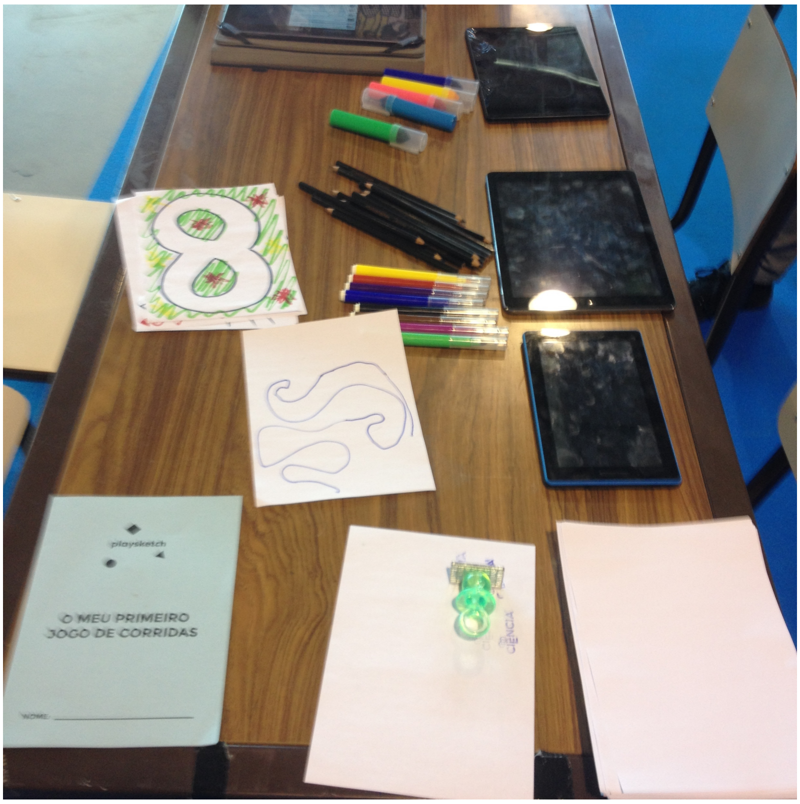


Figure 2.12: Playsketch setup in Festival da Ciência

Even though the experience proved to be positive for the vast majority of the players, requesting to play again and asking where they could get the game, there were a list of observed problems in this real context of use. These problems are divided into four phases of the game: Drawing of the racetrack, Capture of the drawing, Insertion of game elements and overall Playability. The first three phases will not be addressed in the internship's context, as its main goal is to solve the playability in the gameplay phase.

For the Drawing of the racetrack phase, the detected problems were:

- The racetracks can be designed with insufficient space for the car to move;
- The racetracks can be designed with a starting point and an ending point that do not have the same position, making impossible to make laps;
- Without the help of the developers, usually players drawn the finish line and game elements that can only be interpreted by the game as solid objects to collide with;

For the Capture of the drawing, the following problems were detected:

- The intensity or colour of the drawing greatly influences the way it is captured by the game. If the drawing is too light, the picture will not pass the binarization threshold and no solid objects will appear in-game;
- Shadows overlapping with the drawing make the game recognise more solid objects than supposed;
- Tablet orientation can make the camera flip horizontally;

Insertion of game elements:

- Some players try to drag the elements in the UI into the racetracks instead of clicking them;
- Touch-based problems make the selection and manipulation of already inserted game elements too difficult, as the selection of one game element might overlap with the selection of another;
- Without proper localisation of the game, with only the English language available, portuguese testers clicked on the “retake” button instead of the “play” button.

Playability:

- The main confusion in the gameplay phase is the controls. Most testers tried to tilt the tablet to make the car turn, some pressed only one button and others pressed both directional buttons randomly. This makes the participants feel frustrated because they perceive the game too hard for them to handle.
- The game world limits do not have collisions. This results in some players, by exploiting the problems from the previous phases, getting out of bounds and being unable to return inside the game world limits.

2.4 Top-Down Racing Mobile Games

To understand how the top-down racing genre is recently being explored, a review of randomly selected mobile games is conducted. This analysis will help us understand what is the consensus in the game development community on this genre, as well as to perceive what solutions might already be implemented to possible flaws with Playsketch Racing.

For this research, 10 video games were selected from both Apple's App Store and Google's Play Store. These games were chosen via an online search on specialty websites[2] and with the requirement of being F2P. Three of the 10 video games were posterior discarded as they had different objectives than the ones that Playsketch Racing is trying to achieve. For example, Paper Racing is a turn-based racing game, instead of a real-time one. This means the player has a period of analysis before making its move instead of continuously controlling the car over time.

The final 7 analyzed mobile games were the following:

- Doodle Kart (J2sighte [2010])
- Head to Head Racing (Craigs games [2013])
- High Octane (Fry-up Productions [2017])
- Micro Racing (Geemzo [2012])
- Mini Turbo GP (RedBit Games [2014])
- Retro Racing (Mr Qwak Limited [2015])
- Vs.Racing 2 (MdLabs AB [2015])

The analysis of the selected mobile games is done involving four categories:

- **Input:** Input are the methods available for players to interact with the game, or, as Swink [2008] defines, "the instrument of expression for the player into the game world". This is very important because it is necessary to understand what kind of inputs give the player the best sensation of "being in control" of the action. This category is subdivided in 3 attributes: Buttons, Tilt and Others. **Buttons** makes use of the touch screen as a metaphor for the buttons on a typical controller, specifically the input signal comes as a boolean value, either a button is pressed or not. **Tilt** utilizes the mobile device's accelerometer in other to understand its orientation, making the character move, or in this case turn, by the player rotating the device. As though the previously mentioned control schemes are more commonly used, other control schemes have been detected with this research, making it necessary to add a third option, **Others**.
- **Automatic (Auto) Acceleration:** Acceleration is the rate of velocity of an object with respect to time. In racing games, automatic acceleration means that the user needs not to interact with the game for the character to move forward. With this kind of games, especially with ones designed specifically for children, this attribute greatly affects the way the player interacts with the game. Without auto acceleration, it is necessary to incorporate the acceleration into the control scheme, greatly affecting the controls by adding another mechanic requiring to be mapped.
- **Camera:** Camera depicts how the action is presented to the player. Is it more important for the player to have the best visualisation of the character they are controlling (Character focused), or is it better to show the world that the player is interacting with (Full Track)?

[2] `http://android.qualityindex.com/charts/19033/top-down-racers-on-android`
`http://appcrawlr.com/app/search?q=top+down+racer&device=android&price=Free`
`http://ipad.qualityindex.com/charts/19029/top-down-racers-on-ipad`
All accessed on 23 October 2018

- **Game Modes:** if one video game has multiple ways to play and different goals to achieve, we say that it has various game modes. This category verifies if the current racing games only have one specific way of playing and achieving those goals, or if the implementation of multiple ones is a common practice, for instance, singleplayer and multiplayer modes, time challenges[3], and so on.

Game	Input			Auto Acceleration		Camera		Game Modes	
	Buttons	Tilt	Others	On	Off	Full Track	Character Focused	Single	Multiple
Doodle Kart	*X*	*X*		*X*	*X*		*X*		*X*
Head to Head Racing	*X*	*X*		*X*			*X*		*X*
High Octane	*X*		*X*	*X*	*X*		*X*	*X*	
Micro Racing	*X*			*X*		*X*		*X*	
Mini Turbo GP	*X*	*X*	*X*	*X*	*X*		*X*	*X*	
Retro Racing	*X*				*X*		*X*		*X*
Vs. Racing 2			*X*		*X*		*X*		*X*
Playsketch Racing	*X*			*X*			*X*		*X*

Table 2.2: Comparative Analysis of the Researched Games

In Table 2.2, it is presented a comparative analysis of the different video games in context to the categories explained above. The symbol X means that the game (row) possesses a certain attribute (column), for instance, Playsketch Racing possesses input via buttons. Inside the same category, one game can have multiple attributes, this means that the game has the option to choose between the attributes or those are implemented to be used simultaneously.

Through analysis of the Input category, it is shown that all except one game use buttons as a control scheme, this buttons can be only directional (left and right) but also used to accelerate or brake the player character, for instance, Doodle Kart as the option to both use buttons only for directional proposes as the option to also use buttons to accelerate or brake, in case automatic acceleration is turned off. We can observe that 3 of the games also have the Tilt option as an input choice in case of Head to Head racing, this is the main way of turning the car left and right, with buttons only being used to activate a turbo option and to brake the vehicle. Both Doodle Kart and Mini Turbo GP only have the tilt option activated if the player chooses it from the options menu. For the Others attribute, Vs. Racing 2 uses a joystick type input as its main control scheme as High octane and Mini Turbo GP have that option available in the options menu.

Automatic Acceleration in mobile racing games is very important as only two of the selected games do not implement it. Half of the ones that have the option to turn it off but all of them have automatic acceleration turned on as their main method to make the car move forward. The Camera attribute had the least discrepancy between results, only Micro Racing uses an angle of view that shows the entire world that the player is interacting with besides focusing the action on the player character.

Relatively to the Game Modes category, this analysis shows that some games implement different modes to playing the game while others don't. The ones that do not implement different game modes focus in the racing against computer artificial intelligence controlled players in a simulation of real life racing competitions or, in case of Micro Racing, time challenges. The addition of different game modes brings curiosity and replayability to the video games, making them more entertaining in the long run.

[3] This mode invites the player to achieve the goal as fast as possible, usually with a competitive intention

Chapter 3

Approach

In this chapter, it is both presented the development method used in the internship's context and also describes the plan defined for the course, detailing tasks to perform and risks detected along its duration. The technology used for the internship was Unity3D, technology selected by the company.

3.1 Methodology

For the development of this internship project the methodology chosen was the Iterative Design methodology, in this case, oriented to the video game development. This methodology is based on a cyclic process of prototyping, testing, analysing and refining a work in progress. *"Iterative design is a play-based design process. Emphasising playtesting and prototyping, iterative design is a method in which design decisions are made based on the experience of playing a game while it is in development."*(Salen et al. [2004]).

As seen in Figure 3.1, the process begins with the conceptualisation of an idea, in this specific case, a video game idea (Concept). After the idea is conceived, it needs to be formalised into game rules and mechanics (Design). The next step is to develop a simple prototype that depicts the essential components of the video game (Prototype). Immediately after the prototype is finished, it needs to be tested (Playtest). The results of the test are then evaluated (Evaluate) and the generated knowledge should serve either as a basis to the design of improvements to the product or as the confirmation that the game is finished (Game). So, this process is present on all the steps of the video game development lifecycle until it is considered final.

3.2 Work Plan

3.2.1 Proposed Activities and Milestones

The work plan for this internship was divided into 7 activities, to be developed over the course of two semesters. These activities are then subdivided into tasks to facilitate the process of estimating the time necessary to complete. In the next paragraphs, the activities and tasks are described and identified with **Ax** representing the **A**ctivity number **x** and **Tx.y** representing the **T**ask number **y** associated with the activity **x**. Milestones are identified with **Mx.y** representing the Milestone number **y** associated with the activity **x**.

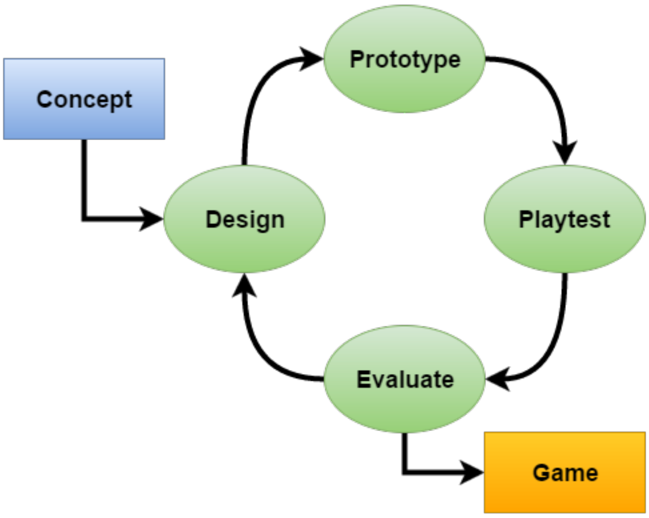


Figure 3.1: Iterative Process for Game Design (Mihealsick [2016])

In Figure 3.2, a Gantt chart with the work plan for the entire internship is presented, listing the activities to complete and their estimated duration.

During the first semester, the work will be based in the research and review of the background and related work related to the challenge proposed in the internship as well as to the current version of Playsketch Racing. With this in mind, the following activities and subsequent tasks are to be done by 21 of January 2019.

- **A1 - State of the art Review -** during the first two-thirds (2/3) of the semester, the intern will review the state of the art about the following subjects:
  - **T1.1 - Game Design Overview:** research about the game design topic and the making of video games
  - **T1.2 - Experience Models:** research about games user research, user and player experience and their respective characterising models
  - **T1.3 - Experience Evaluation:** research about experience evaluation methods
  - **T1.4 - Testing with Children:** research about conducting user testing sessions with children
- **A2 - Work Organization and Definition:** this activity comprises the definition of the work plan, regarding time estimates and work methodologies.
  - **T2.1 - Work Plan Definition:** this task consists in the analysis of potential risks to occur during the internship and the time estimation for the activities to complete and creation of the subsequent Gantt chart
  - **T2.2 - Research and Development Methodologies:** this task consists in the research and selection of methodologies to be used during the internship
- **A3 - Player Experience Evaluation:** in order to better understand the current state of the video game, a user testing session will be conducted during the first semester. This activity is associated with the "Playtest" and "Evaluate" steps of the followed methodology.
  - **T3.1 - User Tests Definition:** definition of the user testing environment and production of the test plan

  - **T3.2 - Prototype Development:** development of prototypes to be used by the testers
  - **T3.3 - Testers Gathering:** contact with testers and delivery of necessary participation forms
  - **T3.4 - User Testing Sessions:** evaluation sessions where the product will be in contact with potential end-users within the target audience

- **A4 - Proposition of Improvements:** this activity (dependent from A3) consists in the analysis of the data retrieved from the session conducted in A3 and subsequent proposition of improvements for the product. This activity is associated with the "Design" and "Prototype" steps of the followed methodology.
  - **T4.1 - User Testing Data Analysis:** Analysis of the data retrieved from the previous user testing sessions. The data will then be processed into visualization models, for instance, graphs and tables
  - **T4.2 - Conclusions and Propositions:** Discussion of the results from the previous analysis and elaboration of solution propositions to the flaws detected

During the second semester, there is a focus on the iterative improvement and evaluation of future versions of Playsketch Racing developed by the intern. After this is concluded, the intern is to develop new video game ideas to become part of the Playsketch collection of video games. The following activities are to be concluded until 1 of July 2019.

- **A5 - Implementation of Improvements:** this activity (dependent from A4) is subdivided into two tasks that will be performed on an iterative faction, after one task is finalised, the other is performed. As there is no way to know at this point how many iterations there are going to be performed until the criteria are met, these two tasks are shown in the Figure as to be performed simultaneously.
  - **T5.1 - Implementation:** this first task is implementing into the video game the solutions proposed in A4, after this first iteration, other iterations may be performed using the results of the next task T5.2. This task is associated with the "Design" and "Prototype" steps of the followed methodology
  - **T5.2 - Evaluation:** this task comprises the evaluation of the video game after the improvements are implemented in the previous task, resulting in the development of new propositions to the product, taking the internship back to task T5.1. When the evaluation criteria is considered met, this activity group is concluded and the next one starts. This this is associated with the "Playtest" and "Evaluate" steps of the followed methodology.
- **A6 - New Products Development:** this activity consists in the design, prototype development and evaluation of new video games to the Playsketch collection. This activity is associated with every step of the followed methodology as consists in the creation of completely new games.
  - **T6.1 - Game Concept:** conceptualisation of two to three new video game ideas
  - **T6.2 - Prototypes Development:** development of rough prototypes for the video game ideas designed in the previous task
  - **T6.3 - Prototypes Evaluation:** evaluation of the new video game ideas using the prototypes developed in the previous task

There is one activity that is transverse to the entire duration of the course, and to be performed simultaneously with the activities described previously:

- **A7 - Writing of the Dissertation:** For the entire duration of the semester, the intern must work in the dissertation's writing and consequential presentations as it is the main output from the internship.
  - **T7.1 - First Semester Writing:** writing of the internship report
  - **T7.2 - First Semester Preparation:** preparation to the milestone M7.1
  - **T7.3 - Second Semester Writing:** same as task T7.1
  - **T7.4 - Second Semester Preparation:** same as task T7.2

  This activity has two milestones associated, each representing, respectively, the mid-term and final deliveries.

  - **M7.1 - First Presentation:** Mid-term presentation addressing the results of activities A1, A2, A3 and A4
  - **M7.2 - Final Presentation:** Final presentation addressing all the work performed over the entire course

3.2.2 Final Activities and Milestones

Figure 3.3 depicts the final Gantt chart of the activities and milestones developed during these two semesters. From the original plan, it is possible to notice that activity A5 - Implementation of Improvements took twice the time estimated. This occurred because of two reasons. First, it was noted in the first semester evaluation that this activity should be the focus of the internship while A6 - New Products Development should be reduced due to how quickly a prototype can be developed. The other reason comprises difficulties in gathering a significant number of playtesters within the target audience range. Communication with schools and associations involve much bureaucracy and only a few of those contacted responded in useful time. Because of this, a second evaluation in the second semester could not be realised.

As stated in task, T6.1 - Game Concept, two to three video game ideas were to be conceptualised but, due to the reasons stated above, only one was in fact created. Task T6.3 - Prototypes Evaluation was not executed due to the same reasons. There was also the inclusion of a new task, T7.5 - Paper Writing, that was proposed by both advisers and consisted in the writing of a scientific paper about the discoveries of the evaluation realised, to submit to a conference. This task was associated with a new milestone, M7.3 - Paper Submission.

