ISCTE ◉ IUL

University Institute of Lisbon

Department of Information Science and Technology

A Proposal for a Framework for Serious Game development based on entertainment Game Design

Ricardo Francisco Vilelas de Almeida

A Dissertation presented in partial fulfillment of the Requirements
for the Degree of
Master in Computer Science

Supervisor
Prof. Dr. Pedro Faria Lopes, Associate Professor
ISCTE-IUL

September 2015

"A well designed world could tell its story in silence."

Hidetaka Miyazaki

Resumo

Os videojogos tornaram-se uma grande parte da sociedade moderna. Altamente valorizada para propósitos de entretenimento esta indústria expandiu rapidamente. Jogos sérios, um tipo de jogos com um propósito mais focado no ensino do que em mero entretenimento, foram criados e publicitados como ferramentas de ensino. No entanto estes últimos não partilham do mesmo sucesso dos primeiros. Nesta dissertação gostaríamos de propor uma framework para o desenvolvimento de jogos sérios assim como de jogos em geral. Construída através da análise de técnicas de *design* de jogos de entretenimento e de jogos sérios, a nossa meta é contruir uma ponte entre as filosofias de desenvolvimento de cada género. Com isto o objetivo principal será ajudar desenvolvedores a produzir jogos sérios que tenham o mesmo sucesso que jogos focados em entretenimento.

Palavras Chave: Sério, entretenimento, jogo, design, eduentretenimento, framework, desenvolvimento.

Abstract

Video games have become a big part of modern society. Highly valued for entertainment purposes this industry has expanded very fast. Serious games, a type of game with a purpose more focused on teaching rather than mere entertainment, have been created and marketed as tools for teaching. However the latter hasn't enjoyed the success of the former. In this dissertation we would like to propose a framework for the development of serious games and games in general. Built through the analysis of entertainment game design and serious game design we aim to bridge the gap between the development philosophies of each genre. With this the main goal will be to help developers produce serious games which will enjoy the same success as entertainment centric games.

Keywords: Serious, entertainment, game, design, edutainment, framework, development.

Acknowledgements

I would like to acknowledge first and foremost my colleagues who worked with me on the original version of *Gestual Life*, Pedro Sousa, Luísa Brito and Marco Capela, without whom this dissertation would not exist as it is now.

To Professor Pedro Faria Lopes whose attention to detail and constant constructive criticism let me write something I can be proud of, more than that a big thank you for the trust he put into my capabilities and most of all the passion I show towards Video Games.

This whole dissertation however, I would like to dedicate to the following people, João Folgado who constantly won at everything but never let me quit. Marina Ferreira who takes the time to listen to my misguided opinions and supports us all, even in a literal way regarding Dota 2. Pedro Pereira with whom I am able to discuss about every media aspect for nights on end.

Carlos de Almeida and Teresa de Almeida, thanks you both for simply being the parents you are, for the support, the patience and love, I made it, thank you!

Fofo, my dog, without you making me laugh time and time again with your antics during this dissertation I would probably have given up a few times buddy.

To my cousin Bruno Vilelas, thank you for always believing I am bigger than my own size, hope to make you proud!

Finally last but certainly not least, to Sara Militão, who by now knows a lot more about video games than she cares to. Thank you for putting up with me, for always smiling when I most need it and for the enormous amount of support you always showed towards me and my passion. Sorry about all those night long lectures about meaningful choices and experiences the player should have.

To all of the above, my most heartfelt, honest, thank you.

Contents

List of Figures

Abbreviations

GDD   **G**ame **D**esign **D**ocument (see page 70)

NPC   **N**on-**P**layer **C**haracter (see page 26)

FPS   **F**irst **P**erson **S**hooter (see page 71)

RPG   **R**ole **P**laying **G**ame (see page 71)

MMORPG   **M**assive **M**ultiplayer **O**nline **R**ole **P**laying **G**ame (see page 71)

Chapter 1

Introduction

1.1 Motivation

Serious games, what they are, what they require and what they actually need to do is still debated today. There have been reports and studies on its definition [42] as well as the creation of serious games with context sensitive game design and framework structure [20].

For this work in particular serious games shall be defined as (digital) games used for purposes other than mere entertainment [42].

Entertainment centric games, on the other hand, have their definition well established and are part of a very profitable industry[40]. Students today are also native digital speakers, born surrounded by technology and making use of it naturally in their lives [45].

Serious games however don't have the same success as games focused on entertainment. Some [42] argue that it's the way knowledge is portrayed and transferred from game to student, others argue [42] that their very nature forbids a focus on creating the gaming experience as a whole and would rather focus on filling the game with all the right information.

Building immersive and engaging experiences in order to induce a flow state into the player is more than possible through game design [22][24], the next step will be utilizing these states' strengths in order to introduce learning to the player and having him explore the games features in order to achieve said knowledge gain.

1.2 Context

Game design is a school of thought that brushes psychology, indeed game designers have been referred to as "deviant psychologists", it focuses on manipulating emotions so the player has fun and feels engaged in the world[15][25][38]. Serious games utilize this concept but in turn focus on transmitting knowledge and assessing if said knowledge has remained with the player [4].

Thus marrying the two spectrums could theoretically produce a fun, engaging and successful learning experience, and this is the goal of this work.

1.3 Research Questions

This project mainly focuses on the following five questions:

1. What are the principal guidelines for development of a serious game?

2. What are the principal guidelines for development of an entertainment focused game?

3. Is it possible to utilize entertainment based game design in serious games?

4. Which practices are transferred directly and which are adapted to achieve learning during play?

5. What set of core design rules deprived of context can we apply to universal serious game design? content...

1.4   Objectives

The main objective of this work is the formulation and production of a serious game design framework, a set of guidelines and components built not as a formula, but as a set of tools to compare, analyse and structure a serious game both for engagement and learning, while creating commitment from the player towards the experience and therefore maximizing the amount of knowledge gained by the player.

In order to achieve its creation first the common practices in serious game development shall be studied and compared to the practices of entertainment game development. The key is to establish common ground between successful products in both domains and gather what Digital Game Based Learning (DGBL) [42] can learn from its counterpart, the (entertainment) Video Game Industry.

Once common ground has been established a Game Design Framework will be proposed based around those principles and subsequently used to build and improve on an existing serious game designed fully with entertainment rules as a main guideline [2].

To assess its effectiveness tests will be done and then compared to the old results obtained through testing traditional game design choices previously implemented.

1.5   Research Method

The research will rely heavily on the Design Science Research method, although its focus will be around an objective-centred solution, through research it was possible to identify both problems and a motivation basis for the project.

The research steps are as follows:

1. Problem identification and motivation

2. Objectives of a solution

3. Design and development

4. Demonstration

5. Evaluation

6. Communication

The first two steps have been covered in this dissertation, they describe the need for such a solution and why that same solution is an important area of game development that this approach will improve upon. Following these there will be more research in regards to practices and production cycles, framework development will begin and after a first version, be used to build a game. During the development both the game and the framework will be iterated upon each contributing to each other's completion.

Lastly evaluation will be done through a test phase and communication will be addressed in the form of a dissertation presenting the framework and its results.

Chapter 2

Literature Review

2.1 Lingua Franca

Ever since it's growing industry has been noticed, games have been consequently studied further and further [6][9][25][38]. Sometimes for pattern recognition or formalization others for their impact in diverse fields such as sociology [7] or cultural significance [21].

These studies all agree on the fact that a *Lingua Franca*, an unified language for game design is needed to push this medium forward, there are comparisons to Mendeleev's *Periodic Table* [38] and attempts at producing patterns or solutions intended for appliance in game development [6][9]. These however are not fully built and accepted solutions as of yet.

As a new medium, both analysing and recognizing patterns are vitally important for formalization however, not even genre description standards are set currently, most are established by the press [23] and aren't at this time questioned by the consumers. This creates several problems the most prominent one being that the unique aspects of what makes a game stand out within it's genre are lost when describing it by it's mechanical features alone [23][22][10].

There have been efforts to formalize and systematize concepts and rule sets both for categorization of genres and game development, these are accepted and used but not widely spread as the norm [22][6].

2.2 Game Design

Developing a game requires a game designer, his objective is to create an experience and to make playing the game mean something to the player [38]. Aside from simply building the game one has to also know how to pitch the idea to the client, the serious games industry suffers with a lack of interest for their products [42], hence knowing how to gather the public attention is very important even before the game is finished. This can be used to gauge the expectations and to give direction to the project.

Game design is akin to psychology, it's main function is to create an experience that has the intent of manipulating the player into having fun, feeling scared, exhilarated or sad among others [15][9][38]. To achieve these "meaningful experiences" designers employ the use of challenge and knowledge acquirement to constantly evolve the player and keep him in a state of *flow* [12].

As mentioned previously there isn't an understanding in the game development industry about which mechanics or systems provide the best experience or even if they will ever exist. Groundwork has been laid by other authors in pattern definitions [6], game development books teaching how to harness the strengths of interactive media such as games [9], how to design meaningful experiences [38] or simply how fun stems from learning and therefore is inherent to video games [25].

To design an experience it is important to ask "what is an experience to the player?", what does the player take or search for in a game? The MDA framework offered three layers of abstraction in order to analyse the experience gathered from a game, they are Mechanics, Dynamics and Aesthetics [22] these are not to be confused with visual aesthetics [39], Aesthetics in this context are aesthetics of

play [10], the truer reasons why people play and enjoy games. Mechanics refer to the actual actions the player can perform (e.g. jumping, shooting, moving etc.), Dynamics are a set of actions that the player can execute through a combination of the mechanics (e.g. jumping and shooting at the same time to become harder to hit) finally Aesthetics are the results of combining Dynamics, these are what describe the experience gathered from the game (e.g. challenge, sensation, narrative etc.).This framework offers a new and more thoughtful approach to genre depiction and game analysis offering perspectives from both the player and the designer standpoint.
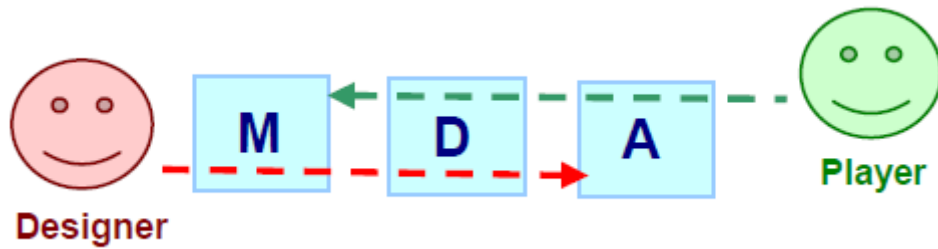
FIGURE 2.1: Game abstraction layers seen through the designer and player's perspective[22]

Designers and players look at games from opposites sides of the spectrum that is why designers must be able to see through the eyes of players in order to build a resonating experience [15][38].

2.3 Serious games and Learning

Serious games have risen as a possible tool for teaching and knowledge transmission [4][42]. These games are built with the intent to transmit knowledge and act as teaching tools for players.

There is a struggle in defining what constitutes a game in its nature, the word "fun" always seems to be used either as a critical component or as a detriment

[42][25] to games, however as we've seen fun isn't something you artificially imbue into the game but rather something that appears from mastery, overcoming obstacles and, ultimately learning .

Learning and mastering the game should be and is its own reward[25][38], and this ties in heavily with serious games and their design. By nature serious games reward the player with mastery over the content of the game itself and not with something arbitrary like "highscores" or equipment for their characters (although these can be part of the design as well), something serious games can take advantage of to become more enticing and adept at transmitting their information.

Great entertainment games have been designed and critically acclaimed without having fun as a central focus to the experience, such as the amazing "Alien: Isolation" by Creative Assembly [5][35]. Furthermore designers have acknowledged that, while difficult to achieve, games don't need to be fun in order to be successful, they need to be engaging [34]. Engagement should come from the task itself and from the context surrounding it. Once engaged the player will actively seek out the solutions for himself and adjust his strategies based on feedback received from the game.

Serious games that rely on fun in order to achieve learning have had marginal success but not enough testing and proof has been given [2]. *Gestual Life* is a game that has fun and traditional game design values as a tenant. The game was designed thinking about what the intended experience was rather than what knowledge was going to be taught. We let the experience and theme guide us to both the content and the way that content would be presented to players. "Fun" wasn't our only goal but it was one of the tools we resorted to, we started by asking "How can we make this game fun?" but after some iterations the question changed into "How can we make this interesting?", fun isn't always what the player looks for but it is a great asset to always have in the design tool kit.

However fun shouldn't be necessarily ruled out nor made the basis of design when building a serious game [42], the lack of consensus in this matter leaves room

for improvement on design philosophies as well as experience driven serious game design.

The components needed for a serious games are a matter of debate, as well as their seemingly contradictory name. Some like Stoll [41] defend that they need to be designed around the learning material or not found in classrooms at all, while others like for instance Ben Sawyer [29] state "serious" and "game" not only belong in the same name but that the "serious" refers to the purpose of the game rather than it's content. On the other side of the spectrum there are those like Zyda [47] stating that learning needs to always be subjective to fun and story, citing the entertainment value of a game above all else. However the consensus is that they need to engage the user, to motivate voluntary action to repeat the task and seek knowledge from the experience.

At the base of this debate is the need to compare serious games to entertainment games, as previously mentioned there have been debates over whether or not the word "serious" and the word "game" aren't in themselves mutually exclusive. Indeed the goals and needs of each are different and each have their own set of rules leading to their creation.

The following table shows the main differences between entertainment games and serious games [42].

As described in Table 1, the goals for each game type are different both in scope and in constraints, while serious games can't make any concessions on reality, entertainment games can use randomness, time compression or simplified simulations in their course.

	Serious Games	Entertainment Games
Task vs. rich experience	Problem solving in focus	Rich experiences preferred
Focus	Important elements of learning	To have fun
Simulations	Assumptions necessary for workable simulations	Simplified simulation processes
Communication	Should reflect natural (i.e., non-perfect) communication	Communication is often perfect

TABLE 2.1: Differences between entertainment games and serious games

There seems to be a crevice between the two types of game, however there is a bridge, the Flow Framework for Game Based Learning [24], a set of principles used to build engaging game elements taking into account associative learning approaches and designing challenges in order to stretch the player's mind to its limits. This framework utilizes the concepts of the theory of flow heavily [12], it stresses the idea that the reward is the action itself not what is to be gained from it, this is the ideal scenario for serious games seeing as learning and knowledge seeking by the players is the end goal.

In order to induce this state in the player the challenge and skill requirements of the task (playing the game) need to be set according to the player's performance. If the task is too easy the player will feel bored, if the task is too hard the player will feel overwhelmed by anxiety. By balancing the game around the skill set necessary to properly convey and teach the information present during gameplay, it's possible to push the players to their limits and make them lose themselves in the experience, playing for the action in itself and not for the rewards.

In conclusion there are several ingredients necessary to immerse the player in the experience, to induce a flow state of mind, to properly convey information and to create a learning space that can both be entertaining and informative at the same time, these need to be provided by the game.

Chapter 3

Preliminary Work

In this chapter some solutions will be proposed based around Chapter 2 and work done previously on *Gestual Life*[2]. As such based on the MDA Framework [22] and it's stated perspectives from the Designer and Player standpoint, a mockup of what that same perspective would look like with "Knowledge" added into the game development as a core feature was developed.

This can be seen in Figure 3.1.
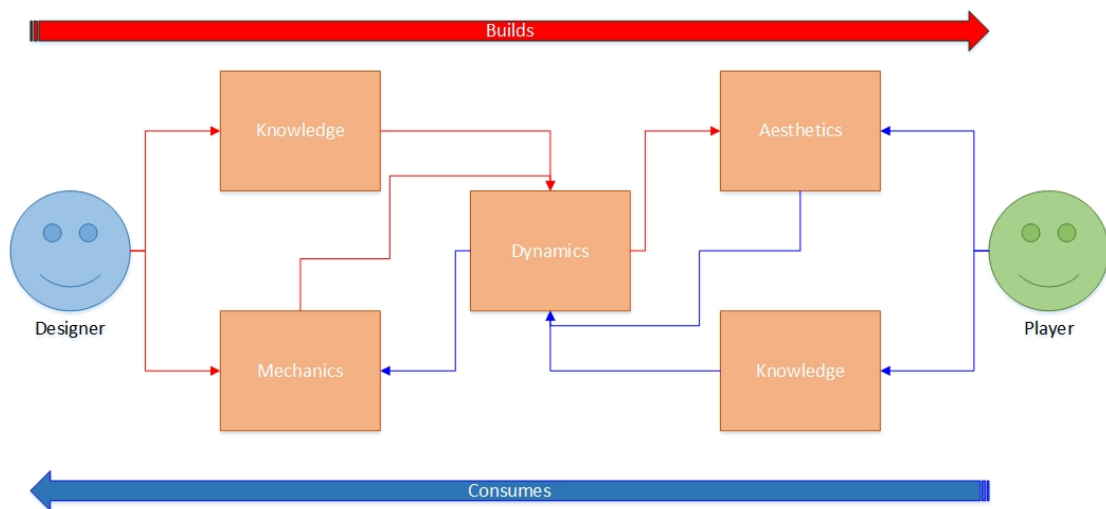
FIGURE 3.1: Perspective from the Designer and Player

The most important to convey was that from a designer's perspective Mechanics and Knowledge are both the first concerns and what he will be implementing,

testing and designing first, these are the *core* of the game. From the player's perspective however, the Knowledge and the Aesthetics are the first he will come in contact, this symmetry in the Knowledge factor comes from the fact that, as soon as the player picks up the game he will instantly be in contact with both as the *"feel"* of the game can be gathered instantly as it will be discussed in Chapter 4 and the Knowledge present in it is apparent at the start just by the nature of the game.

