

# MANAGING GAME PRODUCTION MANAGEMENT

*The concept of Gamifying the Game Development Process*

**This page was intentionally left blank**

## I. TABLE OF CONTENTS

I.	TABLE OF CONTENTS .....	2
II.	TABLE OF FIGURES .....	6
III.	READER'S GUIDE .....	7
IV.	MOTIVATION .....	8
V.	INTRODUCTION .....	9
VI.	INITIAL PROBLEM STATEMENT .....	10
VII.	INVESTIGATION .....	11
A.	Game Production.....	11
	Pre-production.....	12
2.	Production .....	15
3.	Testing.....	16
4.	Post-production.....	17
B.	Team Composition.....	17
C.	Management Methodologies .....	19
1.	Waterfall.....	19
2.	Agile.....	19
3.	Interpretation .....	21
D.	Engagement.....	21
1.	Gamification.....	23
2.	Game Design .....	26
E.	State of the art:.....	26
1.	Shotgun software.....	27
2.	ChoreWars .....	27
3.	RedCrittter Tracker .....	28

F.	Management Application requirements analysis .....	28
G.	Conclusion.....	29
VIII.	LITERATURE REVIEW.....	31
A.	ChoreWars .....	31
1.	Structure .....	31
2.	Use of gamification .....	32
3.	Limitations .....	32
B.	RedCrittter tracker.....	33
1.	Structure .....	33
2.	Use of gamification .....	33
3.	Limitations .....	34
C.	Types of gamification .....	34
D.	Gamifications effect on motivation.....	37
1.	Goal .....	37
2.	Limitation .....	37
3.	Experimental setup .....	37
4.	Shortcomings.....	38
IX.	FINAL PROBLEM STATEMENT .....	39
X.	EXPERIMENTAL DESIGN.....	40
A.	Introduction.....	40
B.	Application design.....	40
1.	Gamification design.....	40
2.	Agile design.....	42
C.	Requirements .....	42
1.	Important features for the management application.....	42

2.	Important features for the Gamification element.....	43
D.	Prototyping.....	44
1.	Prototype design .....	44
2.	Unity UI system.....	45
3.	Unity editor system .....	46
E.	Usability testing .....	47
1.	Approach.....	47
2.	Observation .....	47
3.	Consideration.....	48
F.	Experimental Design .....	48
1.	Online experiment.....	48
2.	Focus group .....	49
G.	Summary .....	50
XI.	IMPLEMENTATION .....	51
A.	Introduction.....	51
B.	Pre implementations .....	51
1.	Git implementation.....	51
C.	Application implementation .....	52
1.	Master script.....	52
2.	Core structure.....	53
3.	Dragable content.....	53
4.	UI system .....	53
5.	Prefabs .....	54
6.	Gamification.....	54
7.	Animation setup .....	55

XII.	FINDINGS .....	56
A.	Pre-application.....	56
1.	How do you plan your productions?.....	56
2.	What are the largest issues with that process? .....	57
3.	What are the most essential features for a management tool? .....	57
B.	Post-application.....	57
1.	Could gamifying management tools potentially engage users more? .....	57
C.	Usability testing observations .....	58
XIII.	DISCUSSION.....	59
XIV.	CONCLUSION .....	61
XV.	FUTURE WORK .....	62
XVI.	BIBLIOGRAPHY .....	63
XVII.	FIGURE BIBLIOGRAPHY.....	66

## II. TABLE OF FIGURES

Figure 1: Showing the difference between game graphics.....	11
Figure 2: The three categories of features.....	13
Figure 3: Description of code iterations .....	14
Figure 4: Various iterations of a 3D game model.....	15
Figure 5: An example of Game development roles .....	17
Figure 6: The Agile and Waterfall model.....	19
Figure 7: Work engagement .....	21
Figure 8: An example of basic Gamification.....	23
Figure 9: The essence of Game Design.....	26
Figure 10: A screenshot of ChoreWars .....	27
Figure 11: A screenshot from RedCritic Tracker .....	28
Figure 12: The use of avatars in ChoreWars.....	32
Figure 13: Arranged from top to bottom is the Valence, Arousal and Dominance scale.....	38
Figure 14: Unity UI Prototype.....	45
Figure 15: Unity Editor Prototype .....	46
Figure 16: A screenshot of the ManagementSystem script .....	52
Figure 17: The Gamification element .....	54

### **III. READER'S GUIDE**

Before embarking on the journey of gamifying game production management, it should be noted that this study was conducted as a proof of concept and as such should not be considered an investigation resulting in a finite prototype.



#### IV. MOTIVATION

Since the industry has been growing both technologically and economically, there is always something new to discover within the game industry. I have personally, been part of multiple game productions and production management seems to always be a problem for gam developers, either because we forget to complete tasks and slow down pipelines or because we lose our motivation for working. Moreover, it is interesting to see how little gamification is used within the industry being the most qualified to utilize it. Furthermore, since gamification has been proved to motivate and engage people in numerous ways from collecting coupons for a discount to reporting back for a bonus at specific intervals. If this concept could be applied to management and more specifically to particular assignments for a more motivated solution on said assignment it could potentially increase our productivity as humans. Within the game industry, this would mean an increased amount of content in a decreased amount of time, potentially allowing more companies to avoid crunch time or bankruptcy. As such this is an interesting concept to investigate, not only to me, but to the entire industry.