3.2.3 Risk Analysis

In order to properly plan the work to be done during this internship, it is necessary to identify, characterise, analyse and evaluate risks. Risks are events that have a possibility of occurring and must be resolved, within a time frame, to prevent a negative impact to the project in hand.

With this in mind, risks are identified by the intern in an attempt to prevent them from happening and also to construct a plan to mitigate them efficiently if necessary.

Risks are then categorized using the following attributes:

- **Impact**
  - **Catastrophic:** if the risk comes to fruition the internship objectives will not be completed;
  - **Critical:** if the risk occurs the work plan might need to be greatly revised;
  - **Marginal:** the risk happening might slightly affect the work plan, but is easily corrected.
  - **Negligible:** in case the risk occurs, the work plan can continue without worrying about delays.
- **Probability**
  - **High:** Between 75% and 100% chance of risk happening;
  - **Medium:** Between 50% and 74% chance of risk happening;
  - **Low:** Between 25% and 49% chance of risk happening;
  - **Very Low:** Between 0% and 24% chance of risk happening.
- **Time Frame**
  - **Very Short:** The risk need to be resolved within 1 week;
  - **Short:** The risk needs to be resolved within 1 and 2 weeks;
  - **Medium:** The risk needs to be resolved within 2 weeks and 1 month;
  - **Long:** The risk needs to be resolved within 2 months.

Risk #1: We can not find enough testers within the target audience.

- **Description:** User testing is meant to be performed with several children with ages within the target audience range (6-12 years). This specific kind of testers can be difficult to gather in significant numbers;
- **Impact:** Critical;
- **Probability:** High;
- **Time frame:** Medium;
- **Mitigation plan:** The company has connections with elementary and middle school professors who have worked with them before. Despite that, testing with friends and family within the target audience is an option and easier to execute.

Risk #2: The testers parents do not authorise their kids to be filmed for data retrieving purposes.

- **Description:** One instrument used to retrieve data from the play testing sessions is filming the session. That means that the testers parents must give permission for that to happen, situation that some might be against;
- **Impact:** Marginal;
- **Probability:** Medium;
- **Time frame:** Very Short;

- **Mitigation plan:** Besides the video filming of the sessions, we can use screen capture software. Some available options were already used by the intern in order to facilitate the installation and usage of said applications. The recording of audio without image can still be used as a tool for prior analysis.

Risk #3: The instruments used in the evaluations might return biased data.

- **Description:** The interns' lack of experience with user testing and further evaluation might generate surveys and questionnaires with leading and loaded questions;
- **Impact:** Marginal;
- **Probability:** Medium;
- **Time frame:** Medium;
- **Mitigation plan:** Communicating and receiving feedback from people with experience in user testing, such as both the department and the company advisers is of major importance. Also, already validated questionnaires from the related work will be used as the basis for the questionnaires produced by the intern.

Risk #4: Inaccurate estimations.

- **Description:** The interns' lack of experience with user testing and further evaluation might cause the time estimates to be inaccurate;
- **Impact:** Marginal;
- **Probability:** High;
- **Time frame:** Short;
- **Mitigation plan:** As described in the mitigation plan for the risk #3, communicating and receiving feedback from people with experience in user testing, such as the advisers, can drastically improve the time estimates correctness.

Risk #5: Unfamiliarity with C# programming language.

- **Description:** Even though the intern has already worked with Unity before, the knowledge about the C# programming language is little;
- **Impact:** Marginal;
- **Probability:** High;
- **Time frame:** Short;
- **Mitigation plan:** Communicating with the development team and web research can give the intern the information needed to efficiently code using C#.

Risk #6: Poor development documentation.

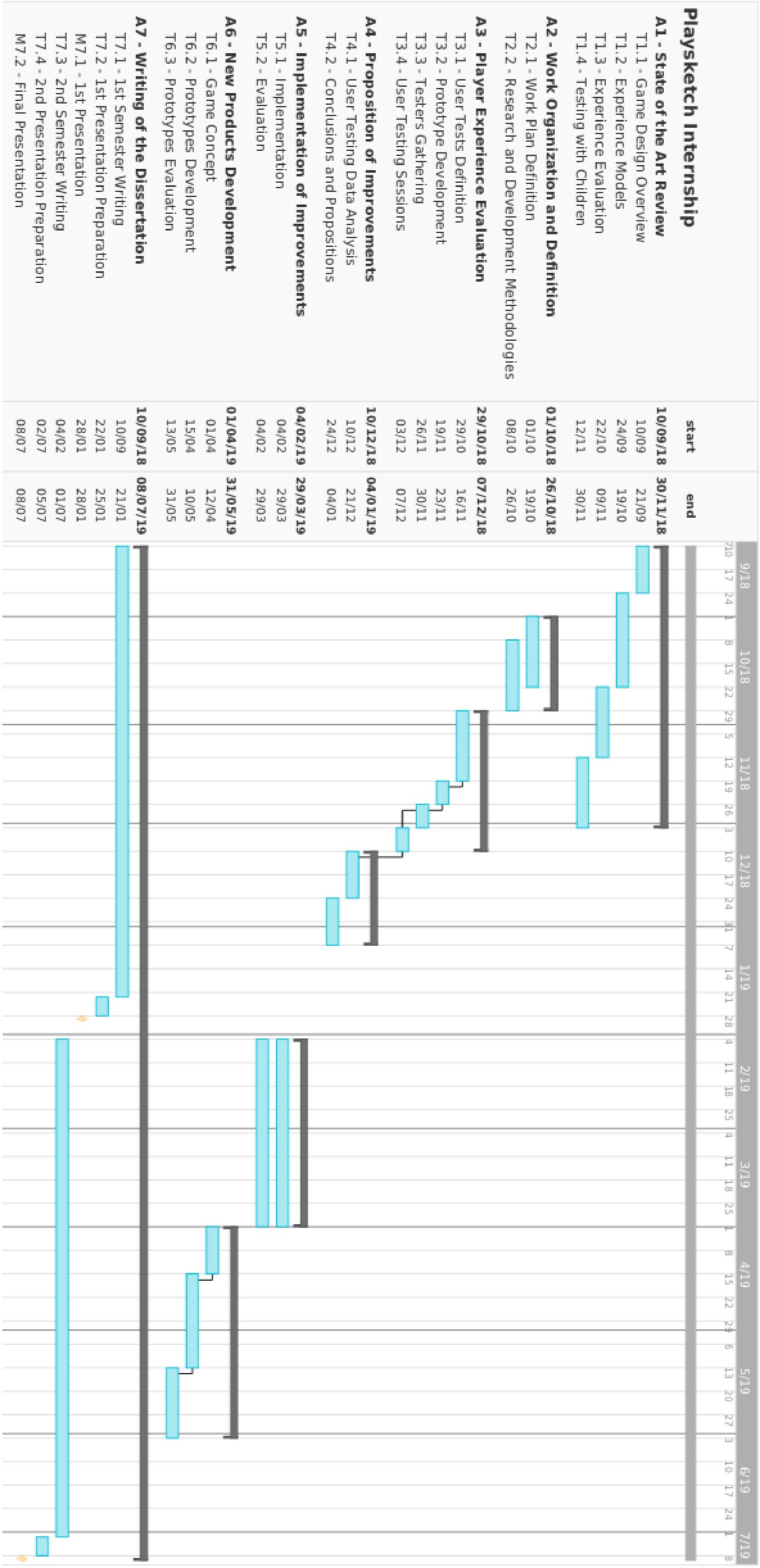
- **Description:** The entry of a new member to the development team of Playsketch Racing can be hard, as there is almost no documentation about the software;

- **Impact:** Critical;
- **Probability:** High;
- **Time frame:** Short;
- **Mitigation plan:** The intern must analyse the existing code during the first week of the first development iteration and communicate with the team any questions that might come up.

The identified risks are then put into a risk matrix, Table 3.1, to understand their scale in terms of the project. As only six risks were detected, a mitigation plan was elaborated to every single one instead of only the most impactful ones.

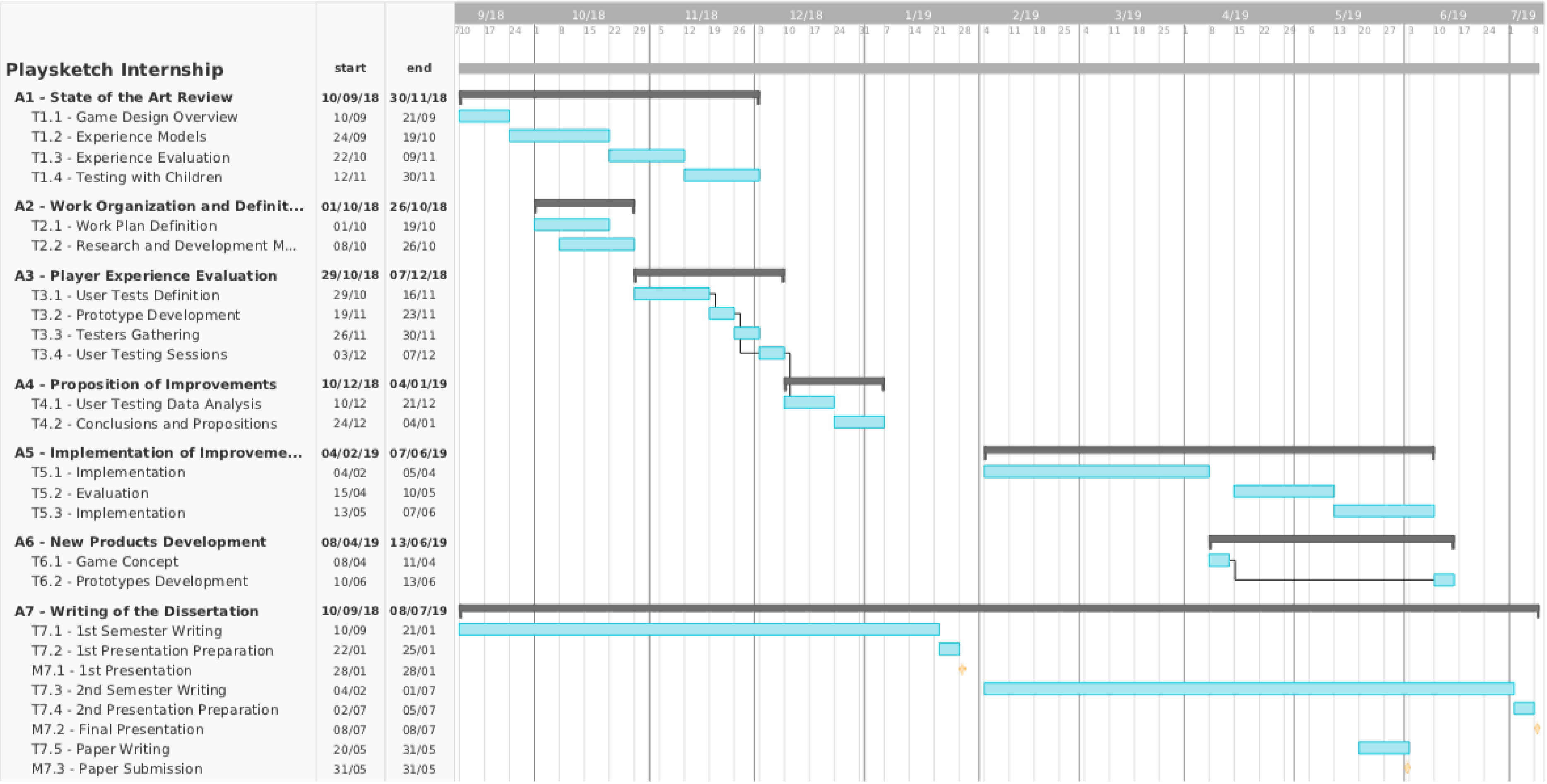
Probability	High	🟨	R#4, R#5	R#1, R#6	🟥
	Medium	🟩	R#2, R#3	🟥	🟥
	Low	🟩	🟩	🟨	🟥
	Very Low	🟩	🟩	🟩	🟨
		Negligible	Marginal	Critical	Catastrophic
		Impact			

Table 3.1: Risk Matrix



Playsketch Internship	start	end
A1 - State of the Art Review	**10/09/18**	**30/11/18**
T1.1 - Game Design Overview	10/09	21/09
T1.2 - Experience Models	24/09	19/10
T1.3 - Experience Evaluation	22/10	09/11
T1.4 - Testing with Children	12/11	30/11
A2 - Work Organization and Definition	**01/10/18**	**26/10/18**
T2.1 - Work Plan Definition	01/10	19/10
T2.2 - Research and Development Methodologies	08/10	26/10
A3 - Player Experience Evaluation	**29/10/18**	**07/12/18**
T3.1 - User Tests Definition	29/10	16/11
T3.2 - Prototype Development	19/11	23/11
T3.3 - Testers Gathering	26/11	30/11
T3.4 - User Testing Sessions	03/12	07/12
A4 - Proposition of Improvements	**10/12/18**	**04/01/19**
T4.1 - User Testing Data Analysis	10/12	21/12
T4.2 - Conclusions and Propositions	24/12	04/01
A5 - Implementation of Improvements	**04/02/19**	**29/03/19**
T5.1 - Implementation	04/02	29/03
T5.2 - Evaluation	04/02	29/03
A6 - New Products Development	**01/04/19**	**31/05/19**
T6.1 - Game Concept	01/04	12/04
T6.2 - Prototypes Development	15/04	10/05
T6.3 - Prototypes Evaluation	13/05	31/05
A7 - Writing of the Dissertation	**10/09/18**	**08/07/19**
T7.1 - 1st Semester Writing	10/09	21/01
T7.2 - 1st Presentation Preparation	22/01	25/01
M7.1 - 1st Presentation	28/01	28/01
T7.3 - 2nd Semester Writing	04/02	01/07
T7.4 - 2nd Presentation Preparation	02/07	05/07
M7.2 - Final Presentation	08/07	08/07

Figure 3.2: Internship Plan Gantt Chart



Playsketch Internship	start	end
A1 - State of the Art Review	**10/09/18**	**30/11/18**
T1.1 - Game Design Overview	10/09	21/09
T1.2 - Experience Models	24/09	19/10
T1.3 - Experience Evaluation	22/10	09/11
T1.4 - Testing with Children	12/11	30/11
A2 - Work Organization and Definit...	**01/10/18**	**26/10/18**
T2.1 - Work Plan Definition	01/10	19/10
T2.2 - Research and Development M...	08/10	26/10
A3 - Player Experience Evaluation	**29/10/18**	**07/12/18**
T3.1 - User Tests Definition	29/10	16/11
T3.2 - Prototype Development	19/11	23/11
T3.3 - Testers Gathering	26/11	30/11
T3.4 - User Testing Sessions	03/12	07/12
A4 - Proposition of Improvements	**10/12/18**	**04/01/19**
T4.1 - User Testing Data Analysis	10/12	21/12
T4.2 - Conclusions and Propositions	24/12	04/01
A5 - Implementation of Improveme...	**04/02/19**	**07/06/19**
T5.1 - Implementation	04/02	05/04
T5.2 - Evaluation	15/04	10/05
T5.3 - Implementation	13/05	07/06
A6 - New Products Development	**08/04/19**	**13/06/19**
T6.1 - Game Concept	08/04	11/04
T6.2 - Prototypes Development	10/06	13/06
A7 - Writing of the Dissertation	**10/09/18**	**08/07/19**
T7.1 - 1st Semester Writing	10/09	21/01
T7.2 - 1st Presentation Preparation	22/01	25/01
M7.1 - 1st Presentation	28/01	28/01
T7.3 - 2nd Semester Writing	04/02	01/07
T7.4 - 2nd Presentation Preparation	02/07	05/07
M7.2 - Final Presentation	08/07	08/07
T7.5 - Paper Writing	20/05	31/05
M7.3 - Paper Submission	31/05	31/05

Figure 3.3: Internship Final Gantt Chart

This page is intentionally left blank.

Chapter 4

Preliminary Evaluation

With the main objective to understand how potential players view and interact with the current version of Playsketch Racing, an opening playtesting session was conducted during the first semester. At this point, the intern had no contact with the game in terms of implementation. The target of this evaluation was the game as it was brought and presented to the intern. In this chapter, it is described the processes that lead to the realisation of this session, how the session was conducted and the results found in this session. Following Schell [2014], every playtest should be defined by five key questions: Why, Who, Where, What and How. The answer to the first question is shown in the first section of the present chapter, Evaluation Objectives. The remaining questions are answered in the Playtest Setup section.

4.1 Evaluation Objectives

The primary objectives for this playtesting session involve the players understanding of the provided control scheme, as it is one of the main problems with the playability of the current version of the video game, as referenced in section 2.3. Another objective is to perceive if the players understand the available hazards and power-ups (oil spills and acceleration tracks) and how do they interact with them. Finally, we want to apprehend what the players think about the "main character", the car, as it is the main means of expression with the video game. We want to know if the players feel that controlling the car brings them a sensation of speed or not.

Another points of study are the social aspect of the game, i.e., understand if the players prefer to play the game alone or with others and the desire to play the game multiple times.

These objectives can be expressed in the form of questions to answer them objectively upon analysis of the results:

- **Q1 -** Do players understand how to play?
- **Q2 -** Is the control scheme hard to use?
- **Q3 -** Do players like the interactions with oil spills and acceleration tracks?
- **Q4 -** Do players think the car is moving too slow, too fast or just right?
- **Q5 -** Do players prefer to play the game alone or with others?

- **Q6 -** Are players ever bored?
- **Q7 -** Are players ever confused by the HUD?
- **Q8 -** Do players want to play the game multiple times?