The previous chapter also had a table depicting the differences between serious and entertainment games, as said the main purpose in this work is to bridge the gap between them. As such an identical table is presented but with an extra column added where a middle ground between both types of games is proposed.

	Serious Games	Entertainment Games	Serious Entertainment
Task vs. rich experience	Problem solving in focus	Rich experiences preferred	Rich problem solving experiences
Focus	Important elements of learning	To have fun	Having fun emerge from learning
Simulations	Assumptions necessary for workable simulations	Simplified simulation processes	Start simple and work towards complexity
Communication	Should reflect natural (i.e., non-perfect) communication	Communication is often perfect	Start out perfect and gradually introduce noise (i.e., to the point of natural and domain specific communication)

TABLE 3.1: Differences between entertainment games and serious games

In this approach the intent is to find common ground between both types, although serious games aren't allowed to make concessions in order to achieve their full purpose, they have no need to be entirely constrained from the start, by introducing a scaling difficulty level into the design, the approach from game to real world will feel organic and familiar to the player [2].

The idea behind table 3.1 is that if both types of games make concessions then it is possible to reach an enlightening and fulfilling gaming experience the player will want to explore for himself, gaining knowledge as he does so.

Before the aforementioned proposals a serious game project was developed in the context of the Master Degree in Computer Engineering - Multimedia Branch subjects of, Computer Games, Multimedia Management and Interactive Applications, *Gestual Life* [2]. A game built with the objective of teaching Portuguese sign language to both hearing and deaf people, this game was built by myself Ricardo de Almeida, Pedro Sousa, Luisa Brito, Marco Capela and supervised by both Professor Pedro Faria Lopes and Professor Isabel Alexandre.

This framework will be used to build and test a new increment to this project while formalizing steps for development.

This game was built with aid from the APS (Associação Portuguesa de Surdos)[1] and tested with their students. This game was built with only traditional game design in mind and adding in knowledge as an aesthetic and core part of the gameplay.

In order to progress through the game the player had to acquire and use new knowledge, also while trying to keep the player in the flow state [12] the game's difficulty would adjust itself according to the player's performance while solving challenges in the explorable world.

The whole game was designed while thinking of what it means to be a deaf person. As such the main character is deaf, can't hear anything apart from muffled sounds coming from the world and has to rely on an intelligent agent partner, L.Y.S. (acronym for Learn Your Signs). She acts as a window into the world, communicating with everyone for the player and then transmitting back that information to the player through text. When the player has to answer to someone, L.Y.S. presents the player some options from which the correct one needs to be chosen. These options are videos of signs being performed and are the sole means of communication available to the player.

The visual nature of the game, the way information is accessed and the adjusting difficulty ensured the players never felt bored and kept pushing through trying to solve the next challenge and furthering their knowledge about sign language.

In addition the authors [2] hoped that the game would bring deaf people and hearing people closer together by letting them experience what the former face in everyday lives.

It's upon this game that the new framework produced by this work will be applied and tested on, resulting also in an increment to *Gestual Life*.

Chapter 4

The Framework

4.1   Students and Players

Games exist all around our everyday lives, in fact one could even argue that our whole "job system" is a game, the parallels are there. Each work assignment is the "current quest", which we as the players, "grind" or complete until we accumulate enough days or "points" of successful questing in order to receive a reward, which in most cases is the salary, but what about the Wizards, the Rogues or the Warriors? Well, our function within a company or team is our "Class", whichever skills we possess that other people don't and are in turn complemented by their own set of skills are what define your class, and as such our team is our "party".

Game's roles in society are simple, to teach while being able to fail. Take for instance the game of "tag", in modern era it's a game just about physical agility without any context aside from "don't get caught", however, it's easy to imagine that in the stone ages this game was created with the purpose of teaching children how to chase prey and evade one so as not to become one themselves. The same thing can be said of "hide'n'seek" being able to hide your presence might have been a very useful skill for a society exposed to the elements or warring clans. It's easy to argue that we as a species and since the dawn of time, have invented games and that those two examples were invented with the intent of teaching back

in the olden days, but what about nature? Everyone has seen animals playing, but they aren't just wasting time or indulging they are too, learning [9]. Nowadays we still play games that teach us survival skills, however with the advent of digital games we have started to incorporate different skills and intensities of learning into the gaming experience [25], some of these can be "decision making" and "resource management". While these are also present in games outside of the digital medium (*Monopoly* and *Chess* for example), they can be further enhanced by taking advantage of the different scenarios one can create within a video game, by designing around which skill the player should learn or practice different levels of intensity can be further specified.

Students nowadays are part of this parallel but they lack the most important aspect of it, we have assignments, we have rewards, we have teams and we have "grinds", what students don't feel they have is the ability to fail and learn from those mistakes. Games can fill this role perfectly letting students try and fail as much as they need while pursuing knowledge and learning. As stated in Chapter 2 some defend that "fun" can't be part of the learning experience, while others state the exact opposite and even state that "game-playing is a vital educational function for any creature capable of learning" [9].

This dissertation aims to help game designers take hold of this conscious need to learn already present in humans, by focusing on the strengths of the medium, letting students feel that failing is part of the learning process and letting that failure occur in a controlled environment, students and players alike can learn from a gaming experience rather than a simple lecture. Learn by doing.

4.2   World Map

At the most abstract level Figure 4.1 is how the framework will look, a series of linear main steps accompanied to the sides by two equally important steps that will accompany the whole development process from start to finish.
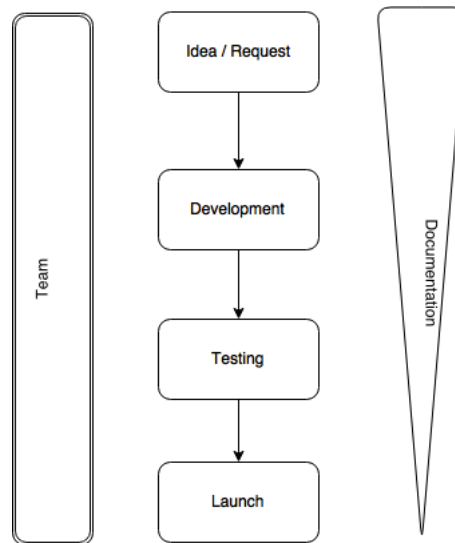
FIGURE 4.1: The topmost level of the framework

The triangular shape of the documentation step will have a thorough explanation in the sub section 4.2.6, it's shape simply represents both the importance and natural behaviour of the documentation load from start to finish, meaning, at the start of the project there will be a lot more documentation than in the later stages of development.

The team isn't so much a step as an ongoing process, at all stages of development it is important that the team feels as one and works together towards a collective goal and vision, not a cluster of different facsimiles of visions.

By traversing through this map and coming out the other side, the journey should have yielded a game, capable of immersing the player, teaching the player and best of all, done so by the player's own volition to keep playing and therefore, learning.

4.2.1 Idead/Request

Every game starts with an idea(Figure 4.2). It can be an idea for a story, a technology one might want to explore, or a learning subject the designer would like players to learn. Often designers receive requests from stakeholders.
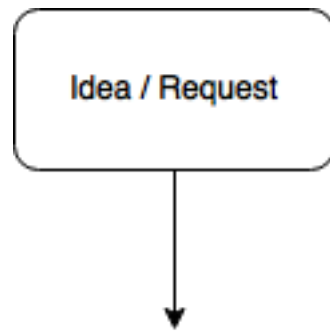
FIGURE 4.2: The first step to any game development process

If a request is given it is important to let the client or stakeholder, be a part of the development process. While including the client in the process is important one must not forget that they often aren't knowledgeable in game development, this means that while their input is valuable try to filter what they really want and need.

The main thing a client can supply the team is direction, establish early on from the clients request and input what is the intended final goal. From here on limit the client input to when strictly necessary.

These situations can be when the team doesn't know which direction to follow or when the client is having reservations regarding development or the direction the project is taking.

This step in the framework seems simple however if enough work is put into this simple step it will alleviate the whole development process later on.

The simple process of thinking about an idea can go something like this:

1. Think about an idea.

2. Try said idea out (any way you can, paper, cardboard, drawings, just enough to give context and a clear vision).

3. Reiterate and change the idea according to results or observations.

This simple method may appear amateurish, however it is what real game designers do [38], simplicity and elegance often lead to the best solution.

20

Fully thinking and idea through, having a clear goal in mind is tantamount to a game development process.

4.2.2 Development

The Development step fully expanded can be seen in Figure 4.3. Each of these steps will be fully explored in the following subsections. Adding to this explanation, how they affected the development of *Gestual Life*'s new increment as well as noting whether or not the previous version's development process retro-actively fits these steps is also going to be discussed.
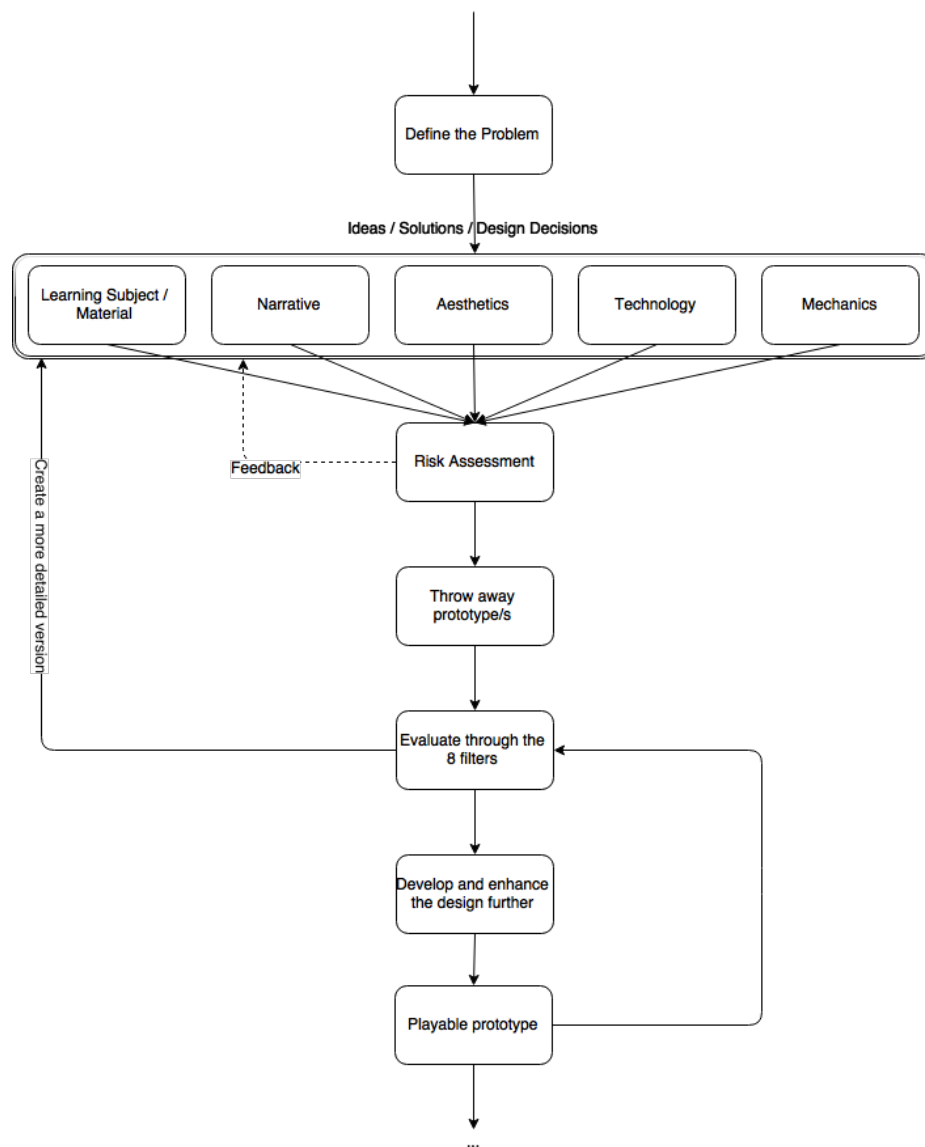
FIGURE 4.3: The development phase fully expanded from the world map

4.2.2.1   Define the Problem

Designers and developers exist for one reason, to solve problems, after receiving either a request or an epiphany, every game developer/designer must first clearly state the problem (Figure 4.4), that is, ask the main question, what am I trying to accomplish with this game?
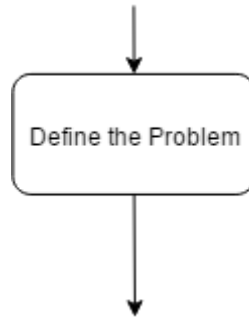
FIGURE 4.4: The first step in development, is knowing what to develop for

Defining the problem will help the team focus on what is the core necessities of the game, as well as measure the final goal of the project.

According to Jesse Schell [38] there are three distinct advantages of stating clearly what is the core problem to be solved:

1. **Broader creative space:** When someone receives an idea or a task they are often quick to jump to solutions and start their creative process there. However, if the process starts by defining the problem there will be a clearer vision of what creative space can be worked with, at times some solutions may even be hidden in the problem itself.

2. **Clear measurement:** By stating the problem the quality of each idea can be measured clearly with the simple question "How well does this solve the problem?".

3. **Better Communication:** When working with a team if the problem hasn't been clearly defined, often team members will be trying to solve different

problems and not even be aware, by stating the problem to be solved, the team has a clear goal they can work together towards.

In regards to the game "Gestual Life" [2](previous version), the problem was defined as follows:

"How can I teach someone sign language and make them feel like a deaf person does?".

This was the question that needed answering, all of the design decisions from this point on were focused on solving this problem.

By adopting this approach and by the tests obtained, the team succeeded in working toward their goal.

In the new version produced with the framework present in this dissertation the question remained the same, with only increments or differences felt further on in the framework.

4.2.2.2 Learning Subject

Defining the Problem can be seen as "asking a question", and in this step the main objective is to give answers (Figure 4.5) to that problem.
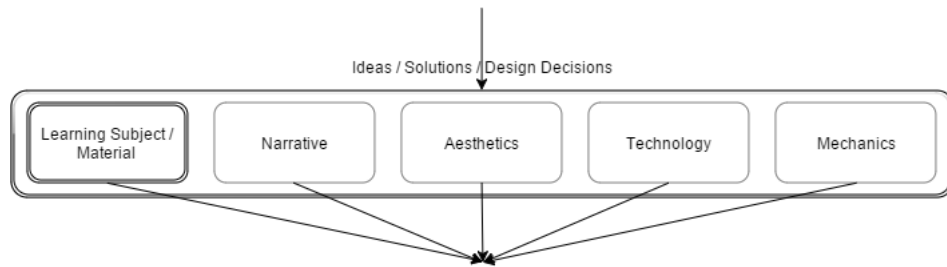

FIGURE 4.5: What do we want to teach?

Although the "Learning" part of a serious game can and is contained in the other components of this step, we (Ricardo Almeida and Professor Pedro Faria

Lopes) decided to give it it's own separate step and it's own solutions so as to give it as much attention as any other aspect of a game.

There is an inherent problem with learning, people are lazy [25]. We humans are compelled to learn to acquire new knowledge but at the same time if people are pushed into it or forced into learning they usually respond with resistance. One of the possible responses in these situations is cheating, although this is seen as a bad behaviour or unsportsmanlike, from a strict survival standpoint it is a winning strategy. Ever since there were rules for war [44] the winning strategy has always been to subvert or ignore them. Of course while most are for the protection of civilians some conventions like not ambushing or not attacking by night are often times ignored and even Sun Tzu himself is quoted saying "All warfare is based on deception".

This ties in to this step for one simple reason, if the players are able to cheat and circumvent a challenge, they will. The reason they shouldn't is simple, if the real situation happens in the real world they aren't allowed to cheat and the knowledge they didn't gain won't be there to help them.

In order to counteract this designers must design around the problem. Don't encourage cheating instead try to offer different solutions to the same problem, create "fake" cheating. If the players feel like they are outsmarting the designer they will feel more compelled to keep on besting the game through their "smarts" [25] being able to identify and express themselves one of the main reasons people play games [22].

If a problem can be tackled in different ways offer the player the more direct way and implement the other solutions in a more discrete manner, if the players find them they will still answer the challenge correctly but do it on their own terms and therefore not feel forced to learn.