## V. INTRODUCTION

With the growth of our technological society, many new technical complications arise, being both problems we as a society did not know were problems, but also entirely new difficulties we could have never anticipated. One such complication would be with software development, as software is typically difficult to estimate in terms of production time. Some management methodologies have been specifically designed to handle this problem such as agile development and more specifically Scrum. However, we still see problems in certain areas of our technological industries. The game industry has grown massively and is now one of the world spanning industries with the highest grossing products of any industry. As such, many aspiring game developers aim to be part of the industry with their small and less economical dependent products. Within the technological working environments we have started to encounter expressions like: “Crunch-time” and “You make money by shipping things” within the game industry. These expressions indicate multiple issues about the economical point of view on game industry, but also the way developers fundamentally think about their products. Game companies are often economically dependent on a successful development, shipment and income of a single product, hopefully earning a bare minimum of its production value. This puts a lot of game companies at risk when designing and developing games, as these productions tend to take years, meaning that an unsuccessful production could potentially cause a game company to go bankrupt. Thus the game industry turns their focus towards optimizing and improving the management of million dollar game productions. One such way of improving work is to bring the teams into a state of flow through the use of gamification which will both keep them motivated and possibly increase their productivity. Additionally, smaller and possibly inexperienced production teams lack guidelines for how to startup and manage their team and guide them towards a common vision and finally publishing their very first product. These teams might have other issues with employers not using systems to track task completion or simply not pulling their share of the workload. More so is the problem of managing a multidisciplinary group of people with personalized workflows towards a common goal of producing a final product on time and within the budget.

## VI. INITIAL PROBLEM STATEMENT

Based on section V, it would seem that there is a problem with the flow of game productions. Mainly the problem seems to be rooted in the team composition as multiple departments have to be working on various aspects of the same product at the same time. This means that some parts of the production might finish work before others while other parts of the production team might be delayed or completely without work for durations. Additionally if someone forgets to pass on a task, the entire pipeline will be blocked based on that one mishap which in worst case scenario will lead to a worse product. This problem seems to be one coming from the way game productions use pipelines containing multiple disciplines working on a specific part of an asset before passing it on. As such, it comes down to the planning of every asset and ensuring that the team consists of enough multidisciplinary people to adjust the team compositions based on requirements. Or perhaps ensure that the production is balanced depending on the sub teams' size and the work each department has to complete within a specified time. Furthermore, the teams might consist of people with various skillsets, possibly meaning that their approach and handling of the same task might vary. Moreover, it seems to be of outmost importance to a game production that the production team is always motivated to complete the tasks at hand while still maintaining a good morale. This indicates a certain need for an equally approachable system helping multidisciplinary teams focus on the tasks at hand while still providing the people in charge of management with tools to help them keep the production on track. As such, the initial problem statement becomes:

**“Is it possible to make management on a game production more engaging for both the management and content producers of a multidisciplinary production teams?”**

## VII. INVESTIGATION

This chapter will investigate how game productions are structured, their production workflow and the teams involved in the productions. Additionally, this chapter will explore various management methodologies and compare them with the results of the game productions investigation finally reaching a conclusion and a delimitation for this project.

### A. Game Production

About 30 years ago, the game industry was still quite small and making a living as a game developer could be quite harsh(Chandler, 2009). For a single individual it would be possible to develop a game complete with all assets in as short a timeframe as six weeks. That one person would be in charge of the entire production from developing the design of the game as well as implementing the games functionalities, followed by making graphical and auditory assets.

Since the end-user requirements were significantly easier to satisfy back then, a simplistic product with minor errors could entertain users for hours. However, According to Chandler(Chandler, 2009), the game industry has changed significantly, as the end-users of AAA products have come to expect hours of content within a living, breathing and completely immersive world. As such, the requirements for games have evolved and an increasing amount of people tend to be involved in the productions taking place today.

Fundamentally, all game productions can be broken down into four phases: Pre-production, production, testing and post-production. These four phases all have a couple of requirements which when solved advances the project into the next phase. Concurrently, two different departments might be working on two different phases. Following this point, each of the four phases will be elaborated upon.

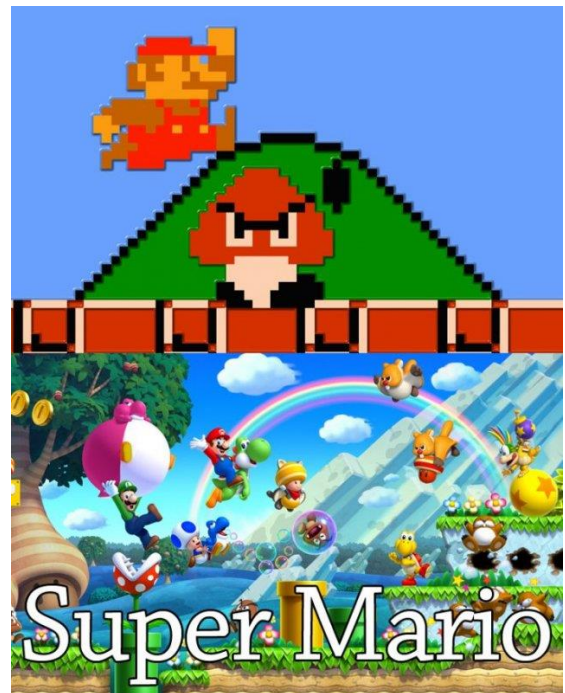


Figure 1: Showing the difference between game graphics

### Pre-production

The pre-production is the first phase of the entire production. Within this phase, the production team strives to develop a concept which is both engaging and captivating to the end-user. Once the core concept is finalized, the team plans the actual production by determining the project requirements and developing a project plan based on those requirements. Having the entire production planned out, it will also become quite prominent that certain areas of the plan could potentially break when a single feature takes too much time to implement or the team is stressed out.