4.2 Playtest Setup

In order to answer the "Who" question, and by analysing the previous questions, the people to try and play the game are children aged 8-13 years that have never seen the game before. This age range was picked so that the participants fall into the upper part target audience age range, to accommodate the interns inexperience with testing with small children. Including two 13-year-old participants also serves to understand if people slightly above the target audience range do also enjoy the game. These participants are also acquaintances of the intern, making the entire session more comfortable and honest both for the intern as for the participants. Future playtests will include participants from the entire target audience age range and unfamiliar with the intern.

A consent form, written in portuguese and presented in Appendix A, was delivered to the participants two weeks in advance so they could think about their voluntary participation in the session and also to gain the consent of their respective tutors for the intern to video record the playtests. Only one of the contacted children did not want to engage in the session.

The participant group was then consisted by 12 participants both male and female and between the ages of eight and thirteen years old.

The session took place at the *Filarmónica Pampilhosense* music school facilities, institution that all participants are part of, making the classroom an educational but also recreational and creative environment, resulting in a place "Where" participants can be comfortable and fully engage in the action of play.

The tests were conducted with multiple participants from the same class simultaneously, following Als et al. [2004] conclusion that acquainted interaction is a better technique of collecting data versus the think-aloud method (Van Someren et al. [1994]). This technique invites the participants to not only talk to themselves while they play but also with others. Despite that, help between participants was not permitted as it would mask potential usability problems that the game might have.

Besides the necessary identification form to characterise the participants, three instruments to collect data necessary for the evaluation of the game were utilised: video recording, surveys and interviews. In Figure 4.1 it is presented the layout of the classroom while in the context of the playtest. A camera records the session while two observers watch and take notes. Both observers had different jobs, the intern, responsible for the session, observed the participants and took notes as an evaluator. The second observer was the teacher in charge of the classroom who helped with the explanation of the questionnaires when participants were doubtful.

Three sessions were then conducted, Table 4.1 depicts the participants age and gender distribution. The same data collecting instruments were utilised despite the fact of the age difference, giving special assistance to the younger participants.

Every session began with the filling of the characterisation forms, checking the participants gender, age, tablet and smartphone proficiencies, preferred types of video games and

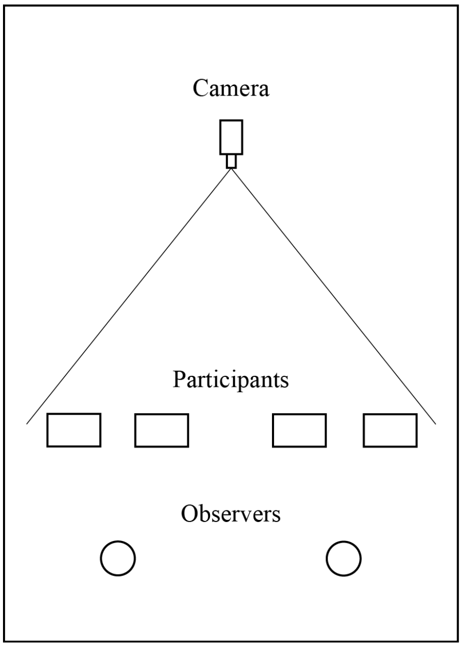


Figure 4.1: Playtest layout

Age Range	Number of participants	Gender distribution
[8-9]	4	2 Male & 2 Female
[10-11]	4	2 Male & 2 Female
[12-13]	4	2 Male & 2 Female

Table 4.1: Participants age and gender distribution

experience with racing games. After this first step, the playing of the game took place.

As this playtest has the main objective to verify problems associated only with the playing part of the game, a map was designed by the intern so that every participant has to race in the same map and not design to their own, evading problems with the creation of maps.

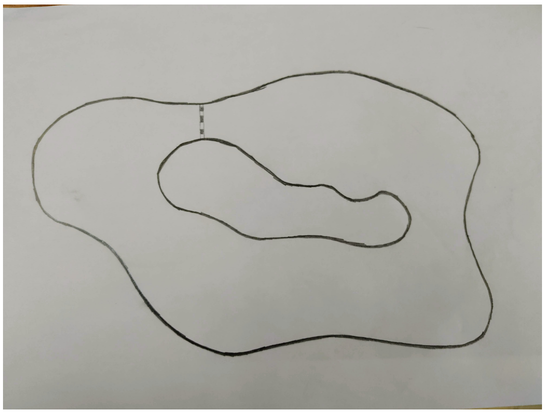


Figure 4.2: Map used in the playtests

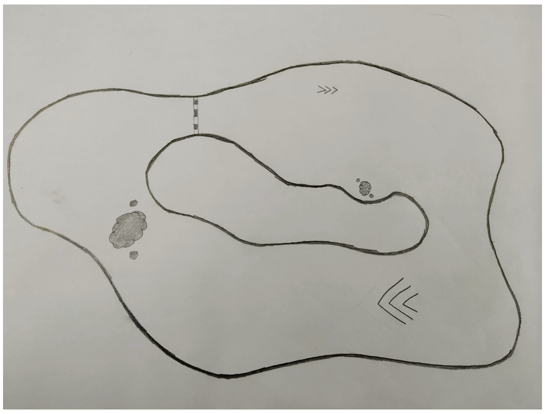


Figure 4.3: Map with game hazards included

The designed map, shown in Figure 4.2, is the only map to be used in the test. This map was played by all the participants three times. The first time, the game is played with only the map to understand how long participants might take until they understand the controls of the game. Next, it was incorporated into the map four new elements, the addition of two oil spills and two acceleration tracks. Two of each element had two different positions, one that the player could not avoid and one that was out of the way of the player, to check the curiosity in the interaction with these elements, as shown in Figure 4.3. Finally, this last map was used for the last activity, a MP game between 2 participants.

After the play sessions finished, the participants filled a survey related with their experience playing the game. The survey had a three point Likert scale as both Griffiths [2014] and Read and MacFarlane [2006] state that children tend to choose the extreme options of the typical five point scale, either they liked it, hated it or are indifferent about it. Read and MacFarlane [2006] Fun Toolkit's Smiley-o-meter was used in the context of this survey. Some questions were based on the GEQ developed by IJsselsteijn et al. [2013] as a way to try to understand the more subjective aspects of the video game experience. To answer the survey questions, the participants chose one of the following responses: "Yes", "Kind of" and "No", accompanied by "smiley faces" describing the response. The survey contained the following questions, originally in Portuguese:

1. It was fun (adapted from question 4 from the core module of the GEQ)
2. I felt bored while playing (adapted from question 16 from the core module of the GEQ)
3. It was easy making the car turn
4. The car moved too slowly
5. The game was too hard
6. I feel I played well
7. I was fast to understand the game objectives (adapted from question 21 from the core module of the GEQ)
8. I want to play again

With the first two questions, we try to understand about the "fun" quality of the game: do players enjoy the experience as is? Are they engaged in a playful experience or participating in this experiment and trying our game was only a chore? The third question serves to provide us how do players feel while interacting with their main character: does this aspect brings unwanted challenge to our game? The fourth question is related with "game feel" Swink [2008]: is the movement of the car well associated with the players expectations? Does the movement of the car take away the challenge intended?. Fifth and sixth questions are connected and serve for us to understand if it is necessary to implement more onboarding mechanisms to ensure the player about its progress. The last question functions as a market-testing tool: will these participants acquire the game when it is launched?

Finally, the participants were then engaged individually by the evaluator to answer some open-ended questions, using auxiliary graphics when needed. The reasons for these questions to appear in this playtest setting are presented in format of questions to be answered by the evaluator.

1. **What do you think about the car? (picture of the car) -** Do players dislike the game's "main character" and that affects the experience? Do players dislike it but they can enjoy the game the same? Do players like its simplicity or want a more "dynamic" character?

2. **Do you like the way how you drive the car? Would you like it to be different? -** Do players feel that controlling the car is intuitive? Does the movement of the car match with the players actions?

3. **What did these symbols do? (picture of the oil spill and acceleration track) Would you like the game to have more? Which ones? -** Do players perceive these elements as obstacles or as fun-making tools to a better experience?

4. **Did you enjoy more playing alone or with others? Why? -** Is the local split-screen/multiplayer aspect an important addition to the game?

5. **What were the things you like better? -** What are the things that are well designed/implemented to provide a good player experience?

6. **What were the things you did not like? -** What are the things that must be corrected or discarded?

7. **What would you like the game to have more? -** Do players care enough about the product so to provide thoughtful advices to make it better?

4.3 Results and Discussion

After all the data from the used instruments is collected, it is analysed and results from that analysis is presented in this section. First, the results taken from the on-site observations and video recordings are described. Next, the questionnaire results and finally, the interviews. Conclusions will be derived from these results and corrections will be proposed.

Observations

From the on-site observations and video recordings we can understand a variety of different results, both expected and unexpected. These are:

- Participants from the younger range of ages (8 and 9 years old) have problems until domain of the control schemes. A recurrent problem seen in 3 of the total participants is the tilting of the device to try to make the car turn to the pretended direction. This becomes even a bigger problem when playing in multiplayer mode, making the device awkward to hold for both players and the device orientation tends to flip, making the player in second place becoming the player in first place. Random pressing of the directional buttons can be observed in 2 of the participants. One participant tried to make the car turn horizontally to the left by continuously pressing the left button, making the car looping. This participant showed frustration and discontent with the game by shouting "It just keeps spinning"[1], without understanding that he was continuously pressing the same directional button. The same participant showed lack of immersion, worrying more about the environment and the surrounding people than the game. This characteristic was not apparent in only one other participant, implying that lack of immersion from the video game is not present.

- Three of the participants did not understand the laps count designation "lap 3/3", thinking that by achieving the number three, the race was already over, while, in fact, this means that they are in their final lap.

[1] Translated from portuguese "Isto está sempre a dar voltas"

- The first impression with the game elements of oil spills and acceleration tracks is correct in all the players, but, during gameplay, some players keep saying to each other that they don't do nothing.

- The final of the gameplay phase and the transition to the victory panel is too fast and players do not understand if the game ended or if they won. The position of the buttons in this panel match the position with the directional buttons during the gameplay, making the players press unwillingly the button to return to the main menu. This becomes a bigger problem in multiplayer mode, as the players involuntary click away the victory panel without knowing who won.

- The times shown in Figure 4.4 shows us that the time to complete the singleplayer maps decreases as the age of the participants increases, as lowest identification numbers correspond to younger participants. Both participants number 3 and 8 took over thrice the time to complete map two than map one because of the curiosity to understand what the oil spills and acceleration tracks did.

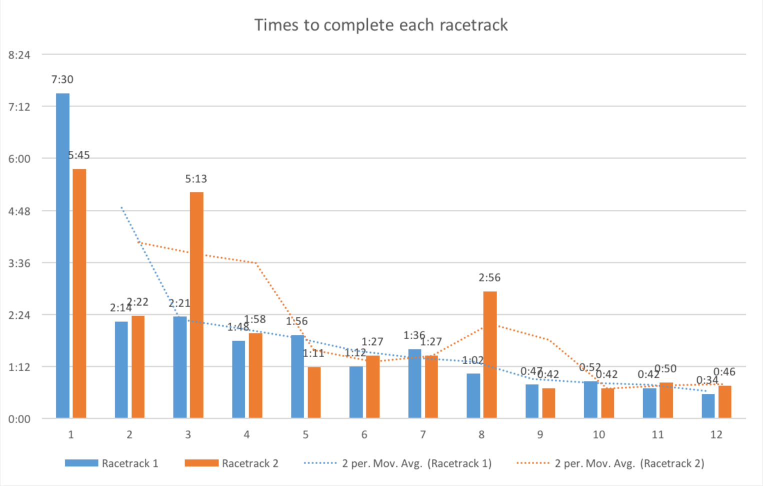


Figure 4.4: Times to complete the singleplayer races

Questionnaires

The questionnaire objective was to understand what the participants felt about their experience, specifically about the player movement and the affects that the game has on them. Answers from the survey are scored using the values 1, 2 and 3 depicting a negative, null and positive impact from the video game. Then, all the scores are added to achieve a number that represents the overall satisfaction with the game, regarding the questionnaire objectives.

In Figure 4.5, the results from the questionnaire are shown where each bar corresponds the satisfaction of the video game related with a specific question. In Table 4.2, the mean, median and mode of the results to each question is presented.

Two of the questions got a perfect score, questions number 1 and 2, related with the fun aspect of the game while question number 3, related with the controls, was the lowest scored question. The frequency of answers to this question can be seen in FIgure 4.6. This reinforces the necessity to find alternatives for the current control scheme present in the game. More experienced racing players thought the car moved too slowly, according to the

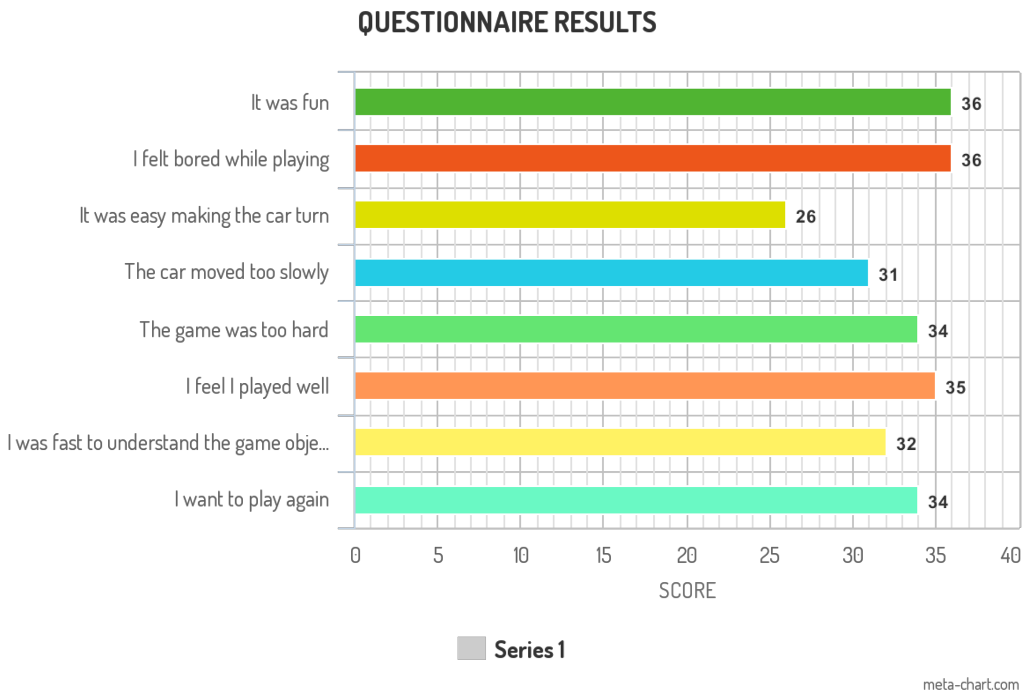


Figure 4.5: Questionnaire results by question

Question	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Mean	3	3	2,167	2,583	2,833	2,917	2,667	2,833
Median	3	3	2	3	3	3	3	3
Mode	3	3	3	3	3	3	3	3

Table 4.2: Mean, median and mode for questionnaire questions

question 4 results. Question 8, related to the replayability of the game, scored almost a perfect score, with only one participant stating that they would not like to play the game again. It is possible to verify that the remaining questions have a positive score.

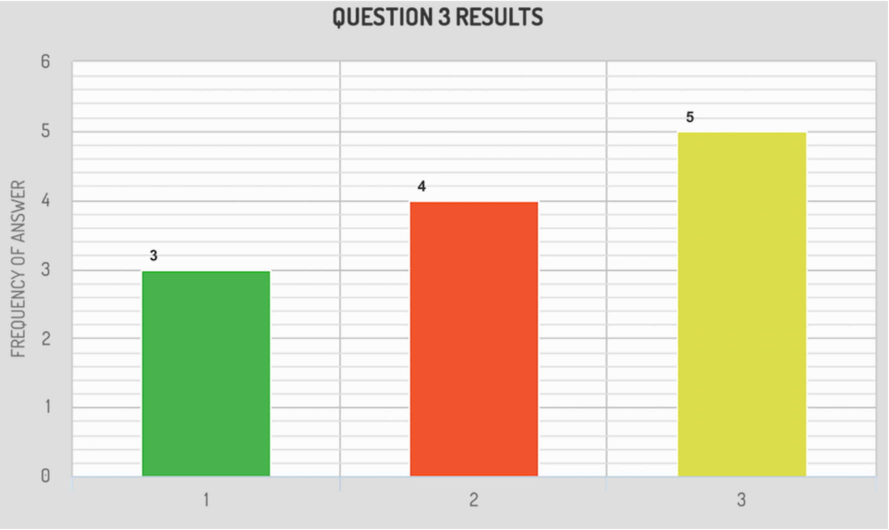


Figure 4.6: Frequency of answers to "It as easy making the car turn"

Interviews

Finally, the interviews conducted with the participants had the goal to understand more deeply what the players think about the game elements and suggestions they might have to make the game better.

For the first question, there were mixed opinions by the participants, while the more experienced players said that the car as slow, small and uninteresting, the least experienced ones stated that it was fast and appealing, even though they would like to see a more detailed vehicle.

The second question brings us a contradictory statement from question 3 from the survey, while in the survey some people stated that they thought the controls were hard, in the interview all but one participant stated that they enjoyed the way how they control the car. This can be explained by the honesty that the questionnaire brings that the interview does not, the participants might feel the necessity to tell the evaluator they enjoyed it, even though the anonymous survey shows otherwise.

The third question came up with some unexpected results by the intern, 6 participants stated either that the oil spills surely did nothing or they could not answer, even though they could answer that the element should make the car slip. The vast majority of the answers shows that the acceleration track is working as intended. There were some suggestions for new game elements such as pikes, roadblocks and portals.

The fourth question showed a fifty/fifty distribution between preferring to play alone or play with another person. This distribution cannot be answered by either age, gender or experience with racing games. The reason stated in favour of singleplayer is the discomfort to play multiplayer games using the same device while the reason stated in favour of multiplayer is that the game becomes more competitive aspect and therefore “more fun”.