When thinking about the learning subject, it is of the utmost importance to never forget one simple thing, the goal it to design a game. A game can be described as a series of important and interesting decisions [26], this and the fact

that a game allows students to experiment without fear of failing and at the same time capitalizing from learning from their mistakes establishes a strong learning environment where students can explore and further their knowledge without fear of consequences but always being wary of them.

When trying to convey knowledge there is a very important keyword, "pacing", this is the speed at which new mechanics and chunks of knowledge are given or unveiled to the player. A game with poor "pacing" can suffer from two consequences, if the "pacing" is too slow, i.e. the game unveils new mechanics or sections too sparingly then the player will feel bored and stop playing, if the game's pacing is too fast then the player will be overwhelmed thinking the game too hard or confusing to keep on playing [25]. It is important to find a balance between conveying knowledge and giving the player enough time to apply and master it.

When designing a traditional game designers have more control over how much is enough, in order to guarantee that the player knows everything he needs before moving to the next section. Usually this is done in the form of a "Boss battle" where the player has to pull from everything learned up until that point to surpass it, one of the basic examples of this is the Legend of Zelda series by Nintendo [31]. The level of teaching skills needed here are almost never possessed by the designer, as such, work closely with an expert or certified teacher in the field of knowledge the game needs to convey. Just like with the client, receive and analyse input, implement it however it will suit best the experience, just because the name "Boss battle" implies a confrontation with an enemy it doesn't mean it has to be.

Bosses are a final test, the culmination and application of knowledge gained beforehand, design the final tests according to this philosophy. If the game teaches players any skill then that same skill can and will be tested, for an excellent example in this one could look at the amazing "Ori and the Blind Forest" which uses "Boss puzzles" or levels as it's final test [18].

In order to achieve learning, it is imperative that advantage of the medium is taken in the act of teaching, the purpose of this framework is to build a *game*, as such the main draw is *gameplay*. Since this is the "Learning" section and not

the "Mechanics", the main goal is to find a *"resonant theme"* [38], in the original *Gestual Life* [2] our theme was "how do people with hearing disabilities struggle in society?", as such we tried to incorporate our subject material into our theme. This theme carried over into the new increment as well, being an addition and not a redesign, the new increment focuses on strengthening the idea behind what being in this minority feels like.

Gestual Life teaches the player sign language because they *need it*, in order to progress in the game you need to answer challenges, the game never forces the player to complete them but, to get access into new areas, new items, videos and challenges a player must first complete a quest or task beforehand. This often means a player explores the world by himself to find knowledge and answers of his own accord, the theme was further enhanced with the addition of LYS, her being a medium between the player and the world, also acting as a door into the game's *magic circle* [21]. This magic circle is a concept which in simple terms means, the abandonment of preconceived notions or rules in order to accept a new set of notions and or rules contained within a new space, in this context, contained within the digital game's world. LYS acts as an usher if you will, directing players to their "seat" in the world so that they may enter it and experience it without reservations. All of the interactions are handled by her, LYS listens to the world and translates everything to the player via text or instructional sign language videos, likewise the player can only "talk" to LYS directly, by choosing from the available sign videos the player selects what answer they want to give to a certain quest, non-player character (NPC) or challenge. By using this method the player needs to actively seek the knowledge from the world and then re-utilize it to progress, likewise this emphasizes the fact that the player character is deaf and limited in his interactions by LYS. Without her, he simply can't communicate, more aspects of these decisions will be discussed further on.

In sum, the main purpose of the step is to try and integrate the learning subject wherever and however possible always striving to adhere to the central theme of the game, the knowledge needs to be part of the gameplay, the aesthetic or the narrative not just a glossary of information.

4.2.2.3 Narrative

Storytelling has always been in tandem with teaching. When most people are young are young their grandparents and parents would tell us stories like "The red riding hood", "The boy who cried wolf" or "Hansel and Gretel" to warn us of dangers and teach us valuable moral and behavioural lessons.

Stories allow us to give examples while setting the stage and context for those examples, incorporating story (Figure 4.6) into game design is what will be discussed.
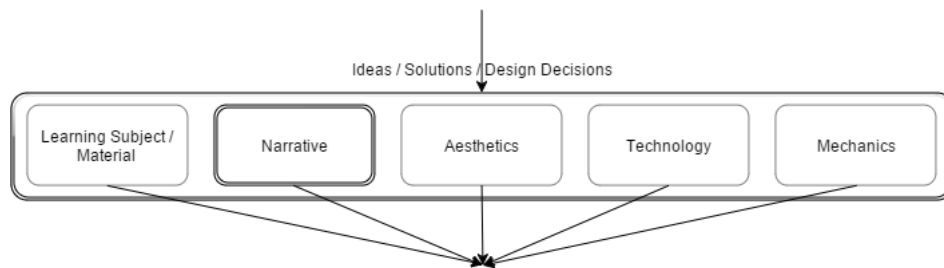
FIGURE 4.6: How will the knowledge be presented to the player?

Nowadays storytelling is still used as a teaching tool but it is much more widespread as either an art form or simple entertainment, sometimes both. However stories haven't lost their edge in terms of teaching, in classrooms the teachers that often get better results are the ones that share stories, that engage student interest and that provide context as to why what they have to teach is important, students don't want stories because of boredom, they need them because they *resonate*(Subsection 4.2.2.2) with them [19].

Game designers design experiences, and stories are one type of experience [38], when designing a serious game there can and should be a story, some context from which the player can gather the "Why" as in *"Why do I need this?"*, however a simple text prompt explaining why the player is playing or studying isn't enough, the narrative needs to *engage* the player, it needs to create interest and investment, here in Figure 4.7 one of the worlds most famous *interest curves* can be seen.

FIGURE 4.7: Star Wars: A new hope - Interest Curve[11]

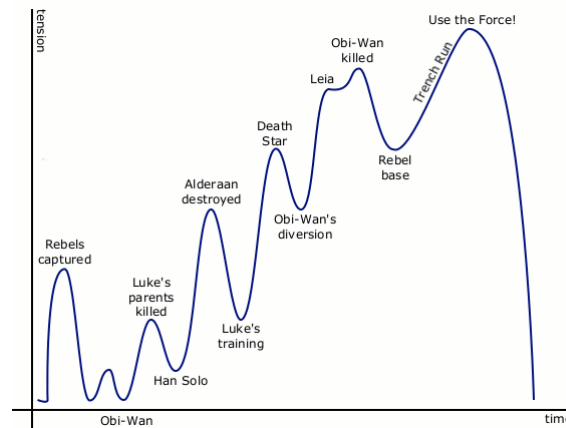Regardless of your stance or personal taste in the aforementioned film, there is no doubt that it was a success, a cultural phenomenon even. At the heart of what made Star Wars famous isn't the lightsabers, the space ships or the aliens, it's the resonant theme within. Portraying the struggle of a simple person (like the viewer) taken from his everyday life and forced to bear the weight of the galaxy on his shoulders (insurmountable challenge the hero needs to face, much like our everyday hardships) but instead of giving up the hero rises up and confronts the challenge with the help of friends and a mentor (very much like the family members or teachers who taught us and we will never be forget) this very human story in a not so human setting was what made Star Wars resonate with people from all over the world.

This interest curve however, is what is important to designers the most, when creating a story for a game it is important to build it in a way that isn't predictable or stable, people crave variation one needs only to look at the word "routine" and see in what context it is usually used.

The player needs to feel the highs and the lows of gameplay, when starting the game the player might witness a big event that raises the implications of failure and be given the tools to overcome that first instance, however if the player never "falls" then the game will become too predictable and the player will lose interest, so why not present the player with a situation next that he isn't expecting and has no chance of winning?

After making the player feel impotent the game should present the player with a way to gather what he needs in order to get back to that same situation in the future and overcome it, maybe even introduce a companion or a mentor that can help the player make sense of situations or objects [2].

All of the above can be seen as the "Hero's Journey" [8], this method of storytelling is widespread and none can argue the success it can have [13].

There are various ways to tell a story, the most common and often used can be called the *string of pearls* [38], this means the story is linear, you have a section of gameplay and then a bit of story, then gameplay, then story and this repeats until the game ends. As with any approach this has problems, the player might not feel that he is part of the story, to alleviate this, choices can be introduced but the consequences need to have impact on the world or it's characters or else the player will feel apart from the story and tricked by the designer.

In the original *Gestual Life* a linear approach to the overall story was used but a free form exploration gameplay driven story was used for each area the player visited. Each area or "level" has a set of smaller stories that contribute to the larger whole, if a player completes every quest he will have a greater understanding of the narrative (and more knowledge of the sign language) if he only completes the main obligatory quests he will have a lesser understanding but will still walk away from the game with a beginning a middle and an end he achieved by himself.

This approach allowed the designers to maintain control over the interest curve and keep it free enough so that players could shape smaller parts of the curve for themselves according to their playstyle and preferences.

The new increment was designed with this choice of storytelling in mind but expanded, when the player is put into the world a simple storyline will be given to the player by LYS. When and how it is completed is up to the player, if the player wants to explore the world first and do the side challenges he/she is free to do so.

However in order to finish the area the player inevitably will have to complete the main story mission, which in this case was a simple "Mail your cousin's medicine".

4.2.2.4   Aesthetics

Usually when aesthetics are mentioned the natural reaction is "we need amazing graphics in our game!", this is wrong, aesthetics are different from graphical fidelity. Aesthetics establish the reality or "feeling" of the game, it makes the player take the game more seriously, graphical fidelity is the amount of polygons that are displayed on screen at any given time, the more polygons rendered on screen the higher the graphical quality of a game.
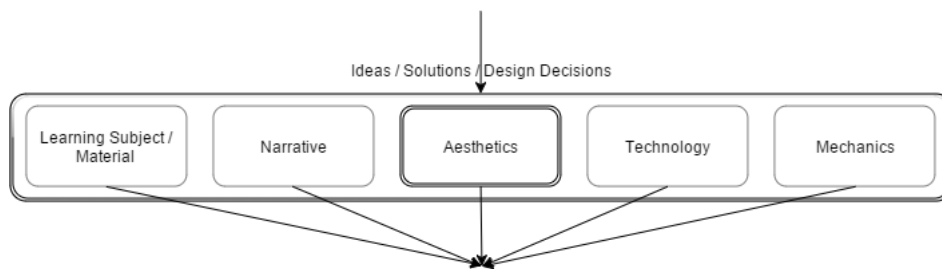
FIGURE 4.8:   What aesthetic style will better convey the experience to the player?

Before explaining how to design aesthetics in the context of game development it's important to take note of Figure 4.9.
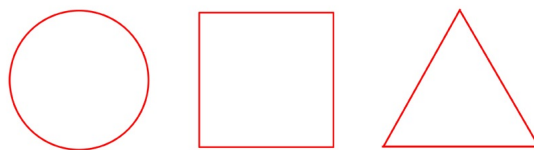
FIGURE 4.9: Circle, Square and Triangle the three most basic forms in art.

Figure 4.9 explains the basics behind visual design from it's conception into current art movements. The circle represents innocence, energy a sense of freedom, the square represents balance, stability, maturity a sense of calm and finally the

triangle represents aggression, masculinity, conflict a sense of danger[39]. These shapes are found throughout visual arts and they are put to great use in game design. Using movement, animation and character design with a notion of what simple lines mean to human interpretation it is possible to further enhance the player's engagement with the game [39].
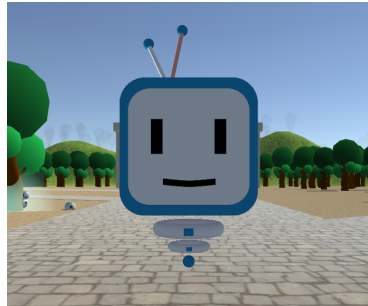
FIGURE 4.10: LYS the protagonist's companion and mentor in *Gestual Life*

In the original *Gestual Life*, LYS the companion was created without knowledge of these forms or of what they meant. She was created by striving to design a simple looking yet friendly character that could relate to the player.

Being the only character in the world that understood the player wasn't enough we wanted her to be a teacher as well, her colour was chosen to be blue on purpose though, as this use of colour allows the designer to convey that she has a calm personality and doesn't punish the player for failing. This was a very important factor since the player needs to experience the game learning from his or her faults without being punished for experimentation or knowledge seeking.

These factors in her personality, being a character you can rely on, a friend and a teacher are curious when looking at her final design and comparing with the shapes above and what they mean.

FIGURE 4.11: LYS's basic shapes appear obvious when broken down

When looking at LYS there is another aspect that is relevant, she does not look realistic at all, her colours are cartoony her shape or locomotion don't exist in the real world, however she got players engaged and learning what she had to teach [2].

To explain this phenomenon the aforementioned aesthetics versus graphics discussion fits perfectly, it is resource saving, cost saving and development enhancing to choose the proper aesthetic for a game. *Gestual Life* had and has a cartoony aesthetic because the goal was to get players to empathize with the world and see it as non-hostile.

Making the game not so taxing on any machine was too one of the reasons for this artstyle, however the biggest reason is by far player immersion, strong aesthetic design has a great advantage over great graphics, the lack of *diminishing returns*.
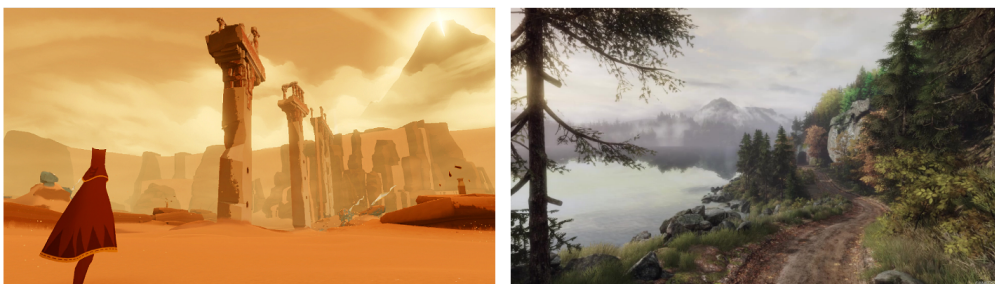

FIGURE 4.12: Left: Journey, 2012 - Right: The Vanishing of Ethan Carter, 2014

In Figure 4.12 it is immediately apparent that *"The Vanishing of Ethan Carter"* has the better graphics out of the two games, the designers even mapped real

photos to produce the textures seen in the world however, *"Journey"* will outlast it for one simple reason, it isn't competing with a graphics engine. As technology advances more and more polygons can be seen on screen this makes it so that great graphics become dated much faster than strong aesthetics, when a new engine comes out a better looking game will be made and *The Vanishing of Ethan Carter* will be noted as "looking really good for it's time", *Journey* however does not suffer from this issue, being stylized and aesthetically pleasing at the same time, it has no grounds for comparison, it simply looks good, more than that it "feels" good and that is what makes a game successful, it's the existence of that *"nameless quality"* [38] that makes a game feel whole.

When choosing the aesthetic of a game remember what theme best resonates with what knowledge the game will transmit as well as what are the capabilities of your team and target audience's expected hardware capability.

When thinking of aesthetics it is important to not forget the audio aspect of a game, audio will convey a sense of "completeness" to the game world, it will turn a game into a serious experience, it will *sell* the world to the player. Usage of audio can vary sometimes the absence of sound is just as important as the presence.

In the 2011 game *"Dark Souls"* the only sounds in the world are diegetic, you are accompanied by your footsteps, breathing and wind sometimes being broken up by scores of music during boss battles or two other specific locations, this is important because before this game no other had attempted this technique with the same success (*Demon's Souls* by the same company had done it but being a Playstation 3 exclusive it was gated to owners of the system). This use of strong artistic design, strong visual direction, use of sound coupled with engaging gameplay and mechanics made *Dark Souls* a staple in what games can strive to be [27].

In the original *Gestual Life*, our use of audio was meant to supplement the experience or to contrast what reality the player knows versus the reality of being deaf. The game starts in a dream sequence where a friend calls to the player and as such sound is present, as soon as the player gets close, he wakes up in his room

and then, no discernible sounds. Being deaf doesn't mean you can't hear at all, just as depending on your level of visual impairment, being blind doesn't mean you can't see at all, the final choice was to represent a middle ground. Over the course of the game players will hear every sound as if their ears were covered, i.e. muffled sounds, with this the game isn't about a child who is deaf, it's about being and *feeling* deaf and therefore needing sign language instead of opting for it.

Aesthetics have a number of advantages inside and outside of the game itself. As such it is of utmost importance that thought is given as to how each piece fits in the overall *feel* of the game, for example, inside the game aesthetics are what brings the player into the world, and, if done correctly never let him return during the course of play. This is what games need to be able to teach players however outside of the game they are as important if not more, having concept art drawn solidifies the idea for the game, it gets developers on the same track and incentivizes both working on and playing the game. A piece of concept art sparks the imagination both for players and designers, both will question what can come next or what can be done in a world where it exists as more than a sketch. On the other hand, having concept art, strong artistic vision and ideas defined and on paper helps with another important matter, securing funding.