#### a) Game concept

A game concept is regularly the first element to be determined of a game production, as the concept is fundamentally the core of the game. A concept for a game typically involves:

- Elevator pitch – describing the essentials of the game in a sales pitch
- Game setting – describing the environment and atmosphere in which the game takes place
- Gameplay mechanics – describing the game systems and player interactions in the game
- Story synopsis – describing the most important elements of the story
- Concept art – describing the visual style of the game
- Audio elements – describing the auditory style of the game

Typically, the concept is developed by a lead team consisting of a specialist from each of the departments to ensure that the concept is equally expansive for each department. This is often to avoid settling on an idea where one department might finish their work in a month while another has to develop for multiple years. After having an initial concept, the lead team might include other members in the concept development to generate new ideas or simply proceed to the next step of flushing out the game concept.

Having an initial concept, the lead along with the rest of the team starts discussing the genre of the game. Notice however, that this step can be overlooked as many developers combine genres or develop entirely new ones themselves. Although sometimes disregarded, settling on a genre often helps the team align the understanding of how the game plays and looks for the entirety of the team.

When developing a game for a specific platform, the concept of the game has to align with the hardware capabilities and utilize the unique hardware functionalities to ensure that the game grants the end-user the best gameplay experience possible. Thus defining the targeting platform can help the team decide on further gameplay elements or game assets.

Having currently settled upon a concept for a specific console or multiple, the next step is to validate whether or not, the production of such a product would be economically beneficial. This is mainly to avoid unpleasant realizations further in the development cycle as extended game productions tend to lead a company towards bankruptcy(Galanakis, 2014).

### b) Game requirements

Since game development is an iterative process, managing, organizing and prioritizing every single feature of the production is extremely important to ensure the best possible product for the end-user. Thus, examining the game concept and defining the features of a game is the next step, followed by prioritizing each feature. Here it is proposed by Chandler to separate each feature into one of three categories, depending on what the feature is about(Chandler, 2009).

The first category is called Process and includes features focused on improving and optimizing the development process and can be anything from updating design documentation to setting up new workflows for approving work.

The second category is called Production and relates to the optimization of any tool or technology used to develop the game, such as new tools to improve workflow or optimization of physics systems.

The last category is called Gameplay and encompasses any feature relating to the gameplay experience of the end-user. These could be the players ability to control vehicles or allowing for the player to customize their avatar.

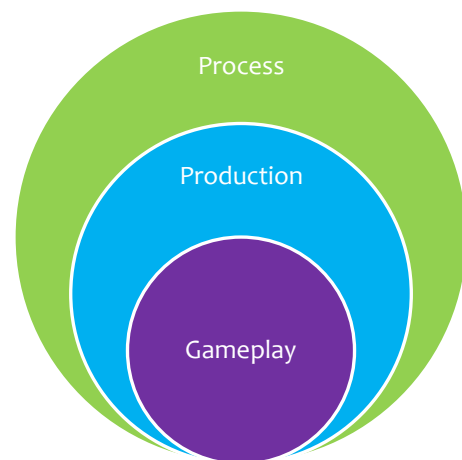


Figure 2: The three categories of features

After having settled on a category for every single feature, the next step is to prioritize the features based on how important they are to each department. Each lead will be tasked with prioritizing every feature based on their idea of how important the feature is to the production. Afterwards the prioritizations are collected and the average determines how important the specific feature is for the overall game. In this way, every department has a say in the tools being developed and the prioritization of assets. The downside to this unfortunately is that if a lead artist and designer do not understand the importance of a tool capable of saving the engineering department some work, it might never be developed.

With a prioritized list of features to develop, the final step of the game requirements can be defined through planning which features should be done in specific timeframes. When developing a game, the game is often segmented into numerous iterations being namely: First playable, Alpha, Code freeze, Beta, Code release. Some productions might have additional milestones, but planning for these basic five production milestones should be suitable for most pre-productions. The milestones should be segmented into specific timeframes and the game requirements for each department should be listed for the appointed lead to distribute sliced tasks.

In order for the production to start the specific tools used during the production should be decided upon and the documentation should have been written.