Question number five showed that the things that the participants most enjoyed about the game were the way the car drifts while doing a turn and the acceleration track. Question number six showed that participants least enjoyed the hazards supposedly being useless and the race track being too simple.

The final question was open-ended and participants encouraged to answer whatever came to their minds. The answers are the addition of new cars, power-ups, camera angles, maps and obstacles, an option to upgrade the car using in-game coins earned by playing the game and the implementation of a multiplayer online system to play with friends over the internet.

4.4 Conclusions

After the analysis of the data is concluded, there are some problems detected with both the game and the process of evaluation conducted by the intern.

The problems detected with the game were the following:

1. Misunderstanding of the controls and further difficulty in usage by the younger participants. This problem might become even more relevant as the new evaluation sessions will include testers with younger ages and it is going the be considered the main problem with the game.
2. Misunderstanding of the oil spill element.

3. Misunderstanding of the lap count element.
4. Transition between gameplay panel and victory panel too abrupt with little feedback to the player.

In order to solve the above-stated problems, the intern proposes a series of corrections to be implemented and further evaluated during the second semester:

1. Development of prototypes with different control schemes based on the alternative control schemes found in the related work stated in section 2.4. These include tilt option with the device accelerometer, joystick option and both directional buttons and manual acceleration and brake options.
2. Implementation of a single full 360 degree uncontrollable spin when interacting with the oil spill element.
3. Implementation of a new lap count element based in painted balls, as shown in Figure 4.7.
4. Implementation of a transition panel with the car coming to a stop between the gameplay panel and victory panel.
5. Device orientation lock to avoid orientation problems.

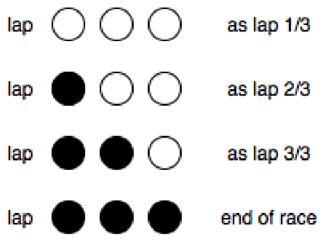


Figure 4.7: Prototype for the new lap count design

The problems detected with the process of evaluation and proposed corrections were:

1. Tests with multiple participants can become chaotic and is hard to grab at 100% the attention of the participants. Avoid these tests and opt to use smaller groups or the think-aloud method.
2. The usage of a 3 point Likert scale returns little information about the degree of satisfaction with certain elements. Future questionnaires will use a 5 point Likert scale.
3. Screen recording was not utilised in this evaluation and some information might be missing because of that. Future sessions will include screen recording as an instrument of observation and analysis.

This page is intentionally left blank.

Chapter 5

Architecture and Implementation

5.1 Unity Architectural Pattern

Unity, as a game engine, follows an architectural pattern that is mostly used in game development, the Entity-Component-System (ECS) pattern. This pattern follows the composition over inheritance principle of Object-Oriented Programming. This facilitates code reuse and polymorphic behaviour across multiple classes and allows greater flexibility in defining objects in the game.

An entity is a general object with no data attached to it besides its identifier. Entities are then populated with components. A component is data that defines objects in relation to a game world or with a system. Systems are jobs that run continuously and perform actions over components. In Unity3d, an entity that exists in the context of a game scene and contains components that interact with global systems is called GameObjects. For example, a GameObject representing a car has a controller component that interacts with the Input system. The Input system reads players inputs and send information to the component to make the necessary changes to the car object. One entity is often populated with many components that have relationships between each other. In the previous example, after a players input, the Input component would send the necessary information to the Transform component, which contains the GameObject's position, rotation and scale data, to make the car move in the game world.

Unity possesses a class, MonoBehaviour, that functions as the base class for every Unity script. This class contains all the core Unity methods linked with its systems. For instance, MonoBehaviour provides methods such as Start(), Update() and more. Start() is a method only called once when a script is enabled and is normally used to initialise objects. Update() is called every frame and is used to implement any kind of game script that must execute every frame, such as the calculation of the next position for an animated dog. FixedUpdate() is similar to Update() but is linked with the physics system, this method does not execute every frame but instead at the frequency of the physics system.

5.2 Controllers

In the previous version of the game, the player controlled the car using two directional buttons, as seen in Figure 5.2. One button steered the car left and another steered the car right. The acceleration of the car was not controlled by the player and was fully

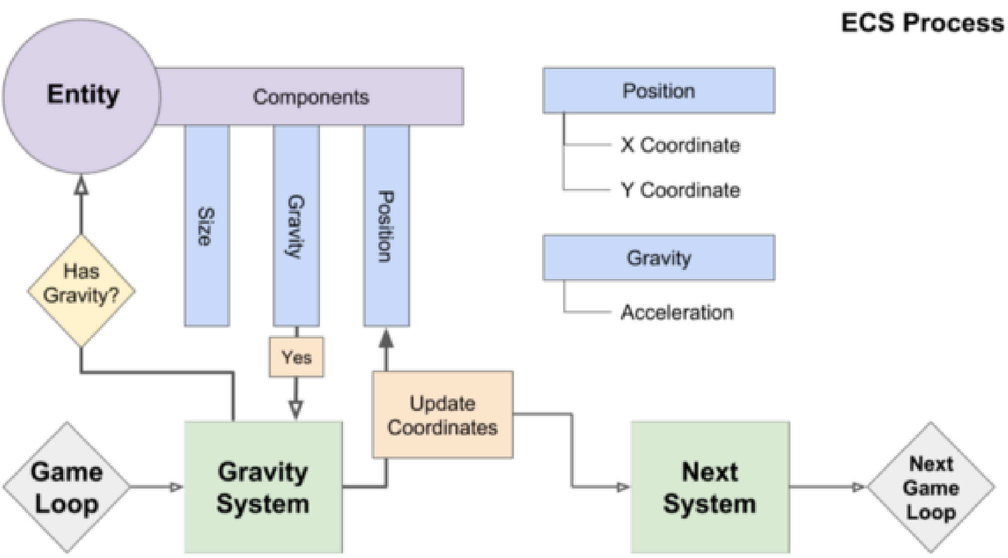


Figure 5.1: Example for a GameObject with components affected by a gravity system (Clevyr [2018])

automatic. As seen in section 4.4, this represented a problem as many players felt they could not control the car as they desired.

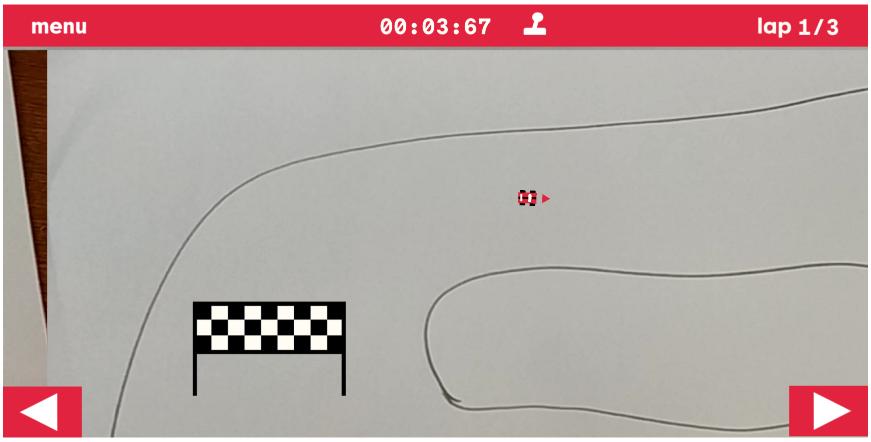


Figure 5.2: Previous controller, using buttons to steer

Five new control schemes were selected from the related work research on similar mobile games, described in section 2.4. These were:

- Analog Joystick: manual acceleration controller with a thumbstick used to both to steer the car and to control its speed;
- Joystick with Acceleration Button: manual acceleration using a button and a thumbstick to steer the car;
- Steering Joystick: automatic acceleration where dragging a thumbstick left and right make the car turn left and right, respectively. This was later discarded during implementation as it did not take full advantage of the 360 degree capabilities of the joystick and was not comfortable to use it only to steer the car;

- Steering Wheel: automatic acceleration where rotating the wheel clockwise turns the car right while counter-clockwise turns it left;
- Tilt: automatic acceleration based control where turning the device makes the car turn;

5.2.1 Input Management

In the previous version of the game, there was a component attached to the cars' GameObject, a C# script DriftingCarMovement, that controlled both the management of players input as well as the movement of the car. This was a problem as, to implement new input options, it would be necessary to modify the car movement logic as well. This also meant that the code used to manage player input could not be re-utilised in new controllable objects or even new games.

So, it was necessary to divide this previous component into two separate ones:

- DriftingCarMovement: previous C# script without the input management portion. This component is responsible by applying the forces to move and turn the car, following the Physics Unity system.
- InputManager: new component responsible for reading input values and sending them to DriftingCarMovement component. This component functions as an abstract class that is after used to create the desired input component such as AccelerometerInput, AnalogJoystickInput, ButtonJoystickInput, SteeringWheelInput and DirectionalButtonsInput.

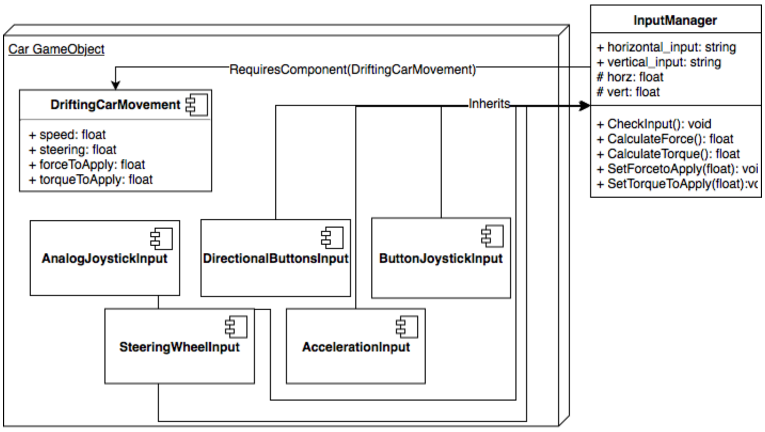


Figure 5.3: The car GameObject contains a component for the movement and different components for the controllers

5.2.2 New Control Schemes

Unity possesses an Input system that uses virtual axis that, when associated with a GameObject, returns values between -1 and 1, with 0 being the neutral position. Then, in the InputManager component for the car GameObject and the input GameObject, we need to associate these virtual axis. "P1 Horizontal" and "P1 Vertical" were defined as horizontal and vertical axis, respectively. For instance, a joystick GameObject associated with"P1 Horizontal" and "P1 Vertical" is used to control our car. If this joystick is fully pressed to the left, Unity's Input system returns the value -1 to "P1 Horizontal" and 0 to "P1 Vertical". Each controller uses MonoBehaviour's Input.GetAxis(string) method to retrieve the values from each axis.

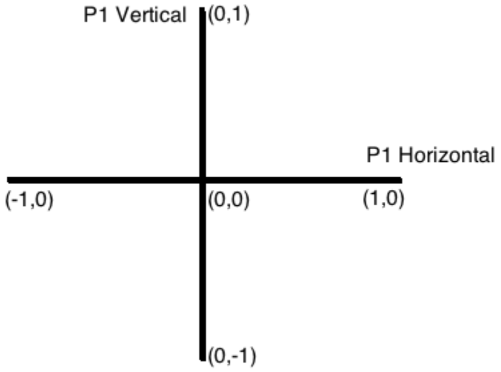


Figure 5.4: Unity input axis

These values are used in order to calculate the in-game physic forces to apply to the car. Two variables are used by the new components to represent these forces: force represents the linear force used to make the car accelerate and decelerate; torque represents the rotational force used to make the car turn.

Steering Wheel

For the steering wheel input, the only virtual axis used is the "P1 Horizontal". The values read in this axis, stored in the "horz" variable, are processed using a logarithmic function to calculate the torque to apply in the movement controller. This approach was selected after self-testing a linear function calculation. Following a linear function, it was difficult to make fine movements around obstacles.

```
public float CalculateTorque(){
float torque = 0;
if (horz > 0)
    torque = (float)-Math.Log(Math.Abs(horz) + 1, 2);
else if (horz < 0)
    torque = (float)Math.Log(Math.Abs(horz) + 1, 2);
return torque;
```

Tilt

Tilting the device, as shown in Figure 5.8, makes use of a special Unity axis called Input.acceleration. This axis reads the values from the device accelerometer according to the axis shown in FIgure 5.6. Only the "x" axis is of importance to our controller, as the game is played with the device in landscape mode.

A threshold was implemented to guarantee that small movements with the device do not interfere with the players actions. A second threshold was implemented in order to make the car fully turn to either direction if the player reaches a certain angle in the device. This

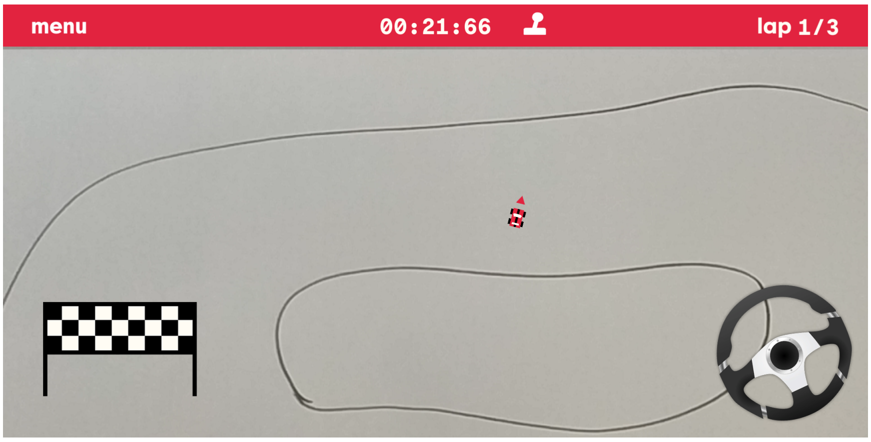


Figure 5.5: Steering wheel input

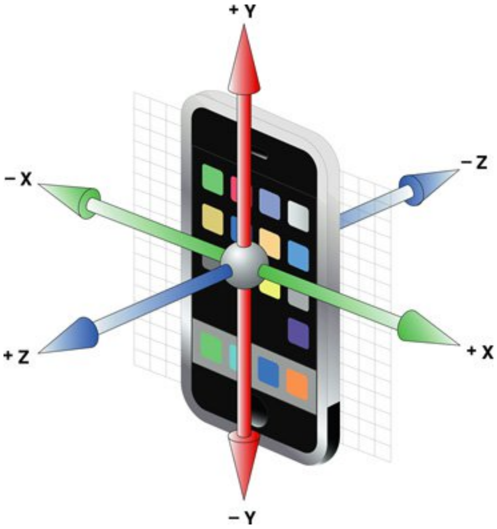


Figure 5.6: Acceleration Axis Ravado [2016]

way, the player does not have to make abrupt rotations to the device in order to rapidly steer the car. The values for the thresholds were result of a trial-and-error session, as most game parameters tend to be (Swink [2008]).

```
public float CalculateTorque(){
    float torque = 0;
    if (Math.Abs(Input.acceleration.x) >= maxThreshold){
        if (Input.acceleration.x > 0)
            torque = -1;
        else
            torque = 1;
    }
    else if (Math.Abs(Input.acceleration.x) > minThreshold){
        torque = -Input.acceleration.x * sensitivity;
    }

    return torque;
}
```

Two small sprites were added to the bottom of the screen to tell the players the importance of tilting the mobile device, as can be seen in Figure 5.7.

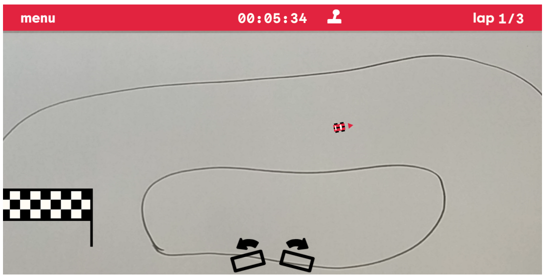


Figure 5.7: In-game indication of tilting controller

Figure 5.8: Tilting the phone makes the car turn

Joysticks

Three different joystick options were implemented: Analogue Joystick, Steering Joystick and Joystick+Button. Regarding the calculation of the linear force necessary to move the car, each option was different. Steering Joystick used automatic acceleration, with the force value being kept at 1 at all times; Joystick+Button used a button, if the button is pressed, the force value equals 1, if not, it equals 0. Analogue Joystick calculates the force by calculating the distance between the centre of the joystick and it's current position on the screen. The current position is received through Unity input system horizontal and vertical axis, represented with the *horz* and *vert* variables.

```
public float CalculateForce(){
        //get distance between center and current joystick position
        current = new Vector2(Mathf.Abs(horz), Mathf.Abs(vert));
        return Vector2.Distance(Vector2.zero, current);
    }
```

While Steering Joystick used the same code as the Steering Wheel for the calculation of torque, both Analogue Joystick and Joystick+Button do not recur to torque physics to rotate the car. Instead, as the car must follow the same direction that is pressed in the thumbstick, the angle that the thumbstick is pressed is used to rotate the car. This makes use of a method called *RotateTowards* that rotates a GameObject from its current to a target rotation over the course of several frames.

To assist with the joystick orientation, a small moving arrow rotates around the car to show the player his chosen direction in the game context, as seen in Figure 5.9.

5.3 Playability Features

Other features were proposed in the opening evaluation in order to make the playability of the game more enjoyable, depicted in Section 4.4. These were the 360 degree uncontrollable spin when colliding with oil spills and the transition between gameplay and victory scenes.