Investors won't simply give funds away they need to see what they are paying for and they need to feel safe in that their money isn't being wasted, as such pitching an idea with concept art and aesthetic design decided or solidified enough for a presentation is, essential[38].

Although it will be explored in the next step, concept art is a type of *prototype* and that has advantages in and of itself.

4.2.2.5 Technology

A games is built with technology. Technology (Figure 4.13) can mean a variety of things, it can be a variety of things. However for this dissertation and for the games

intended to be built with this framework computers, are the basic component, this does not mean however, that the framework can't be used for other types of games.
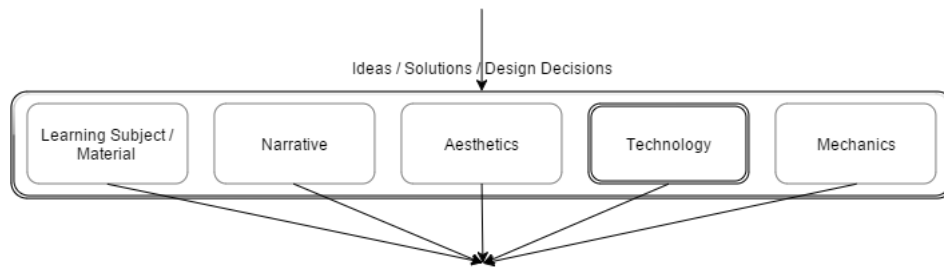

FIGURE 4.13: What technology should be used to develop the game, is it worth the risk?

Board games for example, can be built with this same approach because what matters is that, at the core the goal is to build a game and one that can teach the player while immersing him, technology for these is, cardboard, paper, maybe some plastic and ink, what matters isn't the decoration, it's the behaviour.

Technology has had a very uneven growth throughout history, countless inventions have been born and gathered success within society, countless more have simply vanished. This is called a *Hype Cycle* [38].

FIGURE 4.14: The hype cycle for technology[17]

This graph is extremely important when thinking about which technology would better benefit your game. If a certain technology is currently at the "Peak of Inflated Expectations" you should really think better about whether or not investing time in learning and implementing that same technology will really be necessary for your game to work. As a cautionary tale the recent advent of 3D cinema and subsequent television sets that didn't last nearly two years can be seen

as an example, eager to ride that peak to the top companies jumped on the idea, one or two (depending on company) years later they abandoned it [30] [32] [46].

The main lesson from this tale is, don't get caught by novelty. There are two main types of technology to consider when developing a game, foundational and decorational [38].

Foundational technology is what makes a game work, what supports the game at a core level, for example, the original *Gestual Life*'s foundational technology is 3D movement in a virtual world. We wanted to use a first person perspective so that the player could see the world through the character's eyes and feel what it would be to live in that situation. We built upon this foundation and eventually arrived at decorational technologies, in a future installment or patch *Gestual Life*, will support gesture recognition through Microsoft's Kinect, this is something that will make the game better, but isn't required for the important base experience we wanted for our players.

The new increment uses the same technological choices however since the time investment in learning has already been spent, the development process is less risky and more fluid.

Always consider which technologies will support the game in delivering the experience intended, don't get caught by novelty or marketing words, the next big advancement seems to be Virtual Realities (Oculus Rift, Project Morpheus, Microsoft's HoloLens etc.), but does this mean games need to use this new innovation to prosper? The simple answer is no, Dungeons & Dragons still uses paper and imagination as it's technology of choice and even in current times it thrives [16].

Bottom line is, always weigh the pros and cons of the technology and ask yourself if it is necessary to make the game work or if it will simply make it better, if neither of these happen it is best to just select another technology altogether.

4.2.2.6   Mechanics

Mechanics are in essence what the "game" aspect of a game is, they are the connection between player thought and world interaction. Their execution is what makes the game move and what makes the player play the game, it's the jumping, running, interacting and exploring(Figure 4.15).
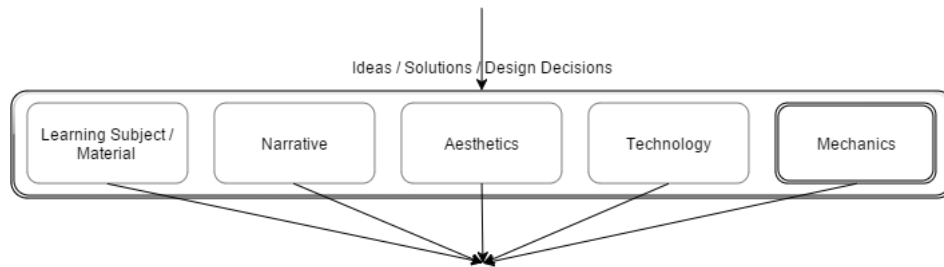

FIGURE 4.15: What can the player do? How? Where? How much? How long?

When designing mechanics it's important to always consider the following, you can be as complex or as close to reality as you deem needed, however, start simple and evolve into more complexity. To better do this it is often useful to consider what skills a player needs to have before playing the game and what skills the player is expected to have by the end of playing, sometimes you can even go further and create this list for each in-game level.

As a designer you should always differentiate between real and virtual skills, what does the player need to be able to start playing, what does he need to learn in-game to proceed? And if the player does not possess one, can he compensate with another? Finding the mix between these two types of skill will ease the player into the flow state [12].

Rules and constraints are a limitation that can be set by the medium or by the designer, whichever the case the player needs to be able to discover and understand naturally, even something simple as "I can't fly in this game, so there must be another way to climb that obstacle" is worthwhile if the player reaches that conclusion by himself.

For this to happen a game needs good goals [38] and they should be:

1. Concrete

2. Achievable

3. Rewarding

The flow state has been mentioned before and here it makes the most sense, the flow state is the greatest asset for a game to produce and can be defined as, a feeling of complete and energized focus in an activity with a high level of enjoyment and fulfilment [12].

The components needed to achieve the flow state are as follows:

1. Clear goals

2. No distractions

3. Direct Feedback

4. Continuously Challenging

Already the parallel between Mihalyi's flow state and Schell's good game goals are apparent, the basic idea is to design mechanics and situations where the player feels challenged but not overwhelmed, where his current skills allow him to learn new ones he can then put to use to solve the current task. In essence try to design in flow.

Put into practice one should start at the most basic and move forward from there as such here provided is a simple list of questions for the designer to answer when thinking of game mechanics:

1. Who is the player?

2. Where is the player?

3. Will the player move? How?

4. Is he supposed to explore? If so how?

5. Can the player interact with what he finds? How?

6. What is the player supposed to learn?

7. Can he collect items? If so does he use them? How?

8. What is the most basic challenge we can throw at the player?

9. How does he overcome it?

There is a reasoning behind the order of each question, and it is as follows, by establishing who the player is within the game one can already infer which limitations should be set. For example, in *Gestual Life* the player plays a six year old deaf child, this establishes limitations and context for further decisions. If it is a child the player character needs to be shorter than the "adult" characters, if the character is deaf then the player needs another means of receiving the information he might have received by hearing. Next by defining where the game takes place or what space the player is in further refines the idea and the limitations. In this game the world is very much like our own so we as the player are limited by physics, but if you are making a game about sea life in which the player explores the ocean floor then the next question is a pivotal point in which you will either make or break immersion. If the player is simply able to "walk" underwater then the player's suspension of disbelief will be shattered, you will need to think of solutions that best serve the type of knowledge you wish to convey, if it's about simply exploring and learning about wildlife and it's dangers then a simple "swimming" movement might suffice, however if it's about piloting a submarine to explore deep trenches then a more simulation oriented movement is needed. Answer each of the questions in order or out, the order is there to help but is by no means obligatory, just build upon each answer with the next question and you should be able to collect all the building blocks to start and enhance your game design.

The word "simulation" mentioned above brings to mind one very important point, games can be a great practice for the real world, in them it is possible

to break down and simplify models and then consume and understand them one after the other until the more complex real ones can be reached [38][25]. Like it was suggested in the "Serious Entertainment" column in Table 3.1, even when simulating a real world scenario, don't smother the player with information, he will feel overwhelmed and quit playing. This isn't done in traditional teaching methods (first children are taught to sum, not to solve differential equations) so there is no reason to do it in games, start out simple and keep challenging the player until it reaches a level of realism that emulates closely or perfectly the real world scenario you are trying to convey, as stated by Csikzentmihaly *"... it is difficult to ignore challenges once one is aware that they exist."*. This might be the most important lesson so to reinforce the idea, try to design in flow, challenge the player to push further, not to hit a wall until it crumbles.

As stated above it is best to start out simple and move on to more complex scenarios as such, *emergent gameplay*. Emergent gameplay are scenarios which happen during gameplay that haven't been programmed or designed by the developers, also known as *Dynamics* [22]. This is an incredibly powerful event, a game having this occurrence usually is a resounding and newsworthy success [33]. *Eve Online* is one such story where the designers implemented movement mechanics, shooting mechanics and player interaction mechanics, this combination along with human behaviour brought on a story in and of itself in which developers played no part, this is good for the game and the players who feel ownership of it's events and who invest themselves further into this game who they feel is theirs.

Emergent gameplay occurs through the combination of mechanics and player behaviour, here are five tips in order to facilitate the appearance of emergent gameplay [38].

1. More operative actions (Walking, jumping...)

2. Actions that can act on many objects

3. goals that can be achieved in more than one way (Going in through the fron door or sneaking around the back?)

4. Many objects

5. Side effects that change constraints

If done correctly these will create *Dynamics*, which in turn the combination of will produce *Aesthetics* (in the context of play) which are the reasons people play and invest themselves in games [22]. If the player feels that his or her way of playing is unique to him and that he is outsmarting the developer he will invest himself further and further learning of his own accord and not because the designer told him he had to.

Although the mechanics haven't changed in *Gestual Life*, the way they are presented has. Since this is an increment less time is spent teaching basic movement and more time teaching Portuguese Sign Language. Players by this point have gone through the first part of the game in which these mechanics are layered onto one another slowly starting by the basic movement controls, introducing video collection and finally moving on to solving challenges.

The hardest part of designing mechanics is forgetting they are lines of code or actions and remembering who they are for, design to challenge your player, to facilitate his learning but keeping him always on edge and focused wondering, what is behind the next door and can I beat it?

4.2.2.7 Risk Assessment

Based on the answers acquired from each question in the preceding steps and applying them to "*Gestual Life*'s original version retroactively as an example, the possible outcomes for each component would be something like this:

- **Learning Subject:** Portuguese sign language and the hardships of being deaf.

- **Narrative:** You are a six year old boy who is deaf and needs to do his day-to-day tasks while coping with this disability.

- **Aesthetics:** Cartoonish look with cel shaded graphics.

- **Technology:** 3D movement and models, gesture recognition.

- **Mechanics:** 3D movement in the first person perspective, item and world interactions, challenges in the form of questions and conversations.

Even after all the care put into each answer and each component there are still risks, this step (Figure 4.16) is simple yet one of the most important, before beginning work on modelling, level designing or programming, create a list of risks, or reasons why each component might not work.
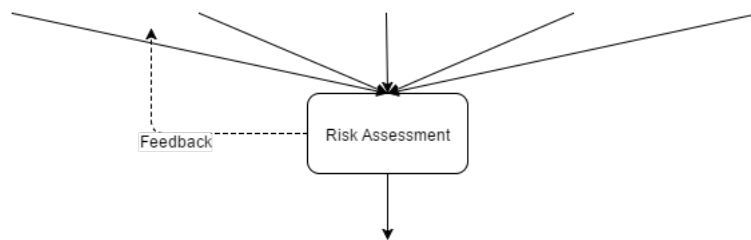

FIGURE 4.16: After getting each answer, what are the risks to using them?

Using the above example the risk list achieved might look something like this:

- **Learning Subject**

  Risk 1: The player might not find the subject interesting.

  Risk 2: Too much information and freedom might lead the player to overlooking important sections of the game.

  Risk 3: The gameplay might not be enough to transfer all of the knowledge.

- **Narrative**

  Risk 1: The story might be boring or not novel enough for the player.

  Risk 2: Player motivations might not coincide with the story.

Risk 3: None of us are writers.

- **Aesthetics:**

    Risk 1: Players might think the game too childish because of the aesthetic.

    Risk 2: Modeling characters designed for empathy while not being human is hard and can fall in the uncanny valley.

    Risk 3: Without animation characters aren't emotional enough.

- **Technology:**

    Risk 1: Gesture recognition requires new knowledge and specialized equipment(Kinect).

    Risk 2: Motion controls don't provide physical feedback therefore the player might not be able to complete high precision tasks.

    Risk 3: 3D movement although common in games might be something that non-gamers can struggle to use.

- **Mechanics:**

    Risk 1: World exploration might be too hard for non-gamers limiting the amount of investment they put into the game.

    Risk 2: If the challenges are too easy the player will be bored.

    Risk 3: Collecting and gathering videos might not be as obvious as we think.

Take as long as you need and try to discuss which risks are of higher priority than others because once this list is complete, the next step is prototyping.

4.2.2.8 Throw away prototype/s

At this point there is no game (assuming this is the first loop), only ideas, those ideas carry risk so, in order to mitigate those risks the team is going to test for them. Some might be more complicated than others but all are possible to prototype and here (Figure 4.17) it is going to be explained how.
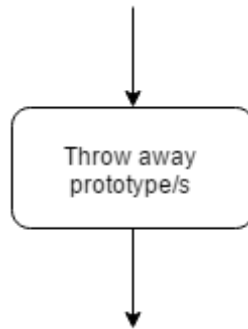

FIGURE 4.17: Quick and dirty

All that matters is that the prototype answers the questions posed by the risk assessment, as such building the prototypes as simple as possible will filter out any unnecessary parts and/or improve others, sometimes one solution may answer another risk.

Taking a look at the above list built for *Gestual Life* one could approach prototyping as such:

- **Learning Subject**

    Risk 1: The player might not find the subject interesting.

    Risk 2: Too much information and freedom might lead the player to over-looking important sections of the game.

    Risk 3: The gameplay might not be enough to transfer all of the knowledge.

    Prototype: Since we can't avoid what to put in the game, instead of "what" knowledge we can prototype "how" to present it. A quick mock-up

of the challenges interface as well as the inventory screen presented to people outside of the team might be enough to gauge how much interest there would be in interacting with the game. Since the interface is part of the experience and making it "feel good" to use is a way to circumvent the problem of the task itself and draw interest into other parts of the game [38]. A game can't be seen as a teacher it's far more valuable as a tool to enhance learning, present the idea of knowledge transfer to a professor versed in the subject to obtain feedback, make sure to analyse said feedback, again "what" isn't important, that can come from a text book, what is important is the "how" think about the experience of learning and adapt it into these answers. In the original *Gestual Life* we didn't think about what to teach, we thought about "why" you would want to learn and "how", adapted it into a core experience and worked to enhance it and so far it worked [2].

- **Narrative**

  Risk 1: The story might be boring or not novel enough for the player.

  Risk 2: Player motivations might not coincide with the story.

  Risk 3: None of us are writers.

  Prototype: This is one of the easy ones, just write a simple draft of your story, a small statement that summarises the plot and present it to someone for feedback (again outside of the team is preferable). If you can't tell your whole premise on a single statement more work is needed on the narrative as it's still just a conglomerate of ideas and not a cohesive whole.

- **Aesthetics:**

  Risk 1: Players might think the game too childish because of the aesthetic.

Risk 2: Modelling characters designed for empathy while not being human is hard and can fall in the uncanny valley.

Risk 3: Without animation characters aren't emotional enough.

Prototype: As said before in the "Aesthetics" step (Sub-section 4.2.2.4), concept art is a type of prototype, use it as such. Draw a rough sketch of your characters, world, interfaces, as simple as you can, as rough as you can. Then present it to other people to gather feedback.

Since the style of choice was cartoon and cel shading LYS was drafted roughly as seen below (Figure 4.18), in order to establish her as a central build piece of the aesthetic design, you can see she lost some elements from this iteration into her final form, but, her base form and main lines are unchanged.
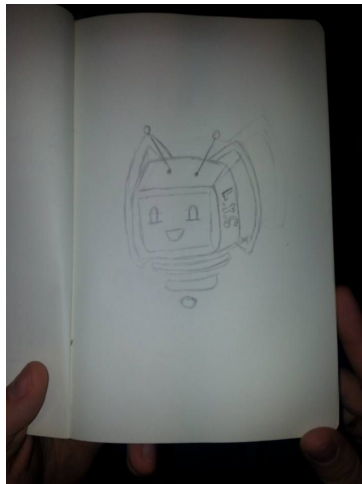

FIGURE 4.18: LYS's very first draft

- **Technology:**

Risk 1: Gesture recognition requires new knowledge and specialized equipment(Kinect).

Risk 2: Motion controls don't provide physical feedback therefore the player might not be able to complete high precision tasks.