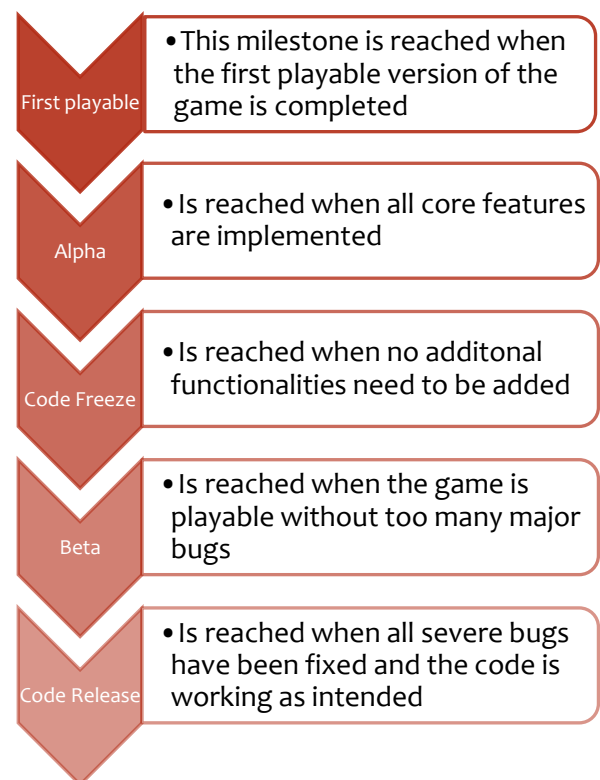


Figure 3: Description of code iterations

- The lead engineer should create a technical documentation containing platform specifications, limitations and code standards in addition to planning the production pipeline with the other leads
- The lead artist should create concept art, art documentation, asset lists and art tool guides
- The lead designer should write a design documentation of the gameplay mechanics



### c) Game plan

In opposition to the two previous sections, the game plan section is quite simple as it mainly focuses on the further flushing out of features trying to break every single aspect into a single solvable task capable of being finalized within a specified timeframe. This includes determining dependencies between assets like making the 3D model of a character before starting the texturing of the character.



Figure 4: Various iterations of a 3D game model

Furthermore, the importance of task tracking is stressed as having an overview of the tasks in progress will help a producer or a lead understanding where problems are occurring and who has a lot of work in opposition to those being without work.

Finally the concept of having a pre-production checklist is proposed to ensure that the project will not start before everything in the pre-production steps is planned and accounted for.

## 2. Production

The main objective of this phase is to produce and track the production of assets as these are essentially what make up a game. Each department will already be hard at work to finalize specific features for the next milestone, but sometimes the plan already agreed upon will not be maintained, thus it is important to make changes to the plan by confirming which features are essential to the product. Furthermore, new tasks might occur, such as updating existing features or developing new assets not yet accounted for. Thus there are mainly three aspects of the production cycle.

### a) Task tracking

All the tasks having been defined in the Game plan chapter are essentially going into the production and in order for a lead or a producer to have any idea of how far along the development has come it is extremely important for the team to report their progress with a specific task. Often times, working on an asset for the game might result in a slower completion time than initially planned for, thus it is



important for both the lead and the producer to keep track of how much time is spent on each individual task. Furthermore, every task should have exit criteria to determine when a product is done.

#### **b) Reorganizing**

It is explained that in order to ensure that the project is on track, it is important to have weekly status reports to have an idea of what problems might have occurred. Typically when working with programmers, there are some features which will end up taking more time than initially planned for, thus it is important to discuss why this might have happened and how to solve the issue. In most cases, this can be solved by either increasing the scheduled time for that specific task, increasing the resources meaning how many are working on that feature, cutting features or functionalities from that specific task and lastly by reducing the quality. Each of these options obviously enforce the project plan to change by anywhere from a slight margin to several months. Additionally, it will be the producer's job to sort out newly created tasks based on their relevance to the final product.

#### **c) Producing**

The last part of the elements making up the production is understandably the production of several assets for the game itself, whether its code, 3D models, drawings or sound clips. Depending on the game concept, there might be more or less departments. For instance, the game could be utilizing only 2D graphics for its graphical assets or maybe the game uses procedurally generated sounds, making the sound assets mainly a programming task. This primarily indicates that the departments can vary quite a bit depending on the game concept or even based on the people at the productions disposal.

Once again it is proposed to use a checklist for making sure that the production is on track from a producer's perspective.

### **3. Testing**

As testing is not a specific phase of the development of a game, it is ongoing throughout the entire production from the first playable prototype to the final product has to be thoroughly tested. Although the testing phase does not have clear goals on its own, the main responsibility of the testing department is to validate the completion of assets according to the exit criteria's of the assets.

Thus the testing department will be dependent on an updated production plan to plan their schedules for testing hardware, language selection and finding bugs.

#### a) Code Release

During production of the game, the developers will develop a number of builds of the game that they expect to be ready for release, called Code Release Candidates or CRC for short. When the product is about to be ready for release, the testing department will go through a number of CRCs and give them one last check to validate if they are releasable.

#### 4. Post-production

Finally once the game has been deemed releasable, the production has to be wrapped up. This can be done in numerous ways. However, Chandler suggests using a post mortem meeting to discuss what went well and the lessons learned during the production to get a better understanding of the production and possibly be enlightened as per the coming production (Chandler, 2009). Additionally there is a final step, basically ensuring that the game as well as all assets, work files, plans and anything else related to the project could be saved for a later usage.

### B. Team Composition

A topic briefly touched upon during section VII. A, is that of the team. On a game production a team composition can change quite drastically depending on the game concept. However, though the game concept often has the biggest influence on the team composition, it is not the only variable. Teams can be varied depending on competences, economical limitations or the target platform of a production.