The uncontrollable spin was achieved by taking away player control over the car and applying a rotational force, torque, to the car in a random direction. Using 4 TrailRenderer

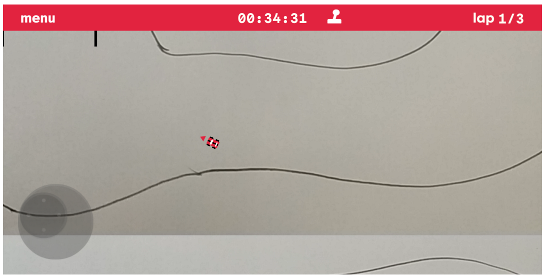


Figure 5.9: Analogue joystick input

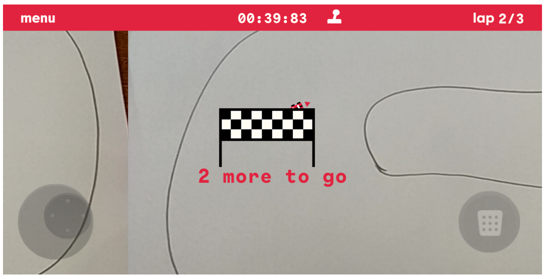


Figure 5.10: Joystick+Button input

components attached to 4 GameObjects that represent the car wheels, an oil trail follows the car while spinning, as shown in Figure 5.11. The game waits for 1.5 seconds before returning the control to the player and annuling the applied force.

The latter one was achieved by decelerating the car while forcing the game to wait for 3 seconds to load the victory scene, by using MonoBehaviour method WaitForSeconds(float), and by showing an in-game "finish" message in the centre of the screen.

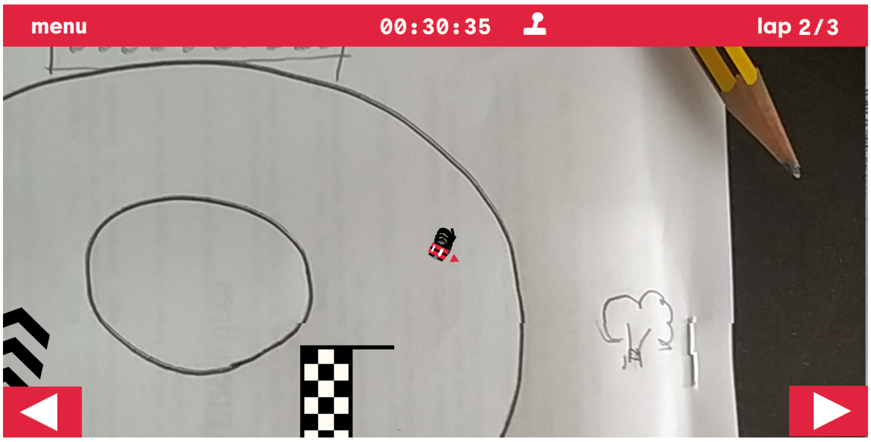


Figure 5.11: Oil trail when interacting with oil spills

Some other features were implemented while observing user interactions and self-testing:

- Final lap and race finished warnings: as described in section 4.4, some participants did not understand when the race finished, due to the representation of the number of laps left to complete the race. To better identify the players progress in the race, warnings with the number of laps left to complete the race were implemented as well as a warning to identify the finish of the race. These warnings are GameObjects with text components that pop in the centre of the screen for 3 seconds and then disappear;

- World limits: during user testing scenarios, depicted in sub section 2.3, when the racetrack drawing is not fully detected by the creation algorithm, players could move the car out of bounds. This represented a problem because players could not finish the race. The picture of the racetrack is associated with an in-game rectangular GameObject. The edges of this object are considered the world limits or the limits of the playable area. Colliders were placed in these limits so to prevent the car from exiting the playable area;

- Speed trail: during the opening playtests, we concluded that some players did not immediately understand that acceleration tracks provided more speed to the car.

A speed trail was implemented when interacting with the acceleration tracks. This speed trail is associated with two GameObjects, each with a TrailRenderer component attached to. When the player passes a speed trail, both of these GameObjects are enabled and spawn an orange trail, similar to fire, as shown in Figure 5.12. This feature, in conjunction with the next feature, provide a better understanding of the acceleration track element;

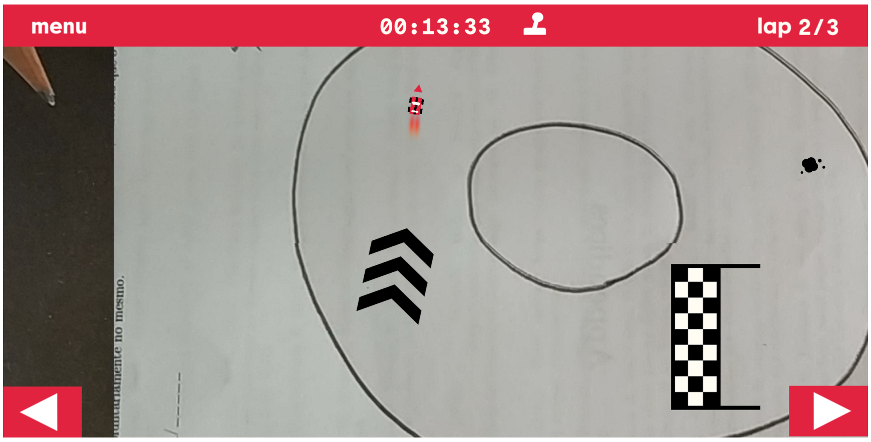


Figure 5.12: Speed trail when interacting with acceleration tracks

- Camera zooms out when car is speeding up: with the previous feature stated, this implementation zooms out the camera that follows the player. This gives a better sensation of speed and it is common practice in racing games;
- Respawn car at initial position when stuck: sometimes, players draw levels that are too narrow for the car to pass through, or create a "pocket" where the car can get stuck and players cannot revert to the racetrack. This feature was implemented using a MonoBehaviour method, OnCollisionStay2D(). This method is called by the physics system at every frame that the associated GameObject is colliding with another GameObject. A counter keeps track of the number of frames that the physics system calls this method for the car GameObject and is incremented if the car's position is equal to the position of the frame prior;

```
//This method is called at every frame that the GameObject is
//colliding with another GameObject
private void OnCollisionStay2D(Collision2D collision){
    VerifyIfStuck();
}

private void VerifyIfStuck(){
     //the difference between the current time and the time
     //when the car's position was last verified
     if ((Time.time − lastCheckTime) > checkStuckSeconds){
        //the distance between the current position and the
        //last checked position
        if ((transform.position − lastCheckPos) < checkStuck){
            //if the counter is bigger than a defined threshold,
            //respawn the car at finish line
            if (stuckCounter >= stuckThreshold)
                Respawn();
            else
```

```
                stuckCounter++; //increment the counter
        }
        else
            stuckCounter = 0; //reset counter

        lastCheckPos = transform.position;
        lastCheckTime = Time.time;
    }
}
```

- New game element to send car to the finish line: one restriction with the creation of a new level is that the racetrack must be a closed loop, so that three laps might be achievable. Some players do not understand this restriction and draw tracks that are not shaped in this way. To mitigate this problem, a new game element, a “portal”, was implemented. This new game element, when colliding with the car, sends it back to the finish line, incrementing the lap counter. In Figure 5.13 we can see how this new element can be used to work around closed racetracks as well as provide interesting concepts, in this case, a tunnel.

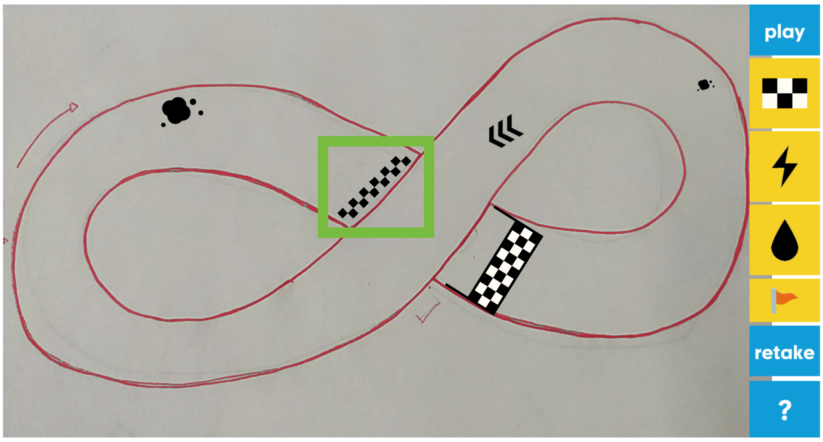


Figure 5.13: Portal element and an example of its usage

Another implemented feature was a set of screens which are shown to the user in case of the first interaction with the product. These screens have the aim of telling the initial information about the product to a new user. In this specific case, there are five screens. The first screen sets the first contact between the user and the game and acts as an introduction, starting with a greeting. The following screens are informative in nature and describe the various steps to fully engage with the game: draw a racetrack, take a picture, customise and play.

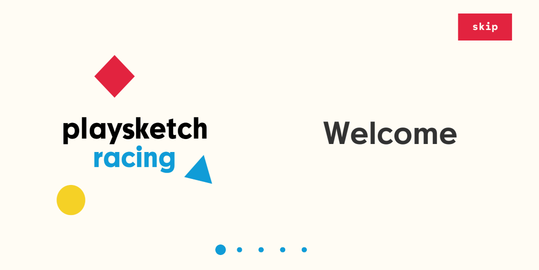
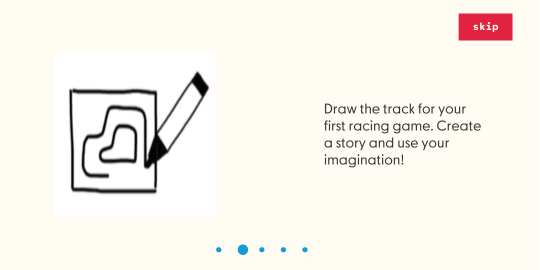


Figure 5.14: Tutorial greeting screen



Draw the track for your first racing game. Create a story and use your imagination!

Figure 5.15: Tutorial first information screen

A new game mode was also implemented, in order to ensure greater replayability to the game. This game mode follows Bushnell's Law: *"All the best games are easy to learn and difficult to master. They should reward the first quarter and the hundredth."*. It was named "Rage Mode" and consists in the completion of three laps, just like the previous "Race Mode" but every time the player collides with a limit or an obstacle of the track, it is sent back to the beginning. Even though the core gameplay is identical, this mode encourages the players to play the game more carefully instead of finishing the race as soon as possible.

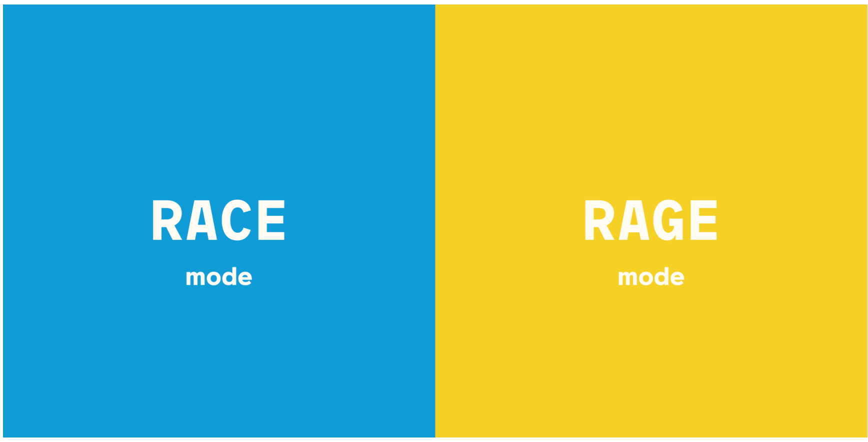


Figure 5.16: Game mode selection menu

Finally, and as creativity is at the core of this game, we developed a way to players play the game using multiple in-game "characters" instead of just playing with the car. Each character has different values for the components associated with the Unity physics system so playing with another character provide different experiences.

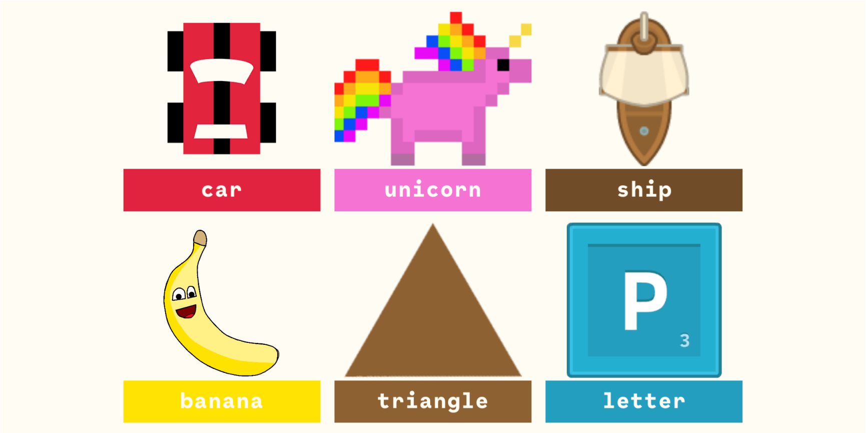


Figure 5.17: Character selection menu

5.4 Binarization algorithm for level creation

One problem detected with the creation phase of the game was related with the binarization threshold and how it impacted the transformation of the drawing picture into a playable level. The previous implementation was a simple comparison between the grayscale value

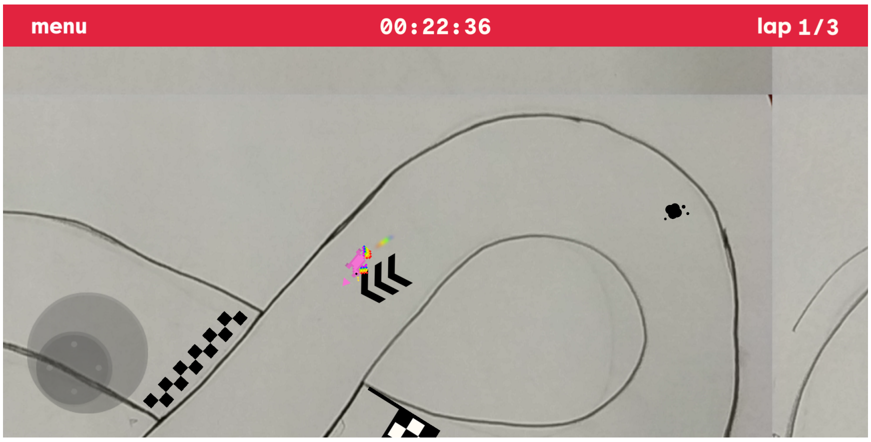


Figure 5.18: Playing with a unicorn instead of the car

of every pixel and a static global threshold, as shown in Algorithm 1.

```
Result: Photo with collidable pixels
foreach pixel in photo do
    if $pixel.grayscale < threshold$ then
        pixel.collidable = true;
    else
        pixel.collidable = false;
    end
end
```

Algorithm 1: Static Threshold Comparison

To tackle this problem, two different algorithms were implemented. The first one, Otsu [1979] method, is a global thresholding method that automatically performs histogram shape-based image thresholding and is considered the most successful global thresholding method (Chaki et al. [2014]). This algorithm assumes that the image contains two classes of pixels, such as a foreground and background, and then calculates the optimum threshold in order to separate them. A C# adaptation of Greensted [2010] was utilised.

The second algorithm makes use of a rectangular moving window through the original photo and applies the Sauvola and Pietikäinen [2000] formula, Equation 5.1, in order to calculate local thresholds to each window. Inside each window, the pixels grayscale values are then compared to the local threshold.

$$T(i,j) = m(i,j) * [1 + k(\frac{\sigma(i,j)}{R} - 1)] \tag{5.1}$$

Where

- i: width of the moving window;
- j: height of the moving window;
- T(i,j): threshold for the window;
- m(i,j): mean of the window grayscale values;
- k: constant between 0 and 1;
- σ(i,j): standard deviation of the window grayscale values;

- R: constant between 0 and 255;

Between the three options (static, Otsu and Sauvola), the Sauvola binarization algorithm was the one that provided better results, as depicted in Figures 5.19, 5.20 and 5.21. Both the static and the Otsu algorithms detect shadows as belonging to the original drawing and make its corresponding pixels collidable, while the Sauvola algorithm does not. This brings a better user experience as players do not need to be fully aware of the lighting available, while previously this had a great impact in the end result.

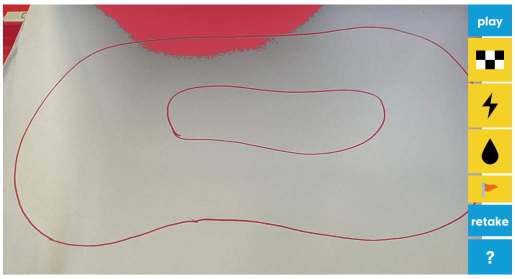


Figure 5.19: Static thresholding

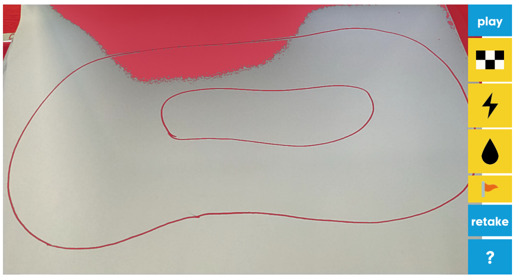


Figure 5.20: Otsu thresholding

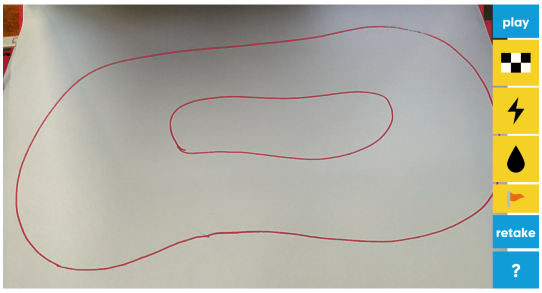


Figure 5.21: Sauvola thresholding

5.5 Multiplayer Server

One important characteristic of this game is the possibility to explore ones imagination and create innovative levels. Sharing them is the next step to provide a more compelling social aspect to the game. With this in mind, it was proposed to implement a web server dedicated to the upload and download of levels created in the game. This new feature was not planned in the start of the project but was deemed as a good stepping stone to future versions of the game to include an online multiplayer mode. As such, only a proof of concept prototype was developed for internal testing and deliberation.