Risk 3: 3D movement although common in games might be something that non-gamers can struggle to use.

Prototype: This is one of the hard ones, not because of building it but because of how deep you need to go or how much is enough. Build a rough prototype of the decided mechanics as fast as possible, in our game gesture recognition and 3D movement.

It's necessary to note that the engine used (Unity3D) was new to us at the time of development of the original *Gestual Life*, as such the time we spent on learning to use the technology and prototyping the movement left us with one conclusion, we needed to abandon gesture recognition in the first build due to time constraints.

It can hurt to abandon ideas and technology you are excited about, this will be addressed in this section, for now it's enough to be able to abandon something if it either doesn't work or, doesn't work on *time*.

This risk was highly mitigated during the development of this new version not only because there had already been tests and development time put into it, but also because in this new instalment usage of the Unity3D engine was already familiar. The risks still exist and are still accounted for, however prototyping is faster and there is a bigger precision due to past experience.

- **Mechanics:**

Risk 1: World exploration might be too hard for non-gamers limiting the amount of investment they put into the game.

Risk 2: If the challenges are too easy the player will be bored.

Risk 3: Collecting and gathering videos might not be as obvious as we think.

Prototype: Have your programmer build a rough demo of the mechanics, even if it is not in the desired tech (3D space in our case), do it in a simpler tech, if all of your programming team is working on the technology prototype already, do it on paper.

Most games can be abstracted into paper [38], for example let's prototype Super Mario Bros, using a sheet of graph paper and another of regular paper. Cut out two shapes, a square which you paint red to represent "Mario"(player) and a triangle which you paint brown to represent a "Goomba"(enemy NPC). You can cut out other shapes to represent obstacles, but after this is done place them all on the graph paper, now using a pendulum strike a time, each time the pendulum ticks you move Mario's piece and the Goomba piece. The objective here is to test the mechanics using time as a measure. In each moment in time where are you? Are there enemies close after a few moves? Can your piece jump? How many ticks until he's at max height? Did you account for the height of objects? Are jumps too high and you take too long to come down?

This isn't the case in all games, it isn't even the most common, but if needed you have to be able to abstract a gameplay mechanic to the furthest possible degree.

In the original *Gestual Life* we did our prototyping by building a flat plane with a character and an object on top. First the player had to navigate to the character, interact, then move to the object and interact, finally go back to the character and solve a challenge with the new knowledge gained from the object. We didn't build this all in one go, but we did it in this order, movement, interaction and changing gameplay modes from movement into challenge.

The new version had a similar prototype. However since base mechanics were already established this prototype had other objectives. A fully built three dimensional map complete with buildings where the idea was to judge movement speed, distances and interactions was the prototype. Here building placement, trigger activation (areas where, if the player is present something happens in the world or reacts) and character interaction were the goals being tested for.

Prototyping shouldn't be taxing on time or on resources, but it can be very taxing on team morale and your own emotions. Having to abandon ideas you held dearly can be painful, but remember, in the end what matters is that the game you are developing is a memorable experience and is so because the team had maturity to pick and abandon ideas correctly.

Since prototyping can be hard here is a list of tips useful when prototyping [38]:

1. **Answer a question:**

   Prototyping serves a clear purpose, finding out if solutions actually do what they are supposed to, if you can't answer at least one of the risks then redesign your prototype.

2. **Forget quality:**

   Again it is stressed, do it quick and dirty, use as little resources as possible, just design what is really important. As seen in the framework, there are loops, they are inevitable, so try to be faster and not a perfectionist, the sooner you answer your questions the better.

3. **Don't get attached:**

   Every designer has to face the prospect of throwing away ideas, sometimes whole game concepts, it is called throw-away prototyping after all, if you regard ideas as tools to further a design instead of being the design itself it will become easier.

4. **Prioritize prototypes:**

   When building a list of risks you might see that several prototypes are needed, as such prioritize them and as said before, one might even answer the questions of another.

5. **Parallelize prototypes:**

   As stated above your whole team doesn't need to work on the same prototype, have them each work on a different set of risks to be efficient.

6. **It doesn't have to be digital:**

   Paper, plastic, cubes, cylinders, whatever you can use to prototype use it, don't become glued to the screen, every game can be abstracted to cubes and most are built that way at the start (*grey boxing*), just get those prototypes done fast and efficiently.

7. **Pick a "Fast Loop" Game Engine:**

   Unity3D was used both in the original and current version of *Gestual Life* which allows developers to build content, run it, navigate to said content and test it. If you have to code everything, wait for the compile or navigate through a complicated section before you can test, you're going to waste a lot of precious time. Decide early on an engine that lets you run new parts of your game fast and with minimal effort.

8. **Build the toy first:**

   First build the novelty in your game, the world, the space, the mechanics and then proceed into content development, in the original and current game all of the base mechanics were built before a single challenge or set piece was put in, because having options open when the developers first entered their own world let them be inspired and have a clear view of how it would "feel" to play it.

Prototyping has been used in all of game development however being accurate and precise is still something game designers struggle with, especially when it comes to abandoning ideas. Prototypes are special, although thrown away, they are why the game grows and becomes memorable, it all starts with them, quick and dirty.

4.2.2.9   Evaluate through the 8 filters

After finishing the previous steps there is one more evaluation needed, while designing a game there are constraints that must be met (Figure 4.19). A game has

goals but it has needs as well that it must fulfil, here a set of eight filters by Schell [38] will be presented. These filters serve once more as questions that need answers in order to progress, however negative answers don't necessarily mean going back to the drawing board.
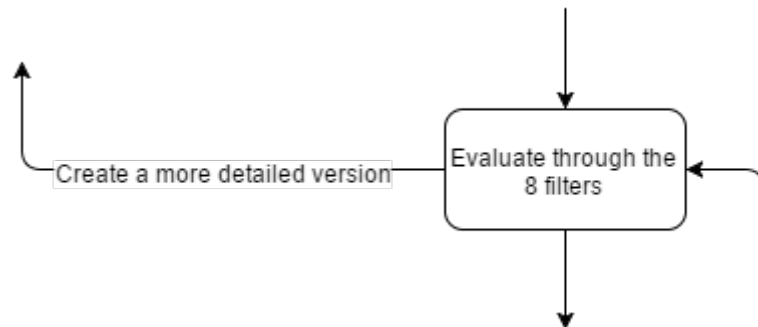

FIGURE 4.19: During design there are constraints

If there is an answer the designer isn't satisfied with then the game needs more detail.

1. **Artistic Impulse**

   A designer has a notion of the final product without ever seeing it, every effort and step along the process until that vision comes true is an artistic impulse, simply ask yourself:

   Does this game *feel* right?

2. **Demographics**

   The game will have a target audience, this sets constraints in both content and it's intensity, a target group doesn't necessarily mean an age group, it can be people who like a certain sport or art form. The idea is to see the game from their perspective and then ask:

   Will the audience like this game enough?

3. **Experience Design**

As stated before a game is above all else an experience, in order to know if a game concept is ready for development compare it to the experience intended to be replicated and ask:

Is this a well-designed game?

4. **Innovation**

Building a game entails it being a new invention, no two games are alike and each game builds on the foundation others left. It's important to push the medium forward and innovate both on gameplay and concept, ask:

Is this game novel enough?

5. **Business and Marketing**

Although serious games have an inherent need to them as teaching tools, they haven't had success in the industry as previously established in Chapter 1. Even if the game is being developed and will be put to use as a tool in a school therefore gathering an audience, think about it as though there wouldn't be that insurance, think about the students and ask:

Will this game sell?

6. **Engineering**

Having ideas or dreams about the next big gameplay innovation or graphics advancement isn't enough. The previous prototypes are intended to answer questions like these, when condensing all of the decisions made until this point ask:

Is it technically possible to build this game?

7. **Social / Community**

Each game will create a community around it whether small (*Darkest Dungeon*) or large (*World of Warcraft*), even classrooms can be seen as the community of a serious game, in these communities there are social goals, in *Gestual Life* the goal was getting players to see the reality of being deaf, so the question was:

Does this game meet our social and community goals?

8. **Playtesters**

Before launching a game a playable-prototype will be developed, at this point
if development is still in the first loop, playtesters will have been few, but
judging from their reaction and from the reaction of playtesters who will test
the demo, ask:

Do playtesters like the game enough?

These eight filters serve as a gateway between development steps, each new
iteration of the game must answer all of these questions and better. There is no
magical number pertaining as to how many loops should be made but, the first
filter might just give the designer the answer he needs. If the game feels whole
and cohesive it is time to advance into development.

4.2.2.10   Develop and enhance the design further

At this point all that has been achieved is planning, however all of this preparation
will reduce development time and mitigate the risks of having to scrap the project
midway through. Now is the time to start coding, modelling, scripting and level
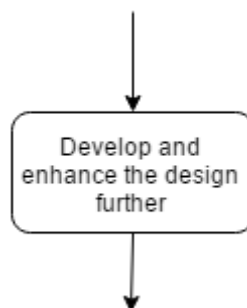designing (Figure 4.20).

FIGURE 4.20: Push each answer forward

Start implementing each answer, a designer's greatest tool is communication,
a designer needs to be able to talk to everyone on the team but more importantly

he needs to keep the team talking between themselves. Artists and engineers don't speak the same technical language, an artist might not comprehend why his thirty gothic 3D modeled realistically rendered buildings can't be on screen each frame and the engineer might not comprehend why the artist would request such a task in the first place.

Each time a new increment is created, build simple demo, test it against the filters, observe each facet and think about how best to continue from that point forward.

At this point in development the most arduous task will be team management, keep communication flowing and listen closely to everyone. Maintaining a good team environment and productivity will be further explored in Section 4.2.5.

4.2.2.11 Playable Prototype

As stated before in Sub-sub-section 4.2.2.9, when developing the game each new milestone needs to be tested against the filters, otherwise a crucial mistake or flaw might be overlooked too late and resources will have already been spent on it.
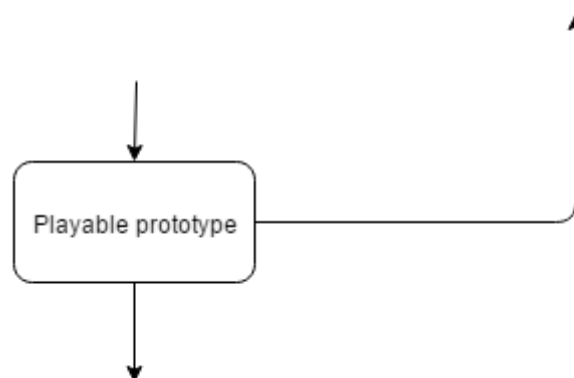

FIGURE 4.21: The first time the game can be seen running

Figure 4.21 refers to a prototype that is more than a throw-away, at this point a functional prototype of the game has been developed. This prototype can range from a simple grey-boxed demo, where all of the assets are simple cubes or shapes without textures, to a full deployment ready build of the game. The only thing

that defines either one or the other are the results from testing which will be addressed in Sub-Section 4.2.3.

Once a playable form of the game is available (even if grey-boxing) compare it to the filters, if it passes keep developing the design further and further, at some point the playable prototype will need to be in a final state or a "play-test" ready state.

There isn't a recipe for which number of iterations a game needs in order to be ready for testing or launching, however budget and time constraints will inevitably do this on their own. As soon as all the mechanics, aesthetics and level design are in place, enter the play-testing segment of development, in this the game will be tested with focus groups, people outside and inside the gaming industry.

This prototype isn't supposed to be of the "throw-away" type however it is always important to be ready for letting go of a whole section if needed, as long as it improves the final game.

Playable prototypes have various states, this should be incremental, coming from a simple grey-boxing demo into the final game format, eventually it will be ready for launch.

4.2.3   Testing

The testing phase (Figure 4.22) is one of the most taxing mentally for the team. In this section which concerns and information are need for each type of test will be covered.
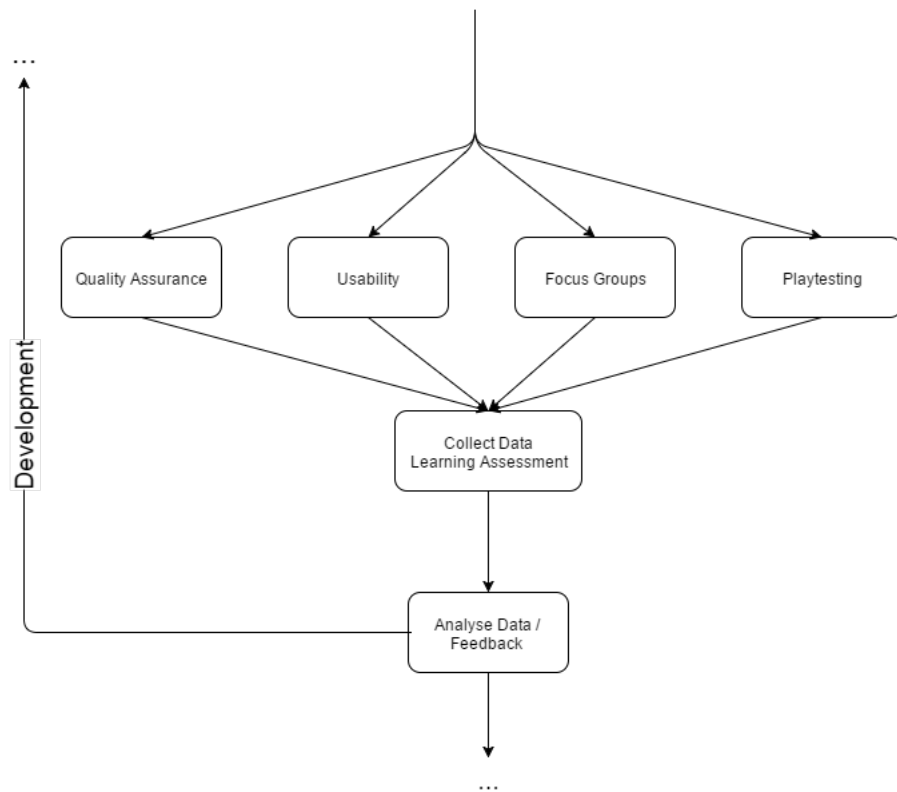
FIGURE 4.22: Testing fully expanded from the World Map

4.2.3.1 Quality Assurance

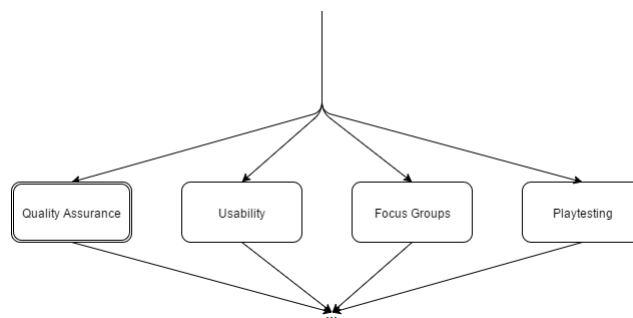The only purpose of this test (Figure 4.23)is to ensure that the game works.

FIGURE 4.23: Does the game work?

During this test the only concern is that the game does not crash, have any game breaking bugs or other flaws.

These types of tests are generally done by technical engineers but in the absence of one they will have to be conducted by the remaining team. When testing try

to focus more on areas of the game that generated problems during development since those are more likely to have errors or overlooked details.

Quality Assurance is extremely important but isn't the hardest of tests, if it crashes or performs badly it needs fixing.

4.2.3.2 Usability

Usability tests' (Figure 4.24) purpose is to ensure that the systems, UI and menus are intuitive and easy to use.
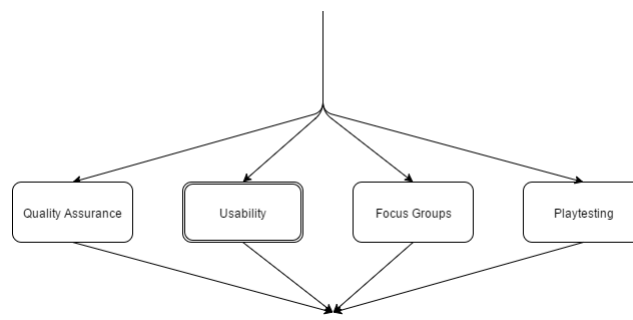

FIGURE 4.24: Are the systems intuitive enough?

While not always directly related to which experience the game wants to transmit they do influence the overall feel of the game. Consider that some games are fully an interface and are critically acclaimed [37][28].

Consider hiring an Usability Expert for this section[38] if no one on the team is comfortable with the task, since a usable, intuitive and fun or rewarding UI will contribute to the overall enticement of the game.

The current demo of *Gestual Life* was extensively tested in regards to usability and this led to a number of changes, even though the game had the correct experience, non-gamer users were having difficulty navigating the world and some context sensitive actions.

These changes will be further explained in Chapter 5. Each change was completed after analysing user feedback and gauging what the core problem was not just the outlying.