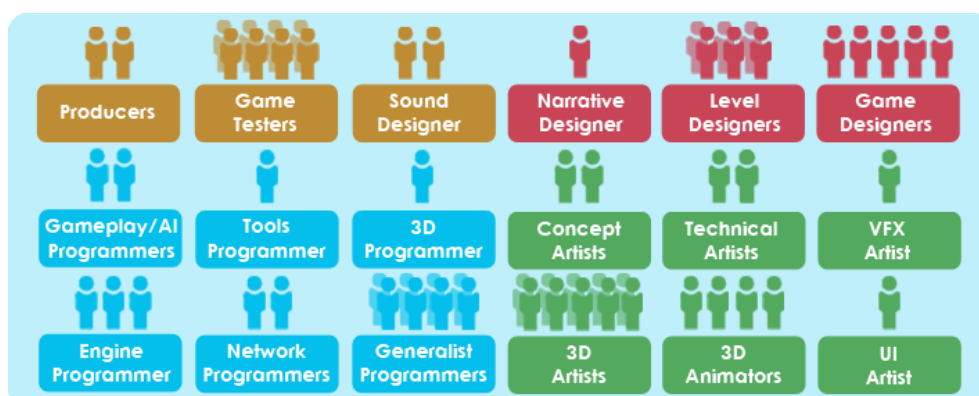


Figure 5: An example of Game development roles

As such, in order to fully understand the different team compositions and the span between, it would be interesting to investigate various game productions in terms of their products as well as the team behind them.

Based on an article by Nathan Heller from Gamesradar(Heller, 2014), it would seem that developing a game as one single individual is hardly rare anymore. Even so, it is possible only a handful of developers who manage to manifest the dedication and motivation to develop an entire game on their own. This is an impressive feat considering the amount of different skillsets, work and money going into other productions in the current videogame generation(Doulin, 2010). Typically, the games produced by a single individual lack in content, but deliver on an experience or underlying thematic whereas many AAA productions focus on the production of content.

In contrast, larger studios like Ubisoft can be taken into consideration as an example of more expansive game productions. Most of the games produced by Ubisoft have a much higher production value than games crafted by a single individual. This is particularly interesting in the art of the game productions at Ubisoft since most of their games look quite different from one man projects. Based on an interview with Pauline Jacquey(Weber, 2014), the teams for AAA productions should have reached about 600 participants with the fourth console generation and as such are capable of producing an enormous amount of content for the players to explore, both visually and auditory.

Finally, another type of to be considered would be the teams realizing their limitations and instead of hiring people to fill in additional roles, they aim to utilize the skillsets already at their disposal. These teams understand their limitations, but yet still manage to produce a complete game containing sound, graphics, code and story. A team could for instance choose to produce their graphics through code if they lack great artists meaning that there would be no need for an art department, but a higher number of programmers to solve the task of producing algorithms capable of creating the graphical assets procedurally. Such a team might even distribute their programming department into multiple departments handling different features for the game.

## C. Management Methodologies

As it becomes clear that there seems to be various team compositions based on several aspects of the production, the management will have to be easily adjustable, but still provide a solid framework for how to organize the participants. Based on that statement, it would be of significant interest to investigate various different management methodologies. When a team decides which type of management methodology to utilize, it seems too often come down to either using a waterfall model or an agile model(Chan, 2013). As such, these two models will be described and compared in accordance with what was discovered in the previous section.

### 1. Waterfall

The waterfall model is the most commonly used model, as it follows a traditional workflow going from one assignment to the next slowly filling in the blanks. Commonly, the requirements for the project are figured out before the project starts. This means that using waterfall has a very structured approach to a project, often leaving it hard to adapt to changes. The model is defined as followed:

- Very structured
- Typically revolves around one big project
- Solving assignments sequentially
- Highly useful for projects without change
- Internal development
- Requires the project requirements to be finite from start

### 2. Agile

The agile model on the other hand is very flexible as it typically slices the requirements of a project into lesser elements, often making it easier to change plans during a project. This is mainly due to the constant reviews of the consumer throughout the entire process, giving the customer the option to discuss changes at multiple occasions during the production.

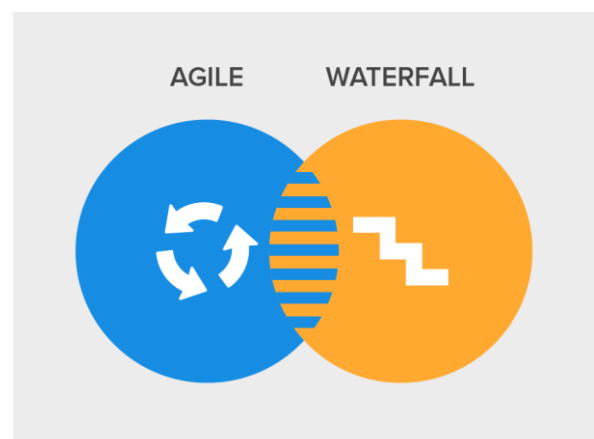


Figure 6: The Agile and Waterfall model

This model is defined by the following traits:

- Very flexible
- Segments a project into smaller projects
- Highly collaborative
- Typically suited for projects aiming at continuous improvements
- Involves the customers in the development
- The requirements are often changing and evolving

#### **a) Scrum**

Scrum is a specific agile model incorporating specific approaches to handling a changing production. It utilizes three specific roles for involved members of the production, namely: Product owner, Team members and a Scrum Master. The Product owner is the client, the team members are the developers of the product and the scrum master is the one ensuring that the product is running smoothly and according to the schedule.

Furthermore, scrum utilizes short daily meetings called standup meetings meant to last only a couple of minutes. These are used to discuss what everyone is currently working on, if any problems have occurred or if someone has finished all their tasks. Additionally scrum slices the production of specific features into short timeframes typically spanning a week to three weeks. Within this timeframe, called a sprint, the feature should be capable of being produced and implemented into the end product. Finally, scrum works in iterations and will typically involve a status meeting with the product owner every sprint or so. This means that the product owner has full control over the direction of the project and which features are added when as the product will have a new version every meeting. Furthermore, this ensures that if the product owner suddenly realizes something that works differently than anticipated, he can immediately comment upon the feature and have it changed.