There are two main types of online multiplayer gaming:

- Turn-based or Asynchronous: in turn-based multiplayer, two or more players normally play against each other and only after one player completes a task, either a success or failure, the other player makes his attempt to complete his task;
- Real-time Online: in real-time multiplayer gaming, two or more players are online at the same time and play the game simultaneously.

For this game, turn-based multiplayer was selected for its online aspect. One player creates a racetrack, does a race in it and uploads both the racetrack and the information about the race itself, for example, the time to complete the race. When playing a downloaded race, it is spawned a grey “ghost” car that represent the uploader position and rotation along the race, as shown in Figure 5.22.

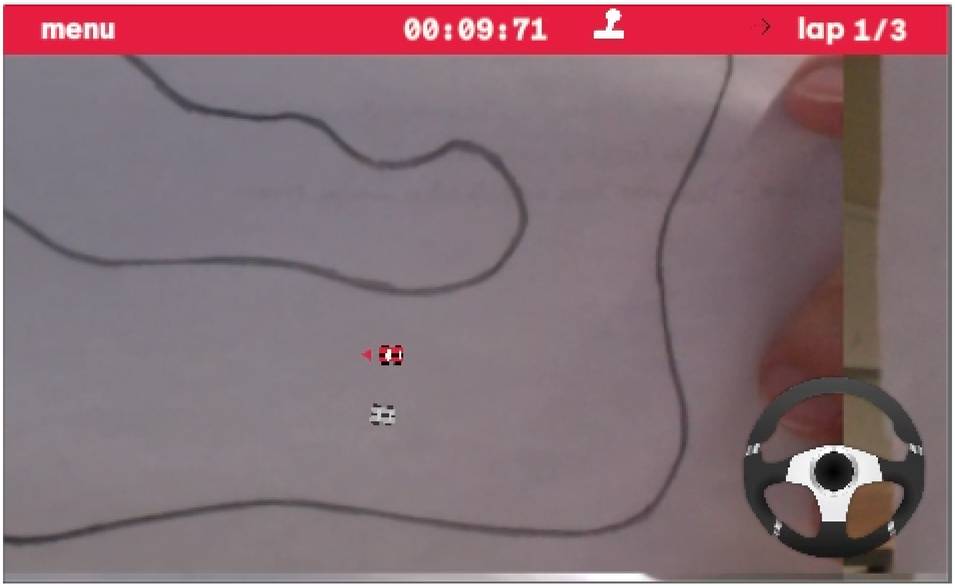


Figure 5.22: A gray "ghost" car represents the other player

5.5.1 Requirements

The prototyping of this server will be based in the following "Must Have" requirements, discussed with the company adviser:

1. Upload Race
    - User story: As a player, I want to share my races so that they become available to everyone to play against
    - Description: The game must have the capability for players to share their creations with their friends via a sharable link or identifier
2. Download Race
    - User story: As a player, I want to download other people races so that I can play them
    - Description: The game must have the capability for players to download other people creations and race in them

5.5.2 System Overview

The server was implemented in NodeJS, technology that the intern had no experience with. To help with its development, two middlewares were utilized, tackling different attributes of the server:

- Express: a minimal framework that provide a set of features for web applications. This was used as basis to the servers’ routing and response handlers.

- Multer: middleware for handling multipart/form-data web forms. This was used to permit the photo files to be sent in conjunction with the race information.

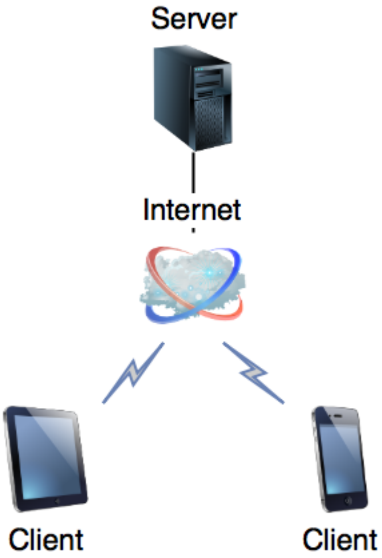
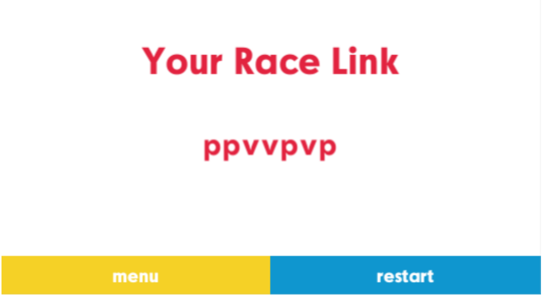


Figure 5.23: Connectivity between server and clients

This server follows a typical client-server architecture, communicating with game instances through the internet, Figure 5.23, handles two types of web requests:

- POST in route /api/create: using the game application and after a race is concluded, the player is given the option to share its race to the server. This happens with a POST request to the server, sending a multipart/form-data web form containing the image file of the drawing and a json package, described below, with information related to game elements present in the racetrack, time to complete the race and the record of the grey ghost car. The server then define a unique identifier to the particular race and stores the image and the json information in separate files but using the same identifier to each filename. This request responds with an identification string, corresponding to the determined unique identifier, depicted in Figure 5.24;

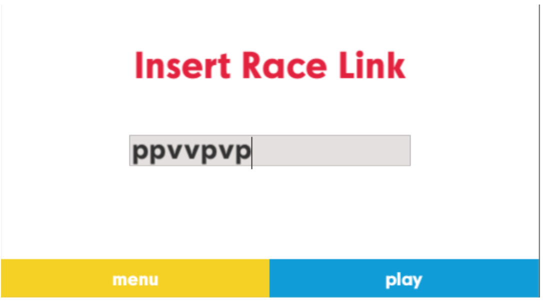
- GET in route /api/play: using the game application, the player is given the option to play a race stored in the server against another player. Making a GET request using the unique identifier results in the receiving of a json package. Figure 5.25 shows how the player makes that request. The player is then prompted to the gameplay screen, and a ghost car represents the same path that the race uploader followed.



Your Race Link

ppvvpvp

Figure 5.24: Upload race screen



Insert Race Link

ppvvpvp

Figure 5.25: Download race screen

All the race information is sent to and from the server via a json package that contains the following attributes:

- coloredImage : byte array. This array contains no information (null), when the player is uploading his race to the server. To download a race stored in the server, this field is filled with the photograph bytes to be used in the race.
- photoFile : string. This string contains the filename of the image stored in the server, if already present. If not present, it is an empty string.
- elements: string. This string is used by the game to store the information of the position, rotation and scale of the game elements present in a level (finish line, oil spills and acceleration tracks).
- time: float. Contains the time it took the uploader to complete this race.
- frequency: float. This attribute contains the time between frames of the recording of the ghost car, it is set by default to 0.1 seconds.
- positionsList: List<Vector3>. This attribute is a list containing the positions regarding the (x,y,z) position axis of the ghost car. The size of this list is variable according to the frequency field.
- rotationsList: List<Quaternion>. This attribute is a list with the rotation regarding (x,y,z,w) rotation axis of the ghost car. The size of this list is also variable according to the frequency field.

In the first version of this server, the frequency of recording of the ghost car positions and rotations was about 60 frames per second. This was implemented this way because, if the frequency was lower, the ghost car would feel "laggy" and would not give a smooth transition between the recorded stages (each stage corresponds to a position and a rotation). This meant that the final json package sent to the server was over 500 kb, for a race with a duration of 40 seconds, making each individual race occupy too much storage space if the game was to be released as is. To solve this problem, we lowered the frequency of recording to about 10 frames per second and apply a mechanism of linear interpolation (using Unity Lerp function, available to both Vector3 and Quaternion variables) between recorded stages so that, at every frame, a new position and rotation are calculated in between the current and the next stages. This reduced the stored file sizes by approximately six times.

5.6 Conclusions

During the implementation period of this internship, various features were implemented to provide a better experience to players of the game. The most important concerned the control schemes to be used during the gameplay phase of the game, as it was the main problem detected in the opening playtests. Other features that increase the playability and replayability of the game were also implemented to provide a better experience to the players. This included the addition of various screens and menus to the final application. Figure 5.26 depicts the navigation flow of the game prior to the implementation while Figure 5.27 shows the final navigation flow.

The implemented Sauvola binarization algorithm, even though it was not planned in the original work plan, provided better results than the previously implemented algorithm and it was seen by the company as an added value to the project and will be part of the official launch of the game as well as future developed games.

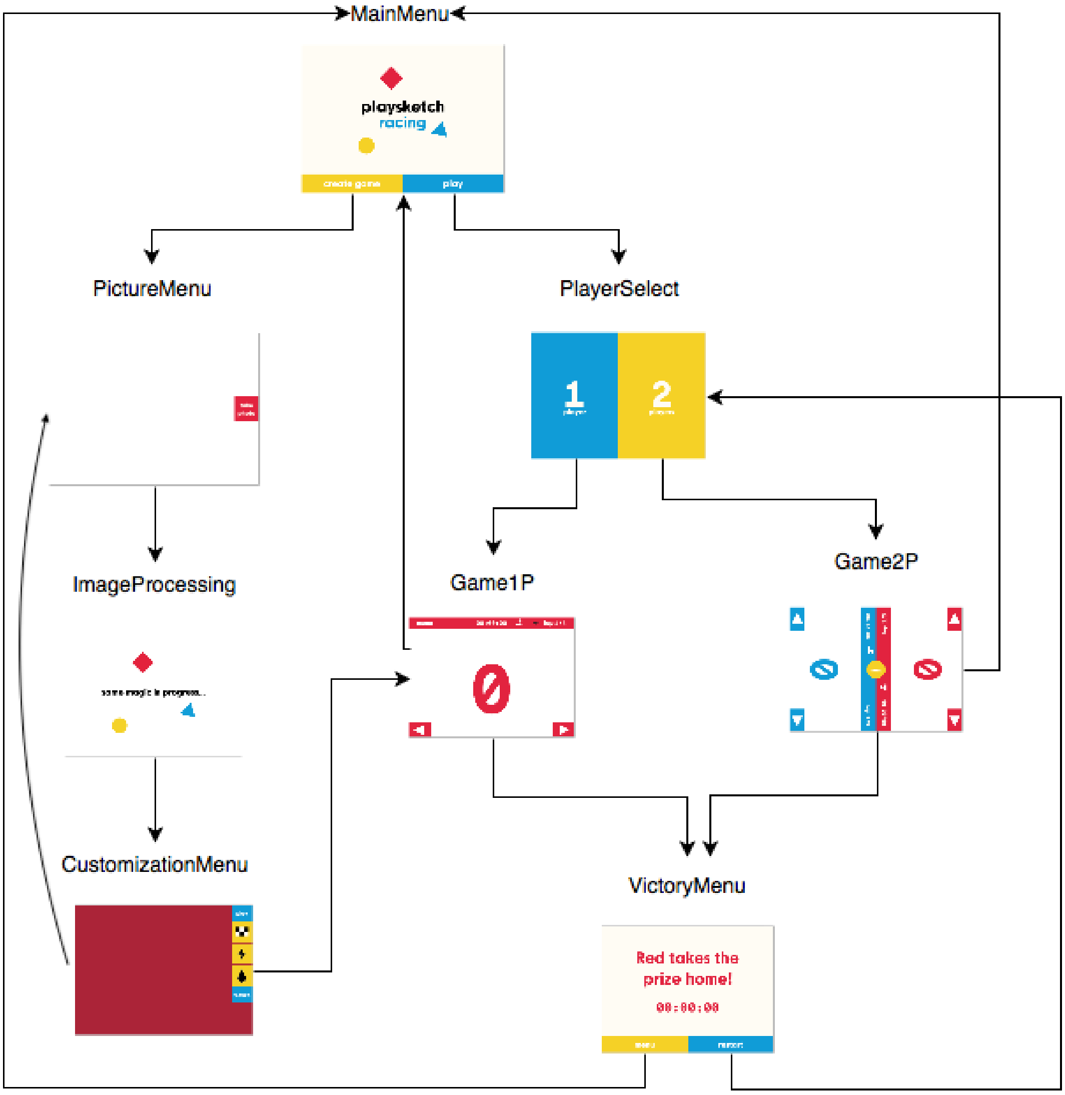


Figure 5.26: Initial Navigation Flow

5.7 New Prototype: Playsketch Labyrinth

One of the objectives to this internship is the development of prototypes for new mobile games for the Playsketch collection. This chapter describes the game concept of the new prototype, depicting its characteristics and interaction. Even though only one prototype was developed, a project template was created to provide a better starting point for new prototypes to be created and various ideas for new prototypes were collected from the playtests.

The developed prototype was based on ball-in-a-maze or labyrinth puzzles that consist of a typically wooden board with a maze and holes. The objective of these puzzles is to manipulate the board so that a ball reach a goal in the maze, without falling in the holes. So, the goal of this prototype is to invite players to use their imagination to create mazes, choose where both the start and end points are located, and where the traps (holes) are placed.

Following the Playsketch concept, players should draw their mazes in paper and take a picture. Prior to that, the player is sent to a customisation menu, depicted in Figure

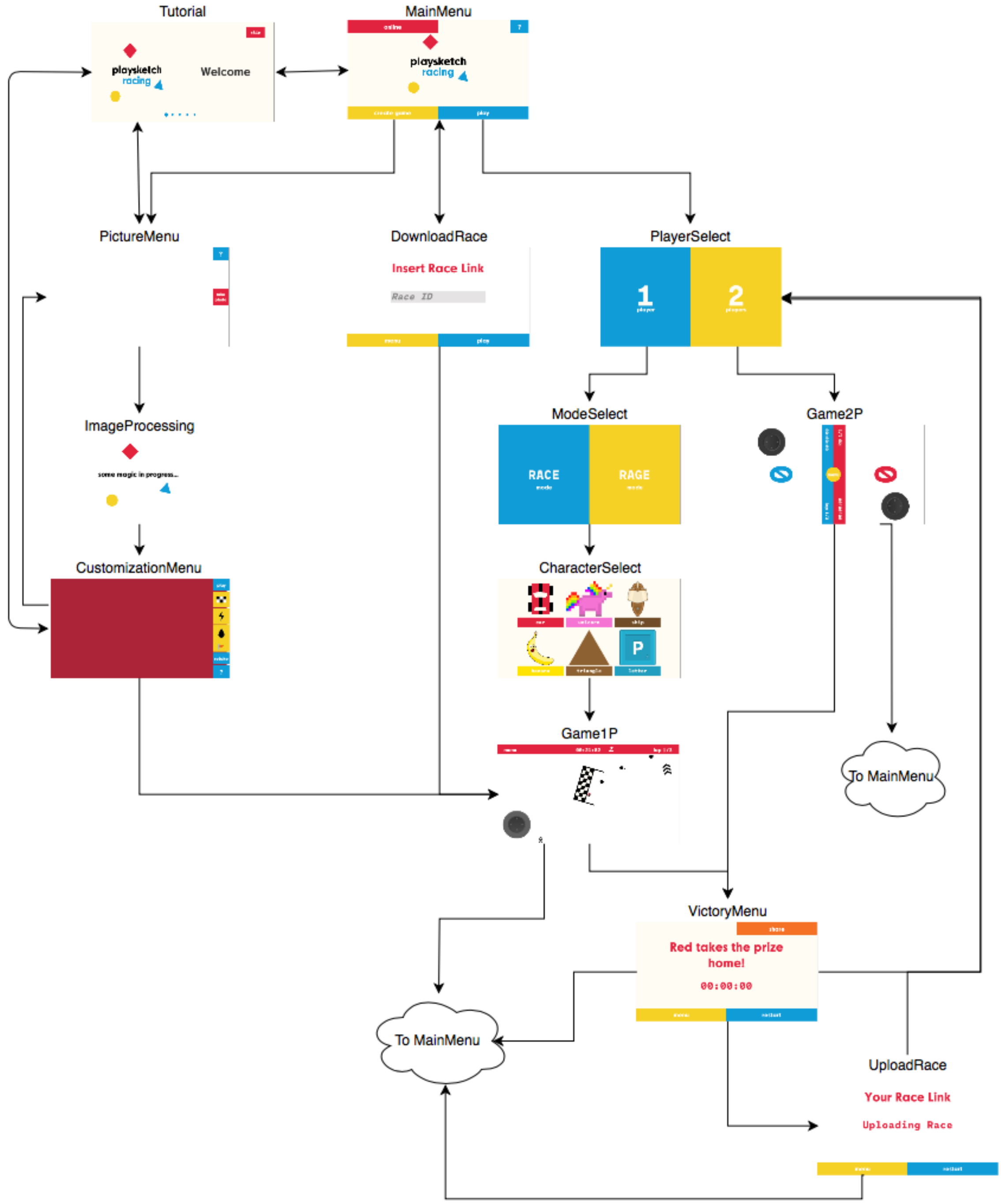


Figure 5.27: Final Navigation Flow

5.28, where he must add a starting point, represented by the blue flag, and an exit point, represented by the "exit" sign. He can then add holes, represented by the black circle, to his maze, if he so desires.

The gameplay was programmed to function as a traditional labyrinth game, as described above. The player must manipulate the ball, red circle in Figure 5.29, by tilting the device, making use of the devices accelerometer. It is assumed that the neutral state is when the device is parallel to the ground. The ball follows the tilting of the device both horizontally as vertically. The degrees to which the device is tilted influence the velocity of the ball, if the device is barely tilted (above the low-pass filter threshold) the ball will move slowly.