4.2.3.3 Focus Groups

This step (Figure 4.25) refers to Focus Groups which are constituted by potential players interviewed about what they would like to or not to see, in a game.
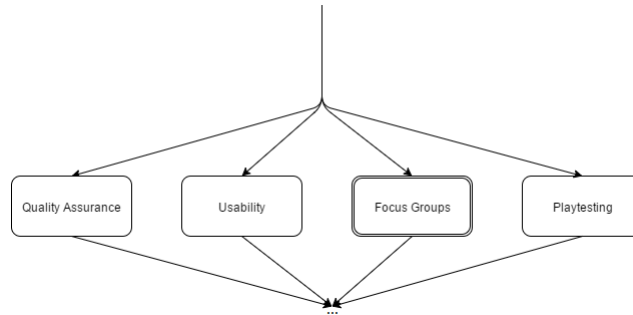

FIGURE 4.25: Will potential players like this game?

This type of testing is very useful for prioritizing ideas that have been detailed and chosen to be in the final product.

Due to their nature these types of tests can be done out of order, it might be more advantageous to do focus group testing right after deciding the Mechanics, Aesthetics, Technology, Learning Subject and Narrative.

In the original *Gestual Life* the authors interviewed both the General Secretary and Students from APSurdos to gauge interest in the possible development of a portuguese sign language video-game.

4.2.3.4 Playtesting

Playtests (Figure 4.26) are the most important to a designer, the main objective is to assess if players are getting the intended experience from the game.
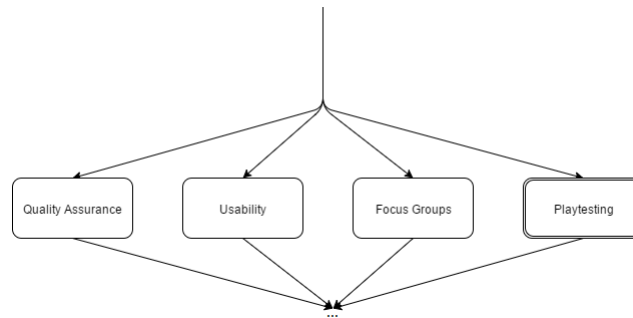
FIGURE 4.26: Is the game conveying the correct experience?

This often comes with big personal issues to designers, as this is basically an invitation for testers to say why the game is bad. In order to have a successful playtesting session the designer needs to know beforehand what it is exactly that he is testing for. What questions need answering?

When setting up the testing session there needs to be a clear "why" as to why the test is being done, the place is also important, should it be a neutral spot where neither the designer nor the test subject are at ease or should it be done in the office where the designer has more control but the subject isn't comfortable with his surroundings?

These questions must be answered in a game by game basis, each game has different needs and each testing phase has different requirements. When testing a certain section of a game the following needs to be considered, how will my player know what to do? Does the section being tested have a tutorial? If not a suggestion would be using the *"Greeter"* as a means to introduce the player to the game and it's controls as well as to what task the player will be performing.

When engaging any testing phase a clear goal has to be set, this goal needs to be composed by:

1. Who should be tested? Should the target audience be considered only or even people outside of it?

2. Where will the test be held?

3. What are we looking for in the results? Should we look at what was done or also at what wasn't?

These questions should be answered by a combination of both the pre-emptive preparation and the test results, it is important to be ready to adapt on site to the way the tests are working.

Depending on the experience observe not only the screen but the player also, in both iterations of *Gestual Life* both were observed and it was found that often times the player would mimic what was being presented on screen, while he was playing [2].

This way of testing let us understand through player body language and expression what was working and what was not as well as which sections of the game were complete or near to. A game needs a player so testing both or at least observing both is a good practice.

In the original *Gestual Life* the authors tested with subjects from the Associação Portuguesa de Surdos (APSurdos), however for the demo in this dissertation, the general public as well as game developers from a portuguese game development company, were the chosen audience. As expected feedback from players and developers is very different, a developer needs to be able to filter useful information from simple opinions or misleading interpretations.

These tests are the ones that will probably offer the most in terms of modifications and corrections to the project.

4.2.3.5   Collect Data and Learning Assessment

Learning Assessment is done in order to verify that your player has in fact gained knowledge while playing the game. In this step (Figure 4.27) collect data through various means, don't stay glued in the digital world created, sometimes player behaviour or expressions may tell a different story than gameplay.
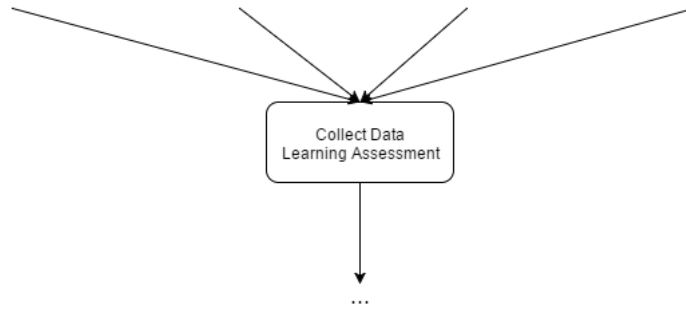
FIGURE 4.27: What did testing yield?

There are several ways to assess how effective a serious game is [4]. Some of these are shared with data collection, the simplest way to understand that the player has in fact learned is to have him do a simple written survey or test. Simply ask the player to repeat a particular piece of important knowledge and gauge how difficult it is for him/her to answer.

In the original *Gestual Life* we asked players to repeat one of the very first gestures we taught them, we took care in choosing which one, we wanted a complex but not too difficult gesture as such we chose the portuguese gesture for "dresser" and almost all of the players were capable of reproducing it [2].

This method of learning assessment and data collection is called an interview, rather than having the subject write the answers we interviewed each one after playing so as to gauge not only the data but also their opinion, although no two people are the same most of the subjects agreed that the game was well liked and we could see that it resonated with people.

This same method of data collection was applied in the new increment, the results will be shown and discussed in Chapter 5.

There are other types of data collections mainly in the metrics spectrum but these are reserved for the launch step.

4.2.3.6   Analyse Data - Feedback

After collecting data from testing it needs to be evaluated (Figure 4.28). Often times simply reading into the data at face value can lead into misleading results. Most feedback is based on personal biases as is most player's ways of playing the game.
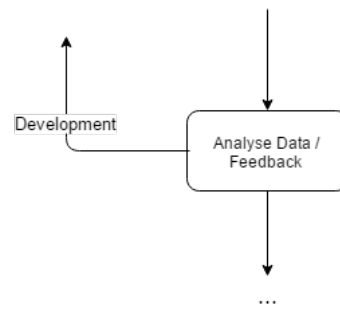

FIGURE 4.28: Interpret the data collected.

What is important in this step is to first find the common themes in the data, for example if a lot of players, don't report it but, are observed having difficulty finding a door or interacting with it you can infer that either the positioning of the trigger isn't correct or that the visibility towards that objective is too low. Likewise when players do report something it almost always won't be in the same words, as designers trying to create experiences the main skill needed is to be able to listen and understand people. If players are reporting that the game isn't "intuitive" for instance, it can be a number of things, controls, menus, popups, level layout or something else entirely, look at the game from their perspective or try to find the one player that says it different than the rest.

Although this is the last step before launching or releasing the game, the truth is that no game will ever be complete, there will always be one more idea, one more test, one more feature, however here is the final gate, analyse the feedback and decide whether or not the game is fit to release or you should go back to development and improve the design further.

In this increment of *Gestual Life*, the common theme found was simple "movement and distance". The game is set from the perspective of a six-year-old child

as such everything seems big, players however don't care. From their feedback they disliked seeing everything big and feeling small, the solution was to increase the distance at which you can interact, they had no problems any more.

Although users complained about the character size what they disliked was having to "look up" to each character by increasing the distance players can look forward and still get all of the information.

4.2.4 Launch

The tests have been positive, the feedback deemed the game a good learning experience, time to release it for the players (Figure 4.29).
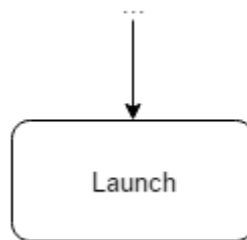

FIGURE 4.29: Release the game.

Just as stated in the previous step (Sub-sub-Section 4.2.3.6) a game is never truly complete, now that it has been release there is a certain responsibility, patches, fixes and metrics. After finishing development the largest test is beginning and the whole player base are the subjects. Collect data from players, are they playing as expected? Are they skipping over content? If so, why? Have they found a strategy that works better than the one designers envisioned? If they use it are they undermining the learning experience or enhancing it?

All of these questions are important, their priority however, can only be gauged in each game as a unique case. Developing patches for a game is a double edged sword, on one hand it shows commitment to the game the team built, but it also consumes resources. It also shows that you failed and did not notice or correct said flaw during development, but again the fact that you noticed it and spent

resources fixing it will get the players to be happy in your teams ability to manage the community.

If important content was cut or is missing at the time of launch the team will either have to be split or enlarged, there needs to be attention to the community and feedback as well as development on the missing content for future release.

Launching a game is the beginning of it's life cycle, accompanying it is a responsibility as a developer and game designer.

4.2.5   Team

This *"sidepath"* (Figure 4.30) of the framework runs parallel to the main process, for the simple reason that, at every step decisions, team environment and project state should run through this path to weed out any problems that might or are arising.
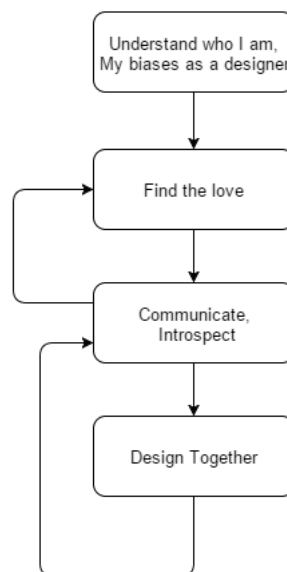

FIGURE 4.30: People build games.

4.2.5.1   Understanding who I am, my bias as a designer

Each and every human has biases (Figure 4.31). The collective sum of our experiences is called "life" and during this life each see and memorize certain patterns.
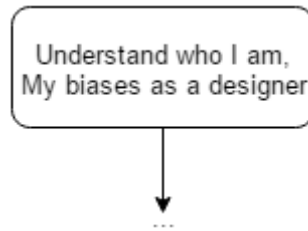
FIGURE 4.31: What do we take for granted?

Some of these patterns people like and some they don't, they are quick to use and reference patterns that are enjoyable and quicker to dismiss those that aren't. This is true in game design, and there is another added problem, familiarity.

If the designer ever finds himself starting a game and after he gains control he instantly has his fingers on the correct buttons be it on a keyboard or a controller that is a bias. By identifying what type of game he is playing the designer knows that those sets of buttons are usually used to control the game. Games that try to change this formula are often cited as having a bad control scheme, this can be true or it can simply be another bias from the reviewer.

There are other concerns though, if the designer isn't conscious of his bias then he might implement something that to him makes perfect sense and is gibberish to the rest of the world, worse than that is a designer who doesn't understand why the world sees it that way.

Before touching any keyboard, before any meeting a designer needs to be aware of his likes, dislikes and of as many patterns he hinges on as possible so as to not let these interfere with the development process if they don't make sense.

4.2.5.2   Find the love

This rather tentative title has a rather simple explanation, members of the team need to love the project, if they don't love the project they need to love the audience. A project without love for either is doomed from the start, if it isn't apparent, find it (Figure 4.32).
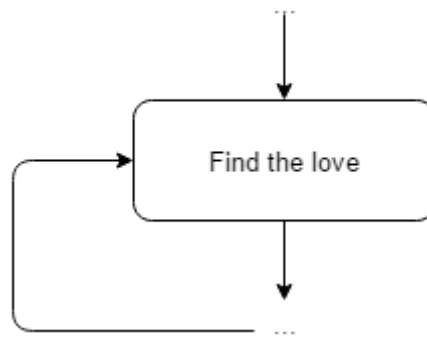
FIGURE 4.32: The main ingredient in every memorable game.

If someone on the team can't love neither, get them of the team [38], it is never enough to remind that a game is an experience and every element that touches a game influences it.

In the original *Gestual Life*, the team didn't start by loving the project, we loved the audience, deaf people, the thought of helping and understanding them led to decisions like the absence of regular sound from the game, the vulnerable perspective of a child the concessions in gameplay for approachability. We kept coming back to discussions about what we liked in the project and each time we did something new popped up, soon we all loved the project and it became a better experience for it [2].

Although not everyone worked on this new project, this step was always present when working both on the increment and the dissertation. When searching for something to love in both the answers were different but equally important and one answer in particular was common, I love this medium.

The team shouldn't force themselves to love the game, but they need to love something, even the idea, for unity within development everyone needs to agree on the vision and not be in love with different visions of the same game or a completely different game altogether.

4.2.5.3   Communicate / Introspect

Finding reasons to love a project isn't enough, people change and with them their likes, dislikes and motivations. Just as during the development keep communication lines established and open, let people on the team talk to you and each other whenever there is a problem. Lastly even when there is no need to communicate, think about yourself (Figure 4.33).
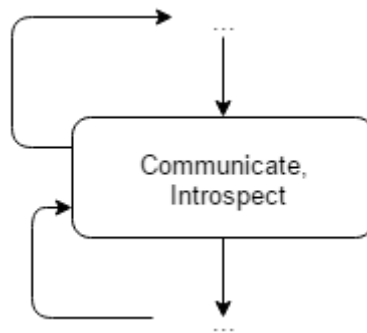

FIGURE 4.33: Keep the team together and motivated.

As development moves further and further team members get tired, some start to succumb to the pressure of delivering a good product. Just like stated in Sub-sub-section 4.2.2.10, not everyone on the team will speak the same kind of technical language. This may lead to misunderstandings and tension, everyone has a different reason for working on the project try to really understand them and put them all on the same track.

Even when working alone, a designer might lose track of the end goal, that is why it is important to take a few moments to think about ones' self.

This might seem something that doesn't contribute directly to game development but it does, and a lot. Designing a resonating experience means that a lot of emotions will be put into the game, if those emotions aren't properly taken care of the player might feel the team's resentment in certain sections of the game, this can shatter the player's immersion or his will to play altogether.

4.2.5.4 Design together

Every person that touches a game influences it's creation. Artists, engineers, designers even marketers have a vivid image of the final product, as such, include them on design meetings (Figure 4.34).
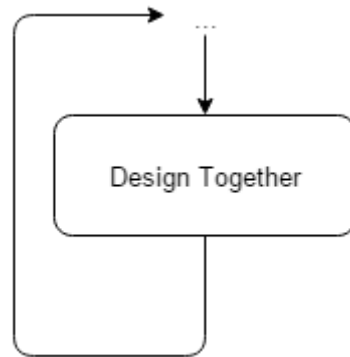

FIGURE 4.34: Everyone is part of the project.

The final decision must always come down to the lead designer or team leader, however, don't exclude any input without seriously considering it first. People like to have ownership of what they create, even if it is just to point and say "I did that", as such learning to listen and communicate is the most important skill a designer can have. Every team member should be able to express an idea and contribute to development, that will motivate them, make them feel like their efforts are rewarding and renew their trust in the leadership [38].

Each type of contribution has a completely different vocabulary (artists, engineers, musicians), as such a different "feel" to it as well. When well applied this makes a game feel diverse and often appealing to a wider range of audiences, since it comes from a wider range of sources as well.

Include as many people in the design process as possible, even management if able, this helps placate fears and also gives room to experimentation and new angles that can be explored outside the designer's own biases.

4.2.6 Documentation

Documentation (Figure 4.35)has two functions, memory and communication [38]. There are several documents that go into the development of a game, each has a different audience but all share the aforementioned objectives.
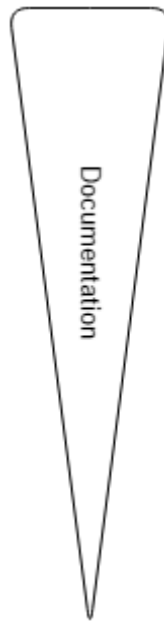

FIGURE 4.35: Communication and record keeping are it's use.

During development a great number of decisions will be made and these in turn need to be recorded, if decisions, choices or details aren't written down the team can find themselves discussing the same features time and time again. To avoid wasting resources keeping an up to date document with the most recent changes to design will facilitate a number of things, one of which is communication.

As mentioned previously a team is composed of many talents, not all of them speak the same language or know the same terms, likewise documentation will have different meanings to different people in the team[36] [38].

Being able to state clearly on paper what it is the game needs will facilitate task assignment, removing the need for team members to constantly feel lost asking on what it is that they should be doing [36].

There are several different types of documents, from Art Bibles [43] that belong to the artists, to the simple and adored Game Manual [43] [14] [38] made for the player. Since each mean different things to different people, in this section the focus will be on the core document shared by every member on the team, the Game Design Document(GDD).

The GDD is a living document, this means it is constantly changing and updating as the game develops [3]. This document's main purpose is to describe the game in it's entirety so as to form a shared vision of the project as a whole that can be shared by all of the team. To put it simply, it provides *context* on what the team is doing and why [36].