#### **b) Kanban board**

Kanban boards is another approach to managing a production, being not a direct opposition of scrum, but somewhat utilizing the same elements(LeanKit Inc., 2015). Kanban boards can be used even while utilizing the flexibility of Scrum development or the structured approach of the Waterfall model although it provides an overview following that of agile development. A Kanban board is essentially a way to visualize workflow and the state of every task on a production. It utilizes a board segmented

into areas such as: To do, in progress and completed. Every task is depicted with a small post it containing the task description and the person assigned to the task. The task is then moved by the person assigned from one area to the other visualizing the flow of the production. This method is great for visualizing the progress and status of a team production, which can lead to potential further engagement for the team.

### 3. Interpretation

As game development often involves both creative artists and organized programmers, utilizing a waterfall model type of management methodology for a game production could potentially cause issues in regards to the iterative process of progress (Galanakis, 2014). In opposition, the Agile model would be preferable for the flexibility of the productions progression, although utilizing a structured approach such as scrum could theoretically limit the creative process of the artists which might cause problems for the product. As such, utilizing an Agile management methodology would be preferable, although the limitations of scrum could pose a threat to the product.

## D. Engagement

Engagement is a concept which most people know from doing their favorite activities, and feeling completely absorbed by that one action. However, if the goal is to make people feel engaged at work there is a need to first fully understand the concept of work engagement.

Keeping the employees engaged at work is currently quite a problem as many feel like there is little to no motivation for during the tasks they are assigned. A report from Gallup containing information from 140 different companies states that as many as 87% worldwide are not engaged with their work (Burger, 2015).



Figure 7: Work engagement

In order to solve this problem, it is of outmost importance to understand the concept of being engaged at work. As such, engagement at work can be described as feeling vigorous, dedicated and absorbed and often presents itself in a feeling of empowerment, losing of track of time and finally a feeling of enthusiasm. Work engagement is described by Schaufeli as:

*“A positive, affective - motivational state of fulfillment that is characterized by vigor, dedication, and absorption”- (Schaufeli, 2011)*

He later describes in details what is meant exactly with each characteristic. He describes that feeling vigorous means having a high level of energy and resilience and being persistent even when there are issues with the work at hand. Furthermore, he describes being vigorous as having a willingness to invest more time and effort in the job.

Moreover, he describes dedication as a strong involvement in the tasks at hand accompanied by a sense of pride, significance and inspiration. Finally he describes absorption as a state of total immersion where it becomes impossible to keep track of time as the only element of importance is the source of the immersion, hence the work.

Understanding the good work engagement can do for employees and employers it would be quite valuable to find a way to design the working environments for such a state of mind.

But the question still remains of how to engage workers. According to Schaufeli engagement is caused by:

- job autonomy
- social support and coaching
- performance feedback
- opportunities to learn and to develop
- task variety
- responsibility
- transformational leadership
- value fit
- organizational justice

Striving to apply every one of these should potentially strengthen the work engagement amongst employees and employers. In addition, it is stated by an employee engagement consultant that there are other means of achieving work engagement amongst employees (Velasquez, 2014). The argument is to use multiple methods containing the usage of rewards, employee metrics and competitive elements. The previously mentioned methods all relate to a single topic which is that of applying engagement through the use of gamification.

### 1. Gamification

Although the most commonly used application of gamification is within the field of marketing, the concept has begun spreading to other fields of interests (Skou, 2012). On a website, 25 actual use cases of gamification were stated ranging from the military recruitment system to solving world problems through simulations and though most involve marketing, there are some focus more on personal motivation (Stanley, 2014). This is explained by Kapp (Kapp, 2014) as the potential Gamification possess to engage users in an activity through interactivity. In addition, Kapp states that if the gamification elements are designed properly, the system can even have an influence on the behavior of that an individual. This can be used to encourage learning or motivate users to



Figure 8: An example of basic Gamification

take certain actions and influence the users' behavior. Although a very simplified definition for gamification, this definition condenses the entire essence of what gamification is into a single phrase:

***"The use of game design elements in non-game contexts" - (Groh, 2012)***

The concept of gamification arose from the idea of gamifying markets, websites and applications to further engage and satisfy the users, leading to a better more substantial business (Skou, 2012).



As such, many products and companies have been utilizing the concept by rewarding users with virtual points, levels or other types of benefits(Bunchball inc., 2016). This in turn makes for a better and more rewarding interaction between the product and the users(Kapp, 2014). While this is a utilization of gamification, many have started to believe that this is the only way to use the concept(Kapp, Blair, & Mesch, 2014). Gamification has become associated with rewarding points and badges based on actions, although gamification itself is the option to apply extensive amounts of game design to a non-game context. As such, in order to utilize gamification, it is tremendously important to understand how to redesign an experience or product into a gamified one. And thus, it becomes evident to differentiate gamification from a regular game in order to utilize the concept, hence the definitions for games and gamification will be compared:

Definition of a Game:

*“A system in which players engage in an abstract challenge, defined by rules, interactivity and feedback, that results in a quantifiable outcome often eliciting an emotional reaction”- (Kapp, 2014)*

Definition of Gamification:

*“Gamification is using game-based mechanics, aesthetics and game-thinking to engage people, motivate action, promote learning and solve problems”- (Kapp, 2014)*

It becomes clear from the definitions that Gamification incorporates many of the same aspects on which we base our games. However, the biggest difference lies with the game being a self-contained unit in contrast to Gamification which is often used in conjunction with other activities, thus not being self-contained. Furthermore, though both are incorporating the same freedom to fail and rewards, the intent of these is slightly different as gamification focuses on motivating the user for an activity rather than self-contained progression as it is with games.