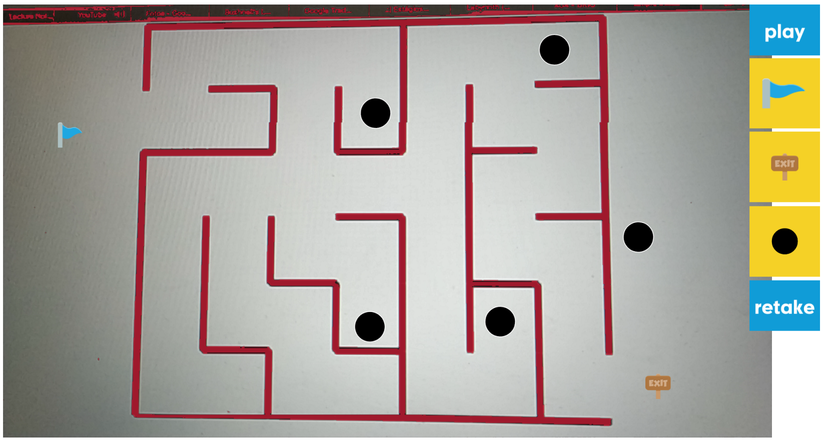


Figure 5.28: Customization screen

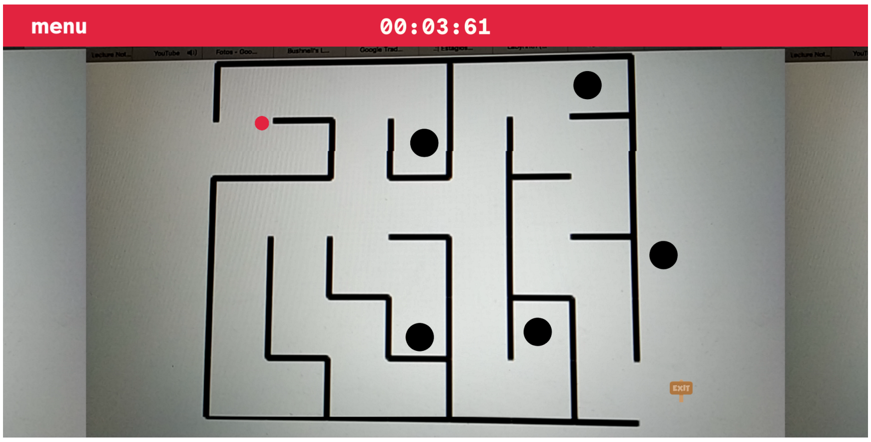


Figure 5.29: Gameplay screen

Regarding the camera, it was implemented as a global top-down view so that the player can see at all times where he stands in the maze and where the starting and end points are located. Using a follow camera, similar to Playsketch Racing, could make players' experience frustration as one could get lost or stuck without indication to where to go next.

Chapter 6

Second Evaluation

This chapter describes the evaluation carried out prior to the implementation of the features proposed in the opening playtest.

6.1 Objectives

The primary goal for this playtesting session is to recon which of the implemented control schemes gives the players a better sensation of control over the player character and, subsequently, more fun. Another objective of the playtest is to perceive if the players understand better the previously available hazards and power-ups now that their interactions with the player character were modified.

Similarly to section 4.1, these objectives were expressed in form of questions:

- **Q1 -** Which control scheme do players prefer?
- **Q2 -** Which control scheme gives players the best results?
- **Q3 -** Do players think the car is moving too slow, too fast or just right?
- **Q4 -** Do players like the interactions with oil spills and acceleration tracks?
- **Q5 -** Do players prefer the old lap count element or the new one?

6.2 Playtest Setup

For this playtesting sessions, both male and female participants with ages between 6 and 14 years old participated. These participants are unfamiliar with both the intern and the game itself. This revealed to be a minor inconvenience as the participants were reticent about sharing their feelings for the game and would normally respond direct questions with apprehension.

In total, 55 children were contacted to take part in this study but, due to time constraints, only 24 were able to take part in the session. Table 6.1 shows the age and gender distribution of the participants. All participants were familiar with mobile devices (smartphones and tablets) and most had a device of their own (only four possessed neither a tablet or a smartphone). Seven participants used mobile devices every day, 2 more than twice a

Age Range	Number of participants	Gender distribution
[6-7]	10	4 Male & 6 Female
[8-9]	10	4 Male & 6 Female
[11-14]	4	3 Male & 1 Female

Table 6.1: Participants age and gender distribution for the second evaluation

week, 11 once or twice a week, and 4 used them less than once per week. Twenty participants stated that gaming is their main activity with the devices and that watching gaming videos online is a usual occurrence. Eight participants referred that they usually play racing games even though most only play this genre in videogame consoles or personal computers.

The informed consent form used in the prior playtests, Appendix A, was delivered three weeks prior to the playtesting session with the necessary changes about the time, date and local of the tests. Of the 24 participants, 4 did not bring the signed consent form but, as they were eager to take part, they participated with no video recording done to them.

The session took place at *Escola Básica Nº2 da Pampilhosa* and *Centro Escolar da Pampilhosa* during class time, as it was not possible to schedule a session after it. While class was going on, pairs of children would leave the class and come to the playtest room.

The tests were conducted using pairs of participants simultaneously, avoiding big group problems noticed in the previous playtest but receiving the benefits from acquainted interaction (Als et al. [2004]) versus the think-aloud method (Van Someren et al. [1994]). This method helped the participants to be more open with the intern, as they could bounce ideas with their peer. Help between participants was not permitted in order to not influence the interaction.

The data collection instruments utilised in the session were: video recording of the participants faces, screen recording of the device during play, surveys and interviews.

After the play sessions finished, the participants filled a survey related with their experience playing the game. The survey contained the following questions, originally in Portuguese:

1. It was fun
2. I felt bored while playing
3. The car moved too slowly
4. The car was too fast
5. The game was too difficult
6. I feel I played well
7. I was quick to understand the game objectives
8. I want to play again

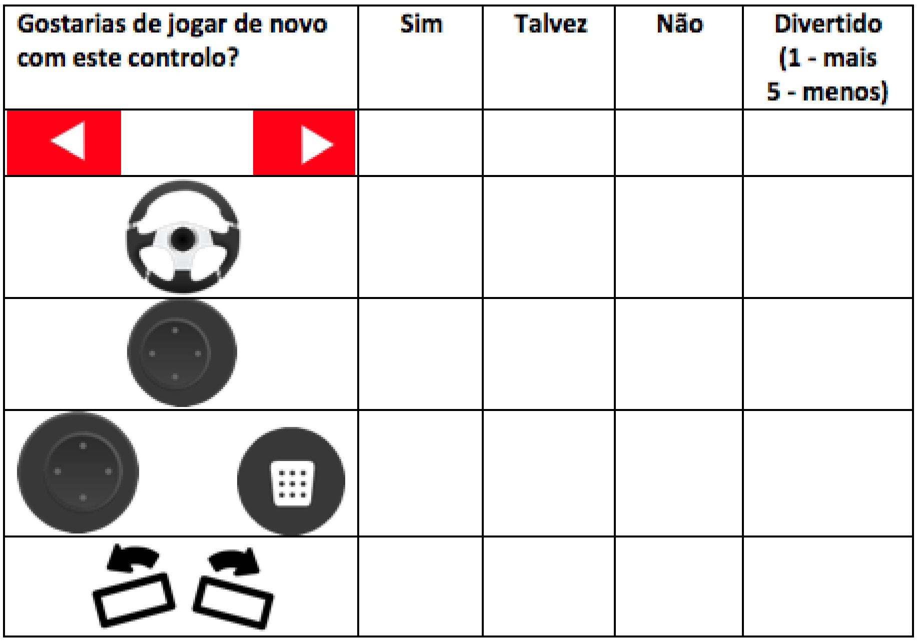
Finally, the participants were then engaged individually by the evaluator to answer some open-ended questions, using auxiliary graphics when needed:

1. Can you order the controllers by your preference?

2. What things did you enjoy the most?
3. What things did you enjoy the least?
4. What would you like the game to have more?
5. Which controller did you enjoy the most? Why?

To answer the first question, the Read and MacFarlane [2006] Fun Toolkit's Fun Sorter and Again-Again Table were utilised, shown in Figure 6.1. The Again-Again Table was dropped after 4 sessions due to participants only choosing their preferred controller to play the game a second time.

As the main aim of this playtest is to understand which of the implemented control schemes is preferred by the players, and because of time constraints, each participant had 45 seconds to 1 minute to play the game using each control scheme. The order which the players used the controllers was random so that the first controllers would not influence subsequent ones.



Gostarias de jogar de novo com este controlo?	Sim	Talvez	Não	Divertido (1 - mais 5 - menos)
◀ ▶				

Figure 6.1: Fun Sorter and Again-Again Table with the evaluated controls

6.3 Results

After all the data from the used instruments is collected, it is analysed and results from that analysis is presented in this section. First, the results taken from the on-site observations and video recordings are described. Next, the questionnaire results and finally, the interviews. Conclusions are going to be derived from these results and corrections will be proposed.

Observations

Through observations, the following problems were detected with each controller:

- Virtual buttons: as the buttons are set in the lower part of the screen, they usually go undetected for a while, making the players try to tilt the device; random pressing of the buttons; continuous pressing of one button without lifting the finger (causing the car to spin).
- Steering Wheel: this control gives the players better control over fine movements but children like to see the steering wheel rotate so they usually overshoot their curves.
- Joystick: No major problems detected with this controller; some participants referred it was too small.
- Joystick with acceleration button: inexperienced players tend to accelerate – stop – turn – stop – accelerate, so on, instead of accelerating and turning at the same time.
- Tilting: some participants tried to press the visual aid that indicated to tilt the device; fine movements are hard to achieve using this control, tilting too little barely turns the car while tilting too much makes the car spin. It is extremely difficult for children to find the "sweet spot" where the tilt is just right to control the car; the order which this controller was played influenced the participants perception: if this was the first controller, they would understand almost immediately what they needed to do in order to turn the car but, if used later than first, the participants would try to press the auxiliary icons that describe the "tilting" motion.

One participant found the button the evaluator used to change the controls and tried to revert to an easier controller when the evaluate changed it to a different one. A pause occurred in the session and the participant was asked to cooperate with the experiment procedures or abandon the study, if he so desires. The participant continued the tests following the agreed procedure.

Some participants kept referring to the game as being too difficult until they got used to one controller. After the habituation period was over, it was common to hear the participants state that the game was not hard after all.

Questionnaire

The questionnaires were handed after both participants finished playing using all the available controllers. To each question of the questionnaire, it was given a score from 1 to 3 points, where 1 represents a negative experience and 3 a positive one. For instance, if a participant answers "Yes" to the question "It was fun", it receives a score of 3, as if one answers "Yes" to "I felt bored while playing", it receives a score of 1 point. So, 24 is the minimum score a question gets and 72 is the maximum. Figure 6.2 depicts the overall total score of each question.

Only 6 participants stated that it was more or less fun to play our game, every other participant stated that it was fun. One participant said that he got bored while playing the game, while 4 others more or less.

Questions 3 and 4 regarded the car movement, most of the players who regularly play racing games said that the car moved too slowly while less experienced players stated it was too fast. In conjunction with the observation notes, most of the less experienced players only thought the car was moving too fast while using certain controllers, specifically steering wheel and tilt, as they could not control the car and kept bumping into the track limits. Only 5 participants expressed that the car did not move too slowly neither too fast.

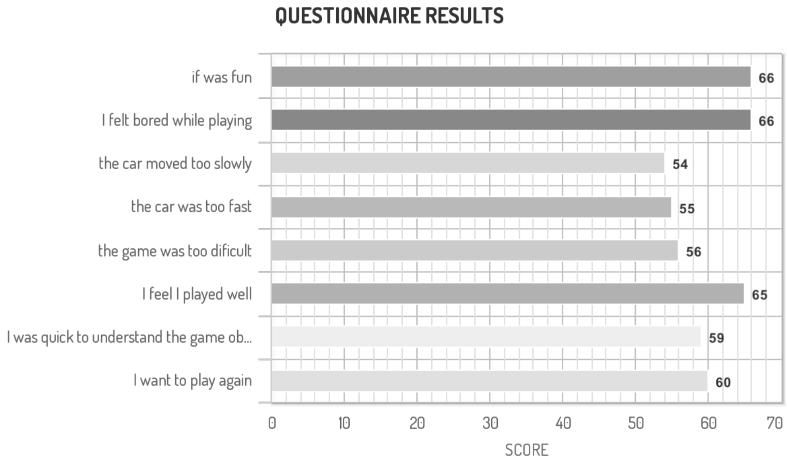


Figure 6.2: Questionnaire Results

Half the participants mentioned that the game was moderately or even too hard. By observation, in most cases, the difficulty of the game was only influenced by the controllers, as the racetrack was very simple: it consisted in a standard oval shape track, as shown in Figure 6.3.

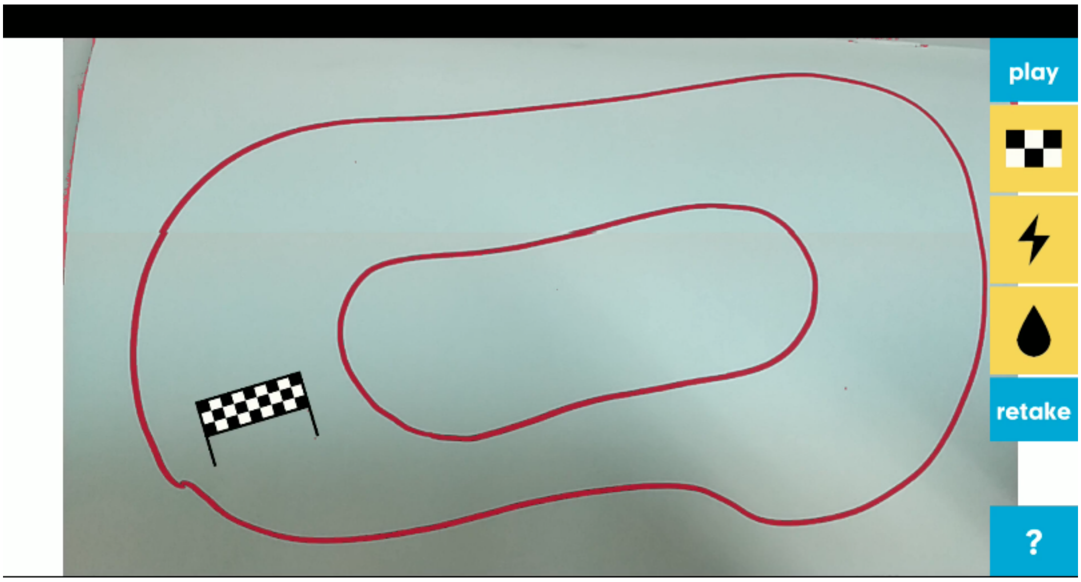


Figure 6.3: Map used in First Iteration Playtest

Eight participants expressed their disinterest in playing the game a second time.

Interviews

At the end of each session, participants were asked to order the used controllers by their preference. The results are shown in Table 6.2. The results indicate a clear preference for both joystick options while the accelerometer-based tilting was voted by most as the worst controller.

In regard to the preferred controller out of both joystick options, we assign greater weight to the choices of the younger participants because the game was designed with the younger players as its main target audience and the older participants had more experience with

Preference	Buttons	Steering Wheel	Joystick	Joystick + Button	Tilt
First (Most Fun)	1	5	9	8	1
Second	3	1	7	11	2
Third	7	6	4	3	4
Fourth	7	9	4	1	3
Fifth (Least Fun)	6	3	0	1	14

Table 6.2: Frequency of controller preference

racing games in game consoles, that use joysticks and acceleration buttons as their main control scheme. Due to the closeness between preference of both joystick options, the factor of playing experience was taken into account. Table 6.3 shows the preference of the participants that stated they never or barely play racing games.

Preference	Buttons	Steering Wheel	Joystick	Joystick + Button	Tilt
First (Most fun)	0	3	8	4	1
Second	1	1	5	8	1
Third	5	4	1	2	4
Fourth	7	5	2	1	1
Fifth (Least fun)	3	3	0	1	9

Table 6.3: Frequency of controller preference to inexperienced participants

Out of the 8 participants that regularly play racing games, 4 selected joystick+button as their preferred controller while none chose the joystick. This means that inexperienced players prefer the joystick without the acceleration button over the other options. These results in conjunction with the evaluator observations of the participants performance, show that the joystick provides better results as some participants could not perform both turning and accelerate actions at the same time.

When asked about why they prefer certain controller, most participants stated that they did not have any specific reason to why. The participant that preferred the virtual buttons expressed the ease to use of that controller. The participant that chose the tilt controller said that turning the device from side to side is fun, even though he could not control the car. Some participants who chose the steering wheel stated that using it felt like they were driving a real car. The participants who preferred the joystick with the acceleration button enjoyed it because the button gave them the impression that the car moved faster than with the other controllers while the ones who chose the joystick without the button stated that it was the easier controller to use as the car moved with relation to where they placed their finger in the thumbstick.

From the questions regarding the personal experience with the game, participants stated that what they enjoyed more was the music and sound while the car movement and level design were the least enjoyable characteristics of the game, disregarding their least favoured controller.

Regarding suggestions for new features for the game, most participants did not know what we could add to make the game more enjoyable, while some stated the inclusion of obstacles, narrow pathways, multiple levels, upgrades for the car, more animations, artificial intelligence opponents and powers as seen in other racing games. One participant mentioned the inclusion of social features to the game such as online multiplayer modes and teams as well as rewards for playing the game on a daily basis.