Like fingerprints each GDD is unique to the game it describes, when redacting the document always consider the *theme* as stated in Chapter 4 and the *feel* of what the team is trying to achieve, write down the decisions that best support it.

Although unique, a GDD should include these features when describing a game [36]:

1. Introduction

2. Background (optional)

3. Description

4. Key features

5. Genre

6. Platform(s)

7. Concept Art (optional but advised)

The introduction is simple, just describe as best you can in a short (single) sentence what the game is about this has to sell the game to the reader [36][38].

Background is marked as optional since if it is the first or single entry the team is building there are no past entries to reference or take inspiration from [36].

Description contains the *experience* the player will have, it is good practice to write this section as if the reader was the player often referring to them directly in the second person (you). Include all of the elements the player will be able to interact with and engage, this description should be able to convince anyone that the game is worth making and more importantly playing.

Key features are what distinguish this game from the rest, what will set it apart and make it memorable to players such as in *Gestual Life*, LYS's AI and companionship, the idea of playing as a deaf character or the "open world" many games use as a marketing draw [2] [36] [38].

Genre is an attempt to define the game, since the MDA [22] isn't used as standard try to find magazines, awards and other community driven classifications to describe your game, some examples are "First Person Shooter" (FPS, as in Call of Duty), "Role Playing Game" (RPG, as in Final Fantasy), "Platformer" (as in Super Meat Boy) or "Massive Multiplayer Online RPG" (MMORPG, as in World of Warcraft). Attribute a genre to the game using these standards so as to establish further context and biases to both the reader and the project.

Platforms are where the game will be running, the target platform can be a home console, a mobile phone or a desktop among others.

Concept Art, as stated previously in the *Aesthetics* section, concept art gives a clear vision for the game, it motivates both buyers and developers, it sparks imagination. Although optional it is very advisable to include art in the GDD, it sells the idea, establishes and defines context and even brings systems to the discussion if mock-ups are included [38] [36].

As stated this is a living document as such alterations and updates are constant, this is reflected in the shape of the framework component, the conical shape is to suggest that, documentation (GDD included) is heavier and more defined at the start of development and gets less and less accurate or updated the further development moves forward. This trend can be seen in artbooks included with

collector's editions of games where often times elements that appear in-game aren't included in the book.

This nature is also the explanation for the triangular shape of this step. Documentation will be heavy and very detailed earlier on in development, but as the process moves forward less time will be spent on writing documentation and more on implementing and testing decisions as quickly as possible. As such in the later development stages documentation changes or creation is very light if at all.

Developing a game is an ever evolving process full of constant design modifications and corrections, at the end stages of development more effort will be put on bug fixing, content curation and less and less on document writing, however without the GDD the team will simply be lost wasting time and resources.

Chapter 5

Testing and Results

A new increment of *Gestual Life* was designed and developed taking into account the framework described in this dissertation.

This new version features an open map (seen in Figure 5.1) that the player can explore without boundaries or constraints (aside from the natural limits of the map). This map features several NPCs that the player can interact with as well as a slew of objects with which to interact and learn from.

FIGURE 5.1: The open explorable map of *Gestual Life*

Once the player is thrust into the world LYS will be the first to greet him and give him the task to complete. The task is simple, purchase and mail your cousin's medicine.

From here the player has to follow a series of steps in order to finish the main task which is "Mail your cousin his medicine". It is important to state that most of the players either ignored or forgot the test script and went on to explore and play on their own, the results of their performance executing each task can be seen in Figure 5.2.
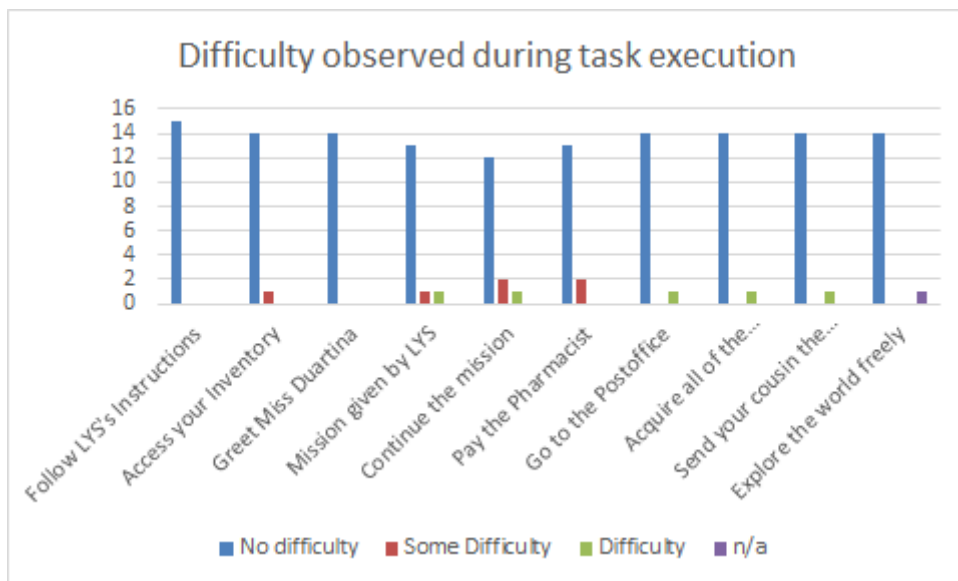

FIGURE 5.2: How well and easily each player completed each task.

Here "Difficulty" is judged by how long a player takes to understand and mechanically complete a task. Subjects that felt difficulty did so for one of two reasons, either the control scheme wasn't something they were familiar with or, they didn't know where to go next.

These issues can be explained by several reasons, types of player and their biases as well as ages, both of these can be seen in Figure 5.3 and Figure 5.4.
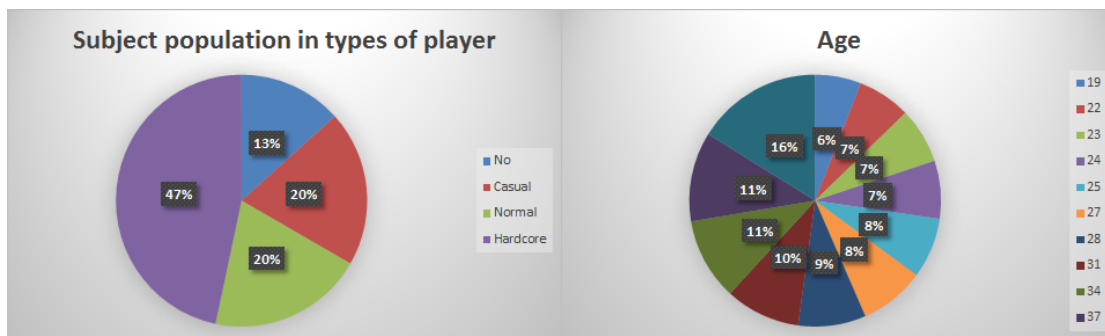

FIGURE 5.3: Types of players.

FIGURE 5.4: Subject's ages.

Some players had difficulty with the controls or interactions because they simply didn't play video games, others because they were accustomed to mobile gaming and lastly *"Hardcore"* players because of their biases. Hardcore players are here defined as persons who play for more than eight hours a week, normal players for about seven hours a week and casual players for about three hours a week.

At any point in time the player can open the inventory (Figure 5.5) and confirm which task needs to be completed as well as consult any videos and items acquired through mousing over. The idea behind making this version a "sandbox" and not a linear level, is that one of the test objectives was to see if the player could identify each objective through it's silhouette or symbol leaving the interpretation of the aesthetics to the player rather than directed by the designer. Some distractions were also placed aside from the main mission to see if players would indeed be distracted or persevere with their task.
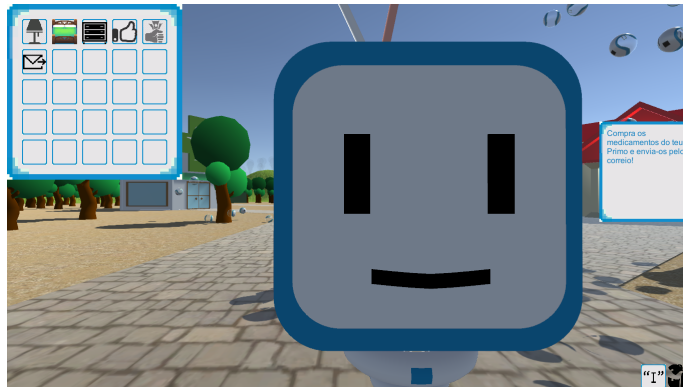

FIGURE 5.5: Inventory screen featuring LYS.

Movement within the game world was achieved using the W,A,S,D or arrow keys, a convention or "bias" in the first-person gaming genre, looking around the world was achieved with the mouse. These controls proved hard for some to master, while normal to hardcore players didn't have any issue with the control scheme, more casual or non-players had problems controlling both the character and direction at the same time.

One thing they did share in common, once the main task was done all of the players were given the choice to stop playing at any time they wished and end the test.

A vast majority of them explored the world further and collected every video while answering every challenge (Figure 5.6).
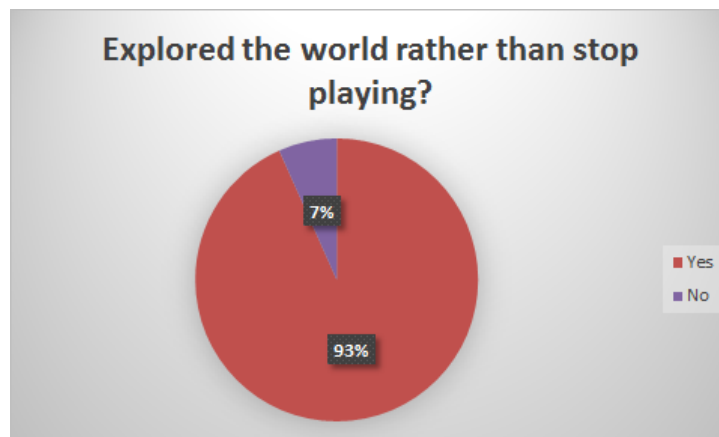

FIGURE 5.6: All of the players explored the world when given the chance to stop playing.

This has all been in terms of game design and the way the level was built so as to guide and motivate the player. However the learning part of this serious game also has interesting results, the very first piece of knowledge the player is shown comes in the form of a greeting by LYS, the Portuguese sign for "Bom dia" (Figure 5.7).

FIGURE 5.7: LYS greets the player as soon as the game begins.

Right after this sequence LYS explains what the player is supposed to do. Then she explains about the inventory system and how the player can use it to learn new knowledge or study some previously obtained.

There are two phases in the act of gaining knowledge, first the player needs to find the video in the world, and then he will have to apply it in a challenge.

The first step is simple, players explore the world and interact with objects that contain a particle effect in order to get their attention, then by pressing the interaction key ("E") a video is collected and placed in their inventory. From this point on players can at any time, open the inventory menu and play the videos any number os times they wish. These videos are the representation of the object they were taken from in portuguese sign language.

The second step is the application of said knowledge, NPCs can react to the player one of two ways. First if the player hasn't acquired the necessary videos (knowledge) to answer their questions, they will offer hints as to where the player can find what he or she is missing. Second, if the player already has all of the videos needed they will be offered a multiple choice challenge (Figure 5.8).
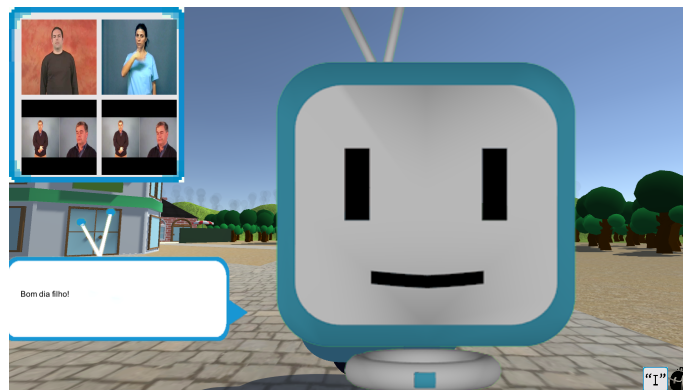

FIGURE 5.8: The challenge interface.

Four videos will be presented to the player, only one of which is the correct answer. The player has to select the correct one to proceed, if the player answers wrongly , the challenge won't exit. This is a conscious choice. The main objective of the game is to teach, to strengthen this feature it was decided early on in development that the player could only exit the challenge with the correct answer.

By only allowing the player to proceed with the correct answer the designers made sure that the player always sees the correct answer as the final one.

The downside to this is that the difficulty of the game will lower for the player, there are three levels of difficulty. If the player performs really well, answering each challenge quickly and correctly on the first try the game will get harder (without telling the player), if the player takes really long to answer or simply makes several mistakes the game will get easier. Difficulty in this sense doesn't come from the mechanical aspect of the game, videos will be in the same spot always, but the options and answers will vary according to the player's performance.

In the original *Gestual Life* players were asked to replicate the gesture for "Cómoda" which means "Dresser" in portuguese. This was one of the first gestures the game taught the player and most of them could replicate it [2].

In this new increment players are asked to replicate "Bom dia" which means "Good morning". The results can be seen in Figure 5.9.
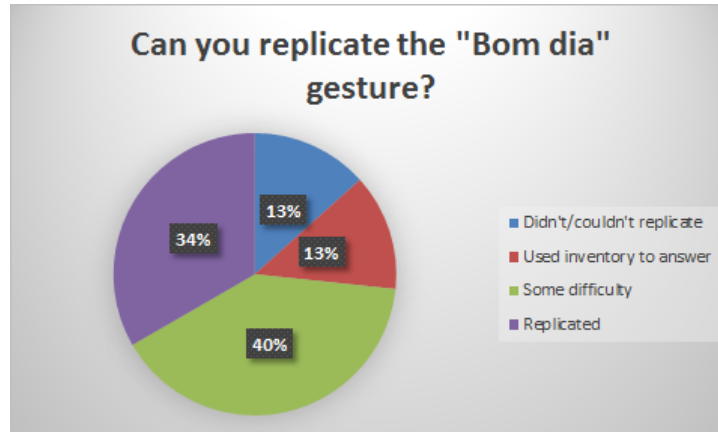

FIGURE 5.9: How players answered when asked to replicate "Bom dia".

Expected results included being able to answer or not being able to answer. Results that surprised us were, replicating any other gesture without being asked to (Figure 5.10) and using the game's systems to answer, some player when in doubt simply opened the inventory, re-watched the video and answered. At first these results weren't going to be considered because it was easy to think of them as cheating, but as stated in Chapter 4 Sub-sub-section 4.2.2.2, this was lateral

thinking encouraged by the game, the systems are there, our objective was for players to use the inventory for this same reason, they did and outsmarted us.
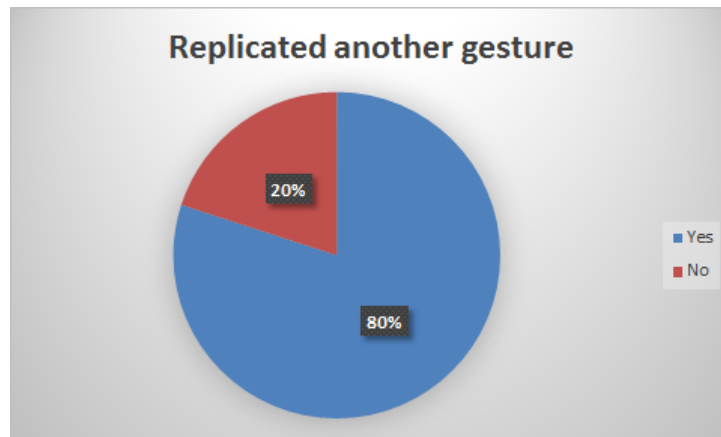

FIGURE 5.10: Players instantly replicated the gestures they remembered most.

Finally there are two last results worthy of note. First, portuguese sign language borrows heavily from mimicry [1], as such when designing *Gestual Life* this strength was considered. When the player has to pay the pharmacist for the medicine the video for "Pay" is in the inventory but nothing ever explicitly tells the player that it is there. As can be seen in Figure 5.11 most of the players paid through mimicry without having to see the video beforehand.
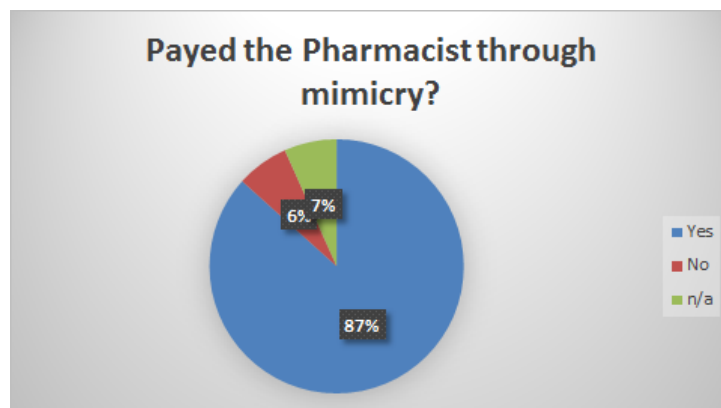

FIGURE 5.11: Most players understood through mimicry what was expected.