In addition Gamification can be split into two subgroups, namely structural and content gamification(Kapp, 2014). Structural gamification is focused on motivating the users through the use of rewards for completion of tasks or progression, whereas content gamification is changing the content to incorporate game behavior, such as racing while learning how to calculate. Additionally, structural

gamification mainly uses extrinsic motivation in opposition to content gamification using mostly intrinsic motivation. However, both can be used to drive users to innovate and build independent skills and knowledge, but there are some advantages to using either one. Content gamification can motivate users to go through content by making it more engaging and as such will help the users acquire new knowledge, whereas structural gamification can motivate users to go through a curriculum and return to the curriculum on a regular basis. Furthermore, structural gamification has clear goals, incremental goals and rewards, progression, real-time feedback, transparency, status, high stakes/challenges, time, rules, reward structures, leaderboards, levels, points and social sharing. Content gamification on the other hand utilizes story, challenges curiosity, character, interactivity, feedback and freedom to fail.

One problem that seems to arise is how a gamified activity becomes increasingly trivial after quite some time interacting with the activity, or how the rewards seems to matter less and less when the users has collected 100 gold stars and is granted yet another one. As such, it is proposed by Kapp to utilize storytelling for the engagement of the users across a predetermined duration(Kapp et al., 2014). Here it is proposed to apply the hero's journey in order to evoke intrinsic motivation in the users through them being the hero. Although creating an entire narrative for a gamified purpose to stay engaging, it takes vastly more time to produce and implement rather than a point system. In addition, the stories would be in need of changing every now and then, whereas a point system is implemented once and it works.

Finally, Gamification can utilize both extrinsic motivation and intrinsic motivation to keep users engaged, however the best gamification utilizes both intrinsic and extrinsic motivation at once. Although in order to fully utilize the concept, it would be of significance to firstly acquire an understanding of Game design as having an understanding of how to design game elements would be beneficial when designing a gamified experience.

## 2. Game Design

As such, understanding game design is an important skill, as without a proper understanding of how to design the gameplay elements of a product, the gamification risks becoming a hindrance rather than a positive experience for the users, leading to lesser sales. Game design can be associated with the design focused on interactive elements in a game, e.g. movement, shooting, maneuvering amongst others. These are referred to as game mechanics, meaning basically every interaction a player has within a game world. In other words, for a product to be gamified, it needs to incorporate some interactivity with the users.

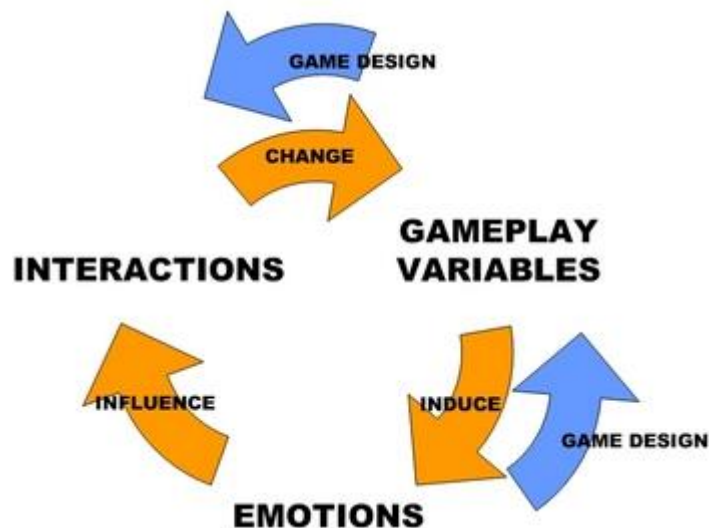


Figure 9: The essence of Game Design

However, interactivity is a broad term which can mean anything from leveling systems to a dinner with the distributor of the product. Here it would seem to come down to two groups, the first being mechanics and the second being feedback. If mechanics would be the impact the player has on the game, the feedback would be the impact the game has on the player. Mainly, feedback systems are used to place extra emphasis on specific elements of a game such as, adding extra motion, colors and sounds to an impact in order to catch the attention of the user (Jonasson & Purho, 2012). Additionally these elements, referred to as Juice, are used to attract the player's attention to specific elements which the player should possibly approach carefully or maybe even interact with. As such, visuals and audio can be used to place emphasis on a specific mechanic of a game and can thus be used to direct the player's attention or even guide the player.

## E. State of the art:

Though gamification has spread like wildfire and many applications have started implementing systems with game design elements, there are still only a couple of management systems utilizing gamification. In addition, most of the more commonly known management systems have no gamification elements,

but are still considered the best tools for managing productions(Chapple, 2014).

This chapter will investigate relevant applications and discuss which elements might be beneficial to the development of the product.

### 1. Shotgun software

Although shotgun software does not utilize gamification, the working environment they created for artists is impressive(Shotgun Software Inc., 2015). If you have a leading position on a team, you can easily navigate all assets, rather than tasks and quickly move between the assets, the level the assets are associated with and even other assets related. Another impressive feat is that upon making a change, the system itself will handle version control like uploading too perforce or git. Additionally, artists easily get notified on work they have completed or gotten feedback upon.