There were no questions regarding the interactions with the oil spills and acceleration tracks but, through observations, it was obvious that all participants understood these game elements. Every participant showed excitement as the car sped up when interacting with the acceleration tracks and only one participant stated that he was surprised that interacting with the oil spills made the car spin. When asked what did he expect from that interaction, he answered that he thought the car would only speed down, but he preferred the spins - "Spinning is cooler"[1].

6.4 Conclusions

This evaluation took longer to happen than expected as it is very difficult to get a hold of a significant number of participants with ages in the target audience range.

This playtest was very useful to better understand would to solve the game biggest playability fault: the control scheme. Through this evaluation we understand that joystick controllers are associated with a better experience as well as better performance with children. While joystick+button had a better score overall but, when analysing the data taking into account players' experience and age, the analogue joystick was selected as the default controller for the game.

Tilting was voted by players the least fun controller but, through observations, we think that it has potential for the younger audience but it needs to be fine tuned and tested in following evaluations.

Even though the car movement received mixed reviews, it should be further evaluated with only the new default controller.

One flaw with this experiment is that, while the order of the controllers was random between participants, the usage of each one was not fully balanced. The frequency of order placements can be consulted in Table 6.4. This had no major consequences on the results because the controllers who were most used in last place were the joystick and the tilt. These controllers were both used 6 times in last place but had completely different results, showing that the joystick was not the preferred controller just because it was the last one used the most.

Order Used	**Buttons**	**Steering Wheel**	**Joystick**	**Joystick + Button**	**Tilt**
First	7	2	2	4	5
Second	5	6	4	3	2
Third	5	4	4	4	3
Fourth	1	4	4	7	4
Last	2	4	6	2	6

Table 6.4: Frequency of controller placement in order of usage

[1] Translated from portuguese: "Andar às voltas é mais giro"

This page is intentionally left blank.

Chapter 7

Conclusions and Future Work

Since its inception, the aim of this internship project was to evaluate the playability and player experience of a work-in-progress mobile game, Playsketch Racing, designed for children and to implement improvements and new features to create the best experience possible. To achieve this goal, a state of the art analysis about game design and player experience was conducted, with similar video games being used as a basis to probable improvements to come. An opening playtest with children with ages in the target audience range was carried out so to understand their experience with the game. This playtest showed that the participants found the controls hard to understand and use and some game elements were not fully understandable either.

With the results of this evaluation, alternatives for the controls, new interactions with existing game elements and new features, such as a multiplayer server and a new algorithm for drawing binarization, were implemented during the second semester. Some of these new implementations were also targeted of an evaluation in a playtesting setting with children.

A prototype for a new video game for the Playsketch collection was also developed following the design of a traditional board game, Playsketch Labyrinth. In the future, this game should also be evaluated in a playtesting session.

For more future work, the results of the second semester evaluation *can be* taken into account to improve even further the player experience of Playsketch Racing, specially regarding the movement of the car and the fine tuning of the alternative controls—if the joystick option is proven to not be the right fit after launch—, and continuing the work, refinement and further testing, of the implemented game mechanic and UI proposals—car selection, tutorial, rage mode, multiplayer, etc—leading to further improvement of the overall gameplay experience. The launch is planned to happen in the near future with some of the proposed features. The implemented features that will not be present in this launch will be further developed by the team.

At a personal level, I think this internship was a success as it allowed me to contact the game development industry and its procedures, deepen my knowledge about game design and experience evaluation, and technologies such as Unity3D and NodeJS. Through all of it, I understood better the processes of evaluating video games with people and experienced in first-hand the difficulties in the gathering of children as playtesters. My experiences in events such as Festival da Ciência in Oliveira do Bairro and NerdAlert in Terceira island, Azores and the interactions with some of the biggest drivers of the national games industry were really eye-opening about how to enter the game industry and how to thrive in it.

This page is intentionally left blank.

References

Als, B. S., Jensen, J. J., and Skov, M. B. (2004). Usability evaluation with children. In *Computer Interaction Research Symposium*, page 29. Citeseer.

Chaki, N., Shaikh, S. H., and Saeed, K. (2014). A comprehensive survey on image binarization techniques. In *Exploring Image Binarization Techniques*, pages 5–15. Springer.

Clevyr (2018). Entity-Component-System For ReactJS. `https://medium.com/@clevyr/entity-component-system-for-react-js-e3ab6e9be776` [acessed on 4 June 2019].

Craigs games (2013). Head to head racing. `https://play.google.com/store/apps/details?id=com.craigsrace.headtoheadracing&hl=en_US`.

Csikszentmihalyi, M. and Csikszentmihalyi, I. (1975). *Beyond boredom and anxiety*, volume 721. Jossey-Bass San Francisco.

Desurvire, H., Caplan, M., and Toth, J. A. (2004). Using heuristics to evaluate the playability of games. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 1509–1512. ACM.

Desurvire, H. and El-Nasr, M. S. (2013). Methods for game user research: studying player behavior to enhance game design. *IEEE computer graphics and applications*, 33(4):82–87.

Desurvire, H. and Wiberg, C. (2009). Game usability heuristics (play) for evaluating and designing better games: The next iteration. In *International Conference on Online Communities and Social Computing*, pages 557–566. Springer.

Federoff, M. A. (2002). *Heuristics and usability guidelines for the creation and evaluation of fun in video games*. PhD thesis, Citeseer.

Fry-up Productions (2017). High Octane. `https://play.google.com/store/apps/details?id=com.fryup.android.highoctane&hl=en_US`.

Fullerton, T. (2014). *Game design workshop: a playcentric approach to creating innovative games*. AK Peters/CRC Press.

Geemzo (2012). Micro racing. `https://itunes.apple.com/ID/app/id515927981`.

Greensted, A. (2010). Otsu Thresholding. `http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html` [acessed on 6 March 2019].

Griffiths, G. (2014). Child's Play: Playtesting with children in the world of Skylanders. `https://www.gdcvault.com/play/1020348/Child-s-Play-Playtesting-with` [accessed on 20 November 2018].

Hunicke, R., LeBlanc, M., and Zubek, R. (2004). Mda: A formal approach to game design and game research. In *Proceedings of the AAAI Workshop on Challenges in Game AI*, volume 4, pages 17–22.

IJsselsteijn, W., De Kort, Y., and Poels, K. (2013). The game experience questionnaire.

J2sighte (2010). Doodle Kart. `https://itunes.apple.com/us/app/doodle-kart/id360564992?mt=8`.

Korhonen, H. (2016). Evaluating playability of mobile games with the expert review method.

Korhonen, H. and Koivisto, E. M. (2006). Playability heuristics for mobile games. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 9–16. ACM.

Martinho, C., Santos, P., and Prada, R. (2014). Design e desenvolvimento de jogos. *Lisboa, Portugal*.

MdLabs AB (2015). Vs. Racing 2. `https://itunes.apple.com/us/app/vs-racing-2/id497693461?mt=8`.

Medlock, M. C., Wixon, D., Terrano, M., Romero, R., and Fulton, B. (2002). Using the rite method to improve products: A definition and a case study. *Usability Professionals Association*, 51.

Mihealsick, M. (2016). Scientific Method in Iterative Playtesting. `http://coalitiongames.com/wordpress/2016/08/16/scientific-method-in-iterative-playtesting/` [accessed on 18 December 2018].

Mr Qwak Limited (2015). Retro racing. `https://play.google.com/store/apps/details?id=com.mrqwak.retroracing`.

Nacke, L. E. (2010). From playability to a hierarchical game usability model. *arXiv preprint arXiv:1004.0256*.

Nordin, A. I., Denisova, A., and Cairns, P. (2014). Too many questionnaires: measuring player experience whilst playing digital games. In *Seventh York Doctoral Symposium on Computer Science & Electronics*, volume 69.

Ofcom (2017). Children and Parents: Media Use and Attitudes Report. `https://www.ofcom.org.uk/__data/assets/pdf_file/0020/108182/children-parents-media-use-attitudes-2017.pdf`.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66.

Paavilainen, J., Korhonen, H., Koskinen, E., and Alha, K. (2018). Heuristic evaluation of playability: Examples from social games research and free-to-play heuristics. In *Games user research*. Oxford University Press Oxford, UK.

Pereira, L. L. and Roque, L. (2012). Towards a game experience design model centered on participation. In *CHI'12 Extended Abstracts on Human Factors in Computing Systems*, pages 2327–2332. ACM.

Pereira, L. L., Santa, P. M., and Roque, L. (2014). What You Draw Is What You Play: a natural approach to participatory game creation.

Pinelle, D., Wong, N., and Stach, T. (2008). Heuristic evaluation for games: usability principles for video game design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1453–1462. ACM.

Pinelle, D., Wong, N., Stach, T., and Gutwin, C. (2009). Usability heuristics for networked multiplayer games. In *Proceedings of the ACM 2009 international conference on Supporting group work*, pages 169–178. ACM.

Rajanen, M. and Nissinen, J. (2015). A survey of game usability practices in northern european game companies. In *IRIS: Selected Papers of the Information Systems Research Seminar in Scandinavia. Issue*, number 6.

Rajanen, M. and Tapani, J. (2018). A Survey of Game Usability Practices in North American Game Companies.

Ravado, R. (2016). 10 Free Accelerometer Apps for iPhone and iPod Touch. `https://turbofuture.com/cell-phones/10-Best-Accelerometer-Apps` [accessed on 26 June 2019].

Read, J. C. and MacFarlane, S. (2006). Using the fun toolkit and other survey methods to gather opinions in child computer interaction. In *Proceedings of the 2006 conference on Interaction design and children*, pages 81–88. ACM.

RedBit Games (2014). Mini turbo gp. `https://itunes.apple.com/br/app/mini-turbo-gp/id827765763?mt=8`.

Salen, K., Tekinbaş, K. S., and Zimmerman, E. (2004). *Rules of play: Game design fundamentals*. MIT press.

Sánchez, J. G., Zea, N. P., and Gutiérrez, F. L. (2009). Playability: How to identify the player experience in a video game. In *IFIP Conference on Human-Computer Interaction*, pages 356–359. Springer.

Sánchez, J. L. G., Vela, F. L. G., Simarro, F. M., and Padilla-Zea, N. (2012). Playability: analysing user experience in video games. *Behaviour & Information Technology*, 31(10):1033–1054.

Sauvola, J. and Pietikäinen, M. (2000). Adaptive document image binarization. *Pattern recognition*, 33(2):225–236.

Schell, J. (2014). *The Art of Game Design: A book of lenses*. AK Peters/CRC Press.

Sweetser, P. and Wyeth, P. (2005). Gameflow: a model for evaluating player enjoyment in games. *Computers in Entertainment (CIE)*, 3(3):3–3.

Swink, S. (2008). *Game feel: a game designer's guide to virtual sensation*. CRC Press.

Tavares, J. P. and Roque, L. (2007). Games 2.0: Participatory Game Creation. In *Proceedings of the 6th Symposium on Computer Games and Digital Entertainment*.

Van Someren, M., Barnard, Y., and Sandberg, J. (1994). The think aloud method: a practical approach to modelling cognitive.

Wijman, T. (2018). Newzoo's 2018 Report: Insights Into the $137.9 Billion Global Games Market. `https://newzoo.com/insights/articles/newzoos-2018-report-insights-into-the-137-9-billion-global-games-market/` [accessed on 15 November 2018].

Appendices

Appendix A

Informed Consent Form for Participation in Study

TERMO DE CONSENTIMENTO INFORMADO PARA PARTICIPAÇÃO EM ESTUDO

Título do estudo: Playsketch – Desenho e Avaliação de Experiência de Jogo

Enquadramento: O meu nome é Fernando Rocha e este trabalho desenvolve-se no contexto do meu projeto de Estágio Final do Mestrado em Engenharia Informática da Universidade de Coimbra.

Playsketch Racing é uma aplicação para dispositivos móveis (tablets e smartphones) que convida o jogador a explorar a sua imaginação e criatividade, criando o seu próprio jogo de corridas por via de desenhos em papel. Para isto, basta ao utilizador desenhar uma pista de corrida num papel, tirar uma fotografia e a aplicação trata de criar um jogo de corridas tendo por base o seu desenho.

O objetivo deste estudo é avaliar como as crianças, entre os 6 e os 12 anos, reagem à vers ao atual da aplicação, passando por uma observação em contexto real desta interação. Assim, este documento tem a finalidade de garantir que os encarregados de educação tomem conhecimento de toda a informação necessária acerca do estudo com vista à permissão da participação dos seus educandos no mesmo. A participação neste estudo é completamente voluntária, podendo a qualquer altura tanto o educando como o encarregado de educação recusar a sua cooperação.

O estudo decorrerá na ..., no dia ... de ... de ..., entre as ... e as ..., onde as crianças estarão em grupo, brincando com a aplicação enquanto o responsável pelo estudo observa, faz perguntas e tira notas. Este estudo será filmado de forma a garantir que todas as interações possam ser analisadas posteriormente. Todas as gravações serão apenas para a minha própria visualização e análise, sendo eliminadas assim que toda a informaçaão necessária ao estudo seja analisada.

Os dados de todas as crianças presentes serão totalmente anónimos e para uso exclusivo neste estudo, significando que os participantes não poderão ser identificados e os seus comentários serão confidenciais.

Gostaria de agradecer a disponibilidade prestada para a leitura deste documento. Se existir qualquer dúvida ou necessidade de esclarecimento poderá contactar-me para o meu número de telemóvel ou então para o meu endereço email.

Despeço-me, agradecendo pela sua colaboração.

Fernando Miguel Cardoso Rocha

Eu,

tendo sido devidamente esclarecido sobre os procedimentos do estudo, concordo que o meu educando participe voluntariamente no mesmo.

Assinatura:

Data: ____/____/_____

Appendix B

Preliminary Evaluation Data

Questionnaire Results								
Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
1	3	3	2	3	2	2	3	3
2	3	3	1	3	2	3	2	1
3	3	3	2	3	3	3	2	3
4	3	3	2	1	3	3	3	3
5	3	3	3	3	3	3	3	3
6	3	3	3	3	3	3	3	3
7	3	3	2	3	3	3	2	3
8	3	3	1	3	3	3	3	3
9	3	3	1	2	3	3	3	3
10	3	3	3	3	3	3	3	3
11	3	3	3	2	3	3	3	3
12	3	3	3	2	3	3	2	3

Table B.1: Participants answer to each question from the first evaluation

Times to complete each racetrack		
Participant	Racetrack 1	Racetrack 2
1	07:30	05:45
2	02:14	02:22
3	02:21	05:13
4	01:48	01:58
5	01:56	01:11
6	01:12	01:27
7	01:36	01:27
8	01:02	02:56
9	00:47	00:42
10	00:52	00:42
11	00:42	00:50
12	00:34	00:46

Table B.2: Times to complete each singleplayer racetrack (mm:ss)

Times to complete multiplayer		
Pair	Winner	Time (mm:ss)
1-3	3	05:19
2-4	4	02:44
5-6	6	01:23
7-8	7	01:43
9-10	10	00:50
11-12	11	00:42

Table B.3: Times to complete the multiplayer race

Appendix C

Second Evaluation Data

Questionnaire Results								
Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
1	2	2	1	3	2	3	3	2
2	3	3	1	3	2	2	3	3
3	3	3	2	2	3	3	2	3
4	2	3	2	3	2	3	3	3
5	2	1	2	2	1	2	1	2
6	3	3	2	3	2	3	3	3
7	3	3	3	2	1	3	3	1
8	3	3	3	3	2	3	3	3
9	3	2	1	1	2	3	3	3
10	3	3	2	1	3	3	3	3
11	3	3	3	2	3	3	3	1
12	2	3	2	3	3	2	1	1
13	3	3	3	3	3	3	3	3
14	3	2	1	2	1	2	1	1
15	3	3	3	2	2	3	3	3
16	3	3	3	2	3	3	3	3
17	3	3	2	2	3	2	3	3
18	3	3	3	3	3	3	3	3
19	2	2	3	2	3	3	3	2
20	3	3	2	2	1	3	1	3
21	3	3	3	3	3	3	3	3
22	2	3	1	1	2	1	1	2
23	3	3	3	3	3	3	2	3
24	3	3	3	2	3	3	2	3

Table C.1: Participants answer to each question from the second evaluation

		Preference of controller			
Participant	Buttons	Steering Wheel	Joystick	Joystick+Button	Tilt
1	3	4	2	1	5
2	4	3	1	2	5
3	3	4	2	1	5
4	1	4	3	2	5
5	4	3	2	1	5
6	4	5	3	2	1
7	3	4	1	2	5
8	4	2	1	3	5
9	2	3	4	1	5
10	5	1	3	2	4
11	2	3	4	1	5
12	3	4	2	1	5
13	4	1	2	3	5
14	3	4	1	2	5
15	5	1	3	2	4
16	5	4	1	3	2
17	4	3	1	2	5
18	5	1	4	2	3
19	4	5	1	2	3
20	2	5	1	4	3
21	3	4	1	2	5
22	3	1	4	5	2
23	5	3	2	1	4
24	5	4	2	1	3

Table C.2: Participants preference of controller

Participants order of controller usage					
Participant	Buttons	Steering Wheel	Joystick	Joystick+Button	Tilt
1	1	4	2	3	5
2	3	5	1	4	2
3	3	2	5	4	1
4	1	3	2	4	5
5	2	3	5	4	1
6	1	2	3	4	5
7	2	3	5	1	4
9	2	4	5	1	3
10	1	5	4	2	3
11	3	2	4	1	5
13	5	1	3	2	4
14	1	2	5	3	4
15	3	5	2	4	1
16	2	4	1	3	5
17	2	3	5	1	4
19	3	5	4	2	1
20	5	2	3	4	1
21	4	1	3	5	2
23	1	2	4	5	3
24	1	4	2	3	5

Table C.3: Participants order of controller usage from 1(first used) to 5 (last used)