The ones tagged as "n/a" were players that before answering the challenge at any point opened the inventory and checked the videos thoroughly. Of these there can't be certainty that their answer was through mimicry or memory.

Of note is that these results were achieved in-game as most of the subjects did not know anything about the portuguese sign language prior to playing, the amount of subjects that answered no when asked if they knew or had contact with it can be seen in Figure 5.12.

FIGURE 5.12: Players with previous knowledge of portuguese sign language.

Chapter 6

Conclusions

In conclusion the main purpose of this dissertation is to propose a framework to aid in the development of serious games by applying development methodologies found in the entertainment games industry.

To aid in testing this framework the new increment of *Gestual Life* was developed with these steps deeply ingrained in the development process. Results show that players want to play it, find the subject intriguing and resonate with the theme as well as the world.

A vast majority of players explored the world rather than stop playing when given the chance, each that did collected every video, without being asked to.

Players were observed actively seeking knowledge, being curious about each video and what word they would learn next. LYS helped immensely, while she doesn't cause a strong first impression her constant help put players at ease and fearless of exploring.

This brings forth the conclusion that, *Gestual Life*'s new increment is a well designed experience. By following the development process described in this dissertation an enticing and interesting game resulted, where the problem wasn't having to tell someone to play but rather tell them to stop.

More than simply playing the results in terms of learning lead to the conclusion that the game isn't just engaging, it teaches as well. Subjects only played once, even in that short time most learned at least one gesture.

Gestual Life's previous instalment was the foundation for this dissertation, the results obtained in it's testing phase lead to the creation of this work. As of now this framework stands as it's legacy and improves on it's efficiency.

Although only used to develop a sign language game, this dissertation can be used to build any type or genre of game.

Lastly during this dissertation the main concept always kept present was that when designing a serious game it is best not to forget that what is being built is a game and therefore, an experience. More than simply "what" to teach the emphasis was on "how" to teach and make that experience something players actively seek.

Appendices

Appendix A

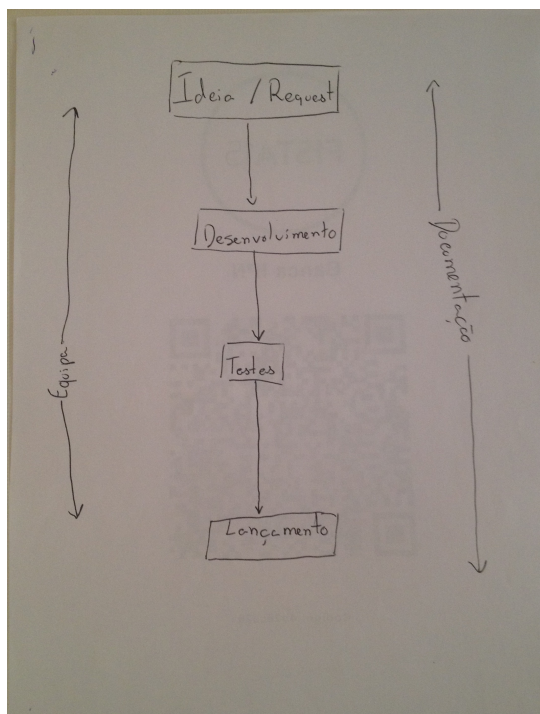# Framework prototype and the new *Gestual Life*'s Screenshots

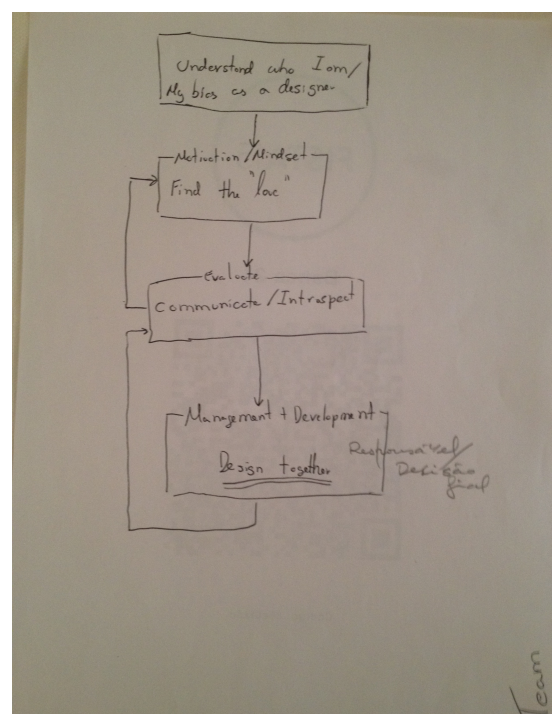FIGURE A.1: Early draft of the World Map step.

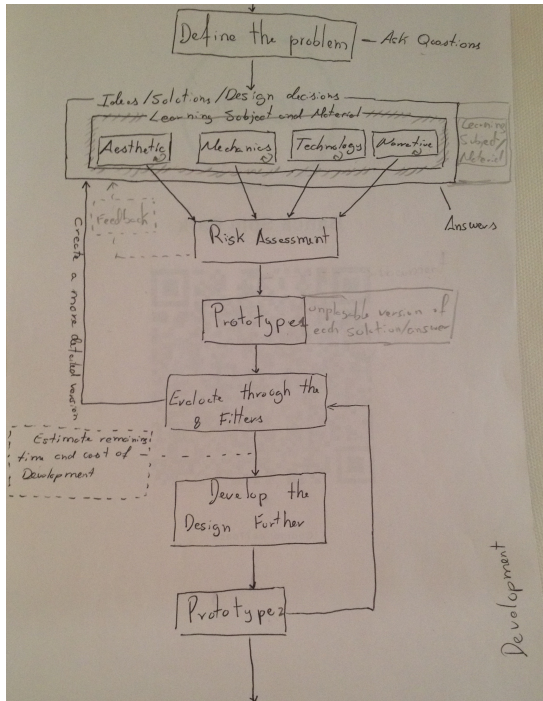FIGURE A.2: Early draft of the Team step.

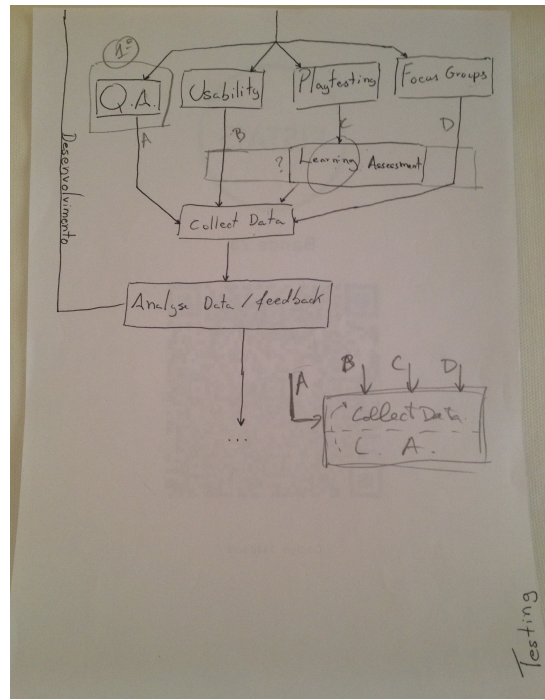FIGURE A.3: Early draft of the Development step.

FIGURE A.4: Early draft of the Testing and Results step.

FIGURE A.5: LYS helps the player learn the keys.

FIGURE A.6: Bar and Supermarket, both optional during gameplay.
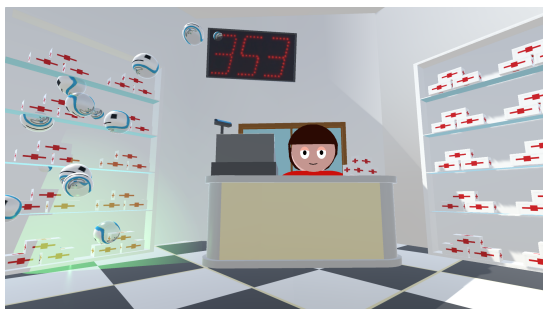

FIGURE A.7: The interior of the Pharmacy where the player buys the medicine.

FIGURE A.8: The interior of the Post Office where players mail the medicine.

FIGURE A.9: The interior of the Bar.

FIGURE A.10: The interior of the Supermarket.

Bibliography

[1] Associação portuguesa de surdos. `http://www.apsurdos.org.pt/`, 1958. Accessed: 2015-09-24.

[2] Ricardo Almeida, Luisa Brito, Pedro Sousa, Marco Capela, Pedro Faria Lopes, and Isabel Alexandre. Gestual life jogo educativo de língua gestual portuguesa. *Livro De Atas EPCG - 2014 21º Encontro Português De Computa¸cão Gráfica 21*, pages 3–9, 2014.

[3] António Andrade. Gdd?! game design document examples. `http://seriousgamesnet.eu/assets/view/238`, 06 2013. Accessed: 2015-09-15.

[4] Francesco Bellotti, Bill Kapralos, Kiju Lee, Pablo Moreno-Ger, and Riccardo Berta. Assessment in and of serious games: an overview. *Advances in Human-Computer Interaction*, 2013:1, 2013.

[5] Nathan Birch. Review: "alien isolation" is proof great games don't have to be fun. `http://uproxx.com/gammasquad/2014/10/review-alien-isolation-is-proof-great-games-dont-have-to-be-fun`, 10 2014. Accessed: 2014-12-03.

[6] Staffan Bjork and Jussi Holopainen. Patterns in game design (game development series). 2004.

[7] Roger Caillois and Meyer Barash. *Man, play, and games*. University of Illinois Press, 1961.

[8] Joseph Campbell. *The hero with a thousand faces*, volume 17. New World Library, 2008.

References
_____

[9] Chris Crawford. The art of computer game design. 1984.

[10] Extra Credits. Aesthetics of play - redefining genres in gaming. `https://www.youtube.com/watch?v=uepAJ-rqJKA`, 11 2012. Accessed: 2015-09-16.

[11] Extra Credits. Pacing - how games keep things exciting. `https://www.youtube.com/watch?v=5LScL4CWe5E`, 05 2012. Accessed: 2015-09-25.

[12] Mihaly Csikszentmihalyi and Mihaly Csikzentmihaly. *Flow: The psychology of optimal experience*, volume 41. HarperPerennial New York, 1991.

[13] Username: dourface. Monomyth or "the hero's journey" in movies. `http://www.imdb.com/list/ls052754941/`, 07 2013. Accessed: 2015-07-20.

[14] Cara Ellison. Without game manuals, buying a game just isn't as exciting. `http://kotaku.com/the-beauty-of-the-game-instruction-manual-1697341910`, 04 2015. Accessed: 2015-9-15.

[15] Blizzard Entertainment. Looking for group. `https://www.youtube.com/watch?v=xyPzTywUBsQ`, 11 2014. Accessed: 2014-12-03.

[16] Larry Frum. 40 years later, 'dungeons & dragons' still inspiring gamers - cnn.com. `http://edition.cnn.com/2014/05/19/tech/gaming-gadgets/dungeons-and-dragons-5th-edition/`, 05 2014. Accessed: 2015-08-28.

[17] Gartner. Gartner hype cycle. `http://www.gartner.com/technology/research/methodologies/hype-cycle.jsp#`. Accessed: 2015-09-25.

[18] Arthur Gies. Ori and the blind forest: The places you'll go. `http://www.polygon.com/2015/3/9/8180595/ori-and-the-blind-forest-review-xbox-one-PC`, 03 2015. Accessed: 2015-09-25.

[19] Melanie C Green. Storytelling in teaching. *APS Observer*, 17(4):37–39, 2004.

[20] Mohammad Mehedi Hassan, Mohammod Shamim Hossain, Atif Alamri, Mohammad Anwar Hossain, Muhammad Al-Qurishi, Yousuf Aldukhayyil, and

Dewan Tanvir Ahmed. A cloud-based serious games framework for obesity. In *Proceedings of the 1st ACM multimedia international workshop on Cloud-based multimedia applications and services for e-health*, pages 15–20. ACM, 2012.

[21] Johan Huizinga. *Homo Ludens Ils 86*. Routledge, 2014.

[22] Robin Hunicke, Marc LeBlanc, and Robert Zubek. Mda: A formal approach to game design and game research. In *Proceedings of the AAAI Workshop on Challenges in Game AI*, volume 4, 2004.

[23] Aki Järvinen. Halo and the anatomy of the fps. *Game Studies*, 2(1):641–661, 2002.

[24] Kristian Kiili, Sara de Freitas, Sylvester Arnab, and Timo Lainema. The design principles for flow experience in educational games. *Procedia Computer Science*, 15:78–91, 2012.

[25] Raph Koster. *Theory of Fun for Game Design 2ed.* O'reilly, 2013.

[26] Sid Meier. Sid meier - interesting decisions. `http://www.gdcvault.com/play/1015756/Interesting`, 2012. Accessed: 2015-07-15.

[27] Metacritic. Dark souls, metacritic score page. `http://www.metacritic.com/game/playstation-3/dark-souls`, 2011. Accessed: 2015-07-06.

[28] Metacritic. Hearthstone: Heroes of warcraft, metacritic score page. `http://www.metacritic.com/game/pc/hearthstone-heroes-of-warcraft`, 2014. Accessed: 2015-09-26.

[29] David R Michael and Sandra L Chen. *Serious games: Games that educate, train, and inform.* Muska & Lipman/Premier-Trade, 2005.

[30] Geoffrey Morrison. Poll: Is 3d tv dead? do you care? - cnet. `http://www.cnet.com/news/poll-is-3d-dead-do-you-care/`, 08 2015. Accessed: 2015-08-26.

[31] Nintendo. Zelda universe. `http://www.zelda.com/universe/`, 2015. Accessed: 2015-09-25.

[32] Darren Orf. 3d tv is dead. let's hope smart tv is next. `http://gizmodo.com/3d-tv-is-dead-lets-hope-smart-tv-is-next-1677633906`, Jun 2015. Accessed: 2015-07-15.

[33] Jenna Pitcher. Eve online wages largest war in its 10 year history. `http://www.polygon.com/2014/1/28/5352774/eve-online-wages-largest-battle-in-its-10-year-history`, 2014. Accessed: 2015-08-28.

[34] Chris Remo. Randy smith: Do games need to be fun? `http://www.gamasutra.com/view/news/117050/Randy_Smith_Do_Games_Need_To`, 11 2009. Accessed: 2014-12-03.

[35] Danielle Riendeau. The best of the rest of 2014: Alien: Isolation is the boldest, best game of the year. `http://www.polygon.com/2014/12/28/7456433/alien-isolation-game-of-the-year-aliens`, 12 2014. Accessed: 2014-12-28.

[36] Tim Ryan. The anatomy of a design document, part 1: Documentation guidelines for the game concept and proposal. `http://www.gamasutra.com/view/feature/131791/the_anatomy_of_a_design_document_.php`, 10 1999. Accessed: 2015-09-15.

[37] Derek Sakamoto. How blizzard's warcraft brought hearthstone to life - gdc 2015. `https://www.youtube.com/watch?v=GstFjGP4WFU`, 03 2015. Accessed: 2015-09-26.

[38] Jesse Schell. *The art of game design*. Elsevier/Morgan Kaufmann, 2008.

[39] Chris Solarski. The psychology of lines, shapes, and volumes. `http://www.gamasutra.com/view/feature/185676/the_aesthetics_of_game_art_and_.php?print=1`, 2013. Accessed: 2015-07-17.

[40] Statista. Video games revenue worldwide from 2012 to 2015, by source (in billion u.s. dollars). `http://www.statista.com/statistics/278181/video-games-revenue-worldwide-from-2012-to-2015-by-source/`, 2012-2015. Accessed: 2015-09-21.

[41] Clifford Stoll. *High tech heretic: Reflections of a computer contrarian.* Anchor, 2000.

[42] Tarja Susi, Mikael Johannesson, and Per Backlund. Serious games: An overview. 2007.

[43] Kshiraj Telang. Game art bible - secret sauce to making great game art. `http://www.slideshare.net/pencillati/game-art-bible-secret-sauce-to-making-great-game-art`, 11 2013. Accessed: 2015-9-15.

[44] Sun Tzu. *The art of war.* e-artnow, 2012.

[45] Richard Van Eck. Digital game-based learning: It's not just the digital natives who are restless. *EDUCAUSE review*, 41(2):16, 2006.

[46] Owen Williams. 3d tvs are dead at ces 2015, now the fight's moved to 4k. `http://thenextweb.com/insider/2015/01/07/3d-tvs-dead-ces-2015-fight-moving-4k-curved-displays/`, 1 2015. Accessed: 2015-07-15.

[47] Michael Zyda. From visual simulation to virtual reality to games. *Computer*, 38(9):25–32, 2005.