### 2. ChoreWars

This application is quite heavily focused on gamification but still manages to put a heavy emphasis on the daily chores such as bringing out trash and cleaning the bathroom(Davis & Spearing, 2007). The

system utilizes avatars and a roleplaying system where every task has a reward for completion but also involves a small combat scenario between the avatar and



Figure 10: A screenshot of ChoreWars

monsters e.g. bed bugs. Various

projects can be created as parties for the avatar to join and participate in the completion of the tasks created either by procedural generation or through manually creating them as the leader of a party.

An element which produces a positive feedback is the levelling system as the user will easily strive to complete one more task to go up a level.

### 3. RedCrittter Tracker

The RedCrittter tracker is a regular management tool providing the team with an interface similar to a virtual Kanban board (RedCrittter Corp, 2011). These boards are utilized for most of the system, like searching for users and adding them to a segmented team or managing the sprint and the backlog from an Agile management perspective. Furthermore, the system contains badges which a user can earn for his team, placing him on the wall of fame feed granting him recognition amongst his teammates. Another aspect of the redcrittter tracker is the reward shop, which the team themselves can pick. These rewards can be bought with virtual coins collected through completing tasks. In the example given on their website, the team is capable of buying a lunch with the CEO, a paid day off and a nerfgun. Finally, every task has a timer which can be started at the click of a button and stopped just the same allowing for simple tracking of time it took to complete an assignment.

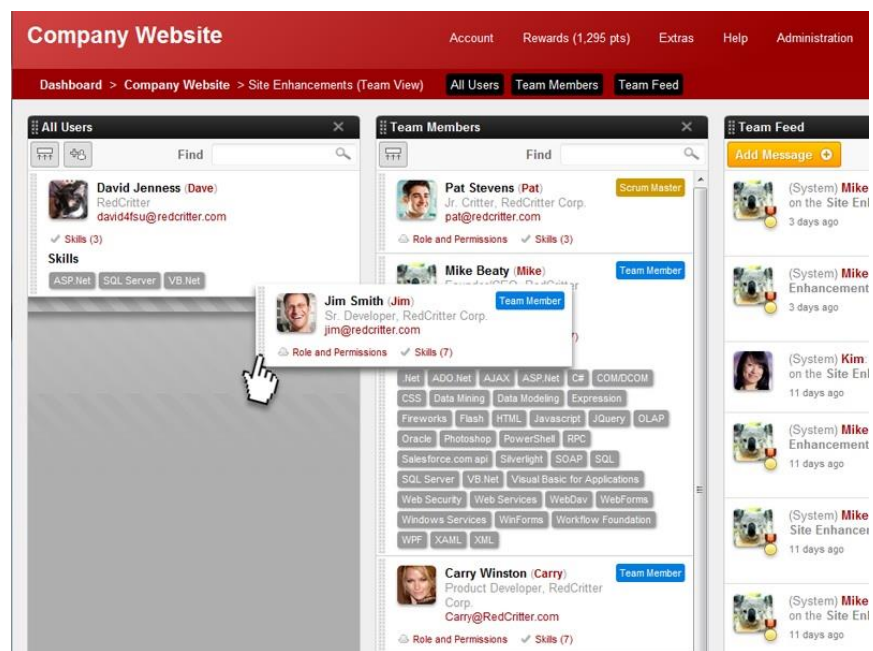


Figure 11: A screenshot from RedCrittter Tracker

### F. Management Application requirements analysis

In addition to having researched a couple of management applications available, a questionnaire was prepared with the aim of getting insight of specific requirements game developers had for a management application. The questionnaire revealed that most of the experienced developers agreed upon the implementation of a version control system into the management application in order to keep track on files and associated tasks. In addition, it was also discovered that most of the inexperienced

developers had little understanding of the topic in general, further proving the concern of young development teams from section V. For more information see Appendix 1.

## G. Conclusion

Based on section VII. A it was discovered that there are multiple phases within a game production each with their own goals and set of objectives. Additionally, it was investigated what each phase would be focused on and when each phase would be considered complete. This led to the indication that the most time consuming and thus the most uncertain phase would be the production phase described in section VII. A. 2. As such this would be the phase of interest from this point going forward.

Furthermore, it was clarified in section VII. A that various parts of a team does not necessarily have to be in the same phase at the same time and that the team composition was interconnected with the game concept as per VII. A. a). In addition, it was discovered in section VII. B that the team size could potentially vary greatly depending on the game concept as well, although the team size or economical limitations could, in reverse, help shape the game concept. Moreover, some team members could potentially be part of two departments at the same time and as such need to be updated on both, while still maintaining a clear overview.

Having gotten a better understanding of the production phases, the next point of interest would be how to engage the workers. Hence the focus became looking into work engagement and what being engaged in a job actually does to a person. It was discovered in section VII. D that being in a positive state of engagement during work will motivate employees to work harder and be more dedicated. Additionally, several objectives for how to produce such engagement were proposed. Finally, it was discovered in section VII. D. 1 that gamification is a tool which when applied can cause further engagement and likewise affect a person behavior. Thus it will be considered the primary source of inducing work engagement henceforth.

From investigating the production and the producers themselves to investigating the way the production would be organized, it was discovered in section VII. C that there were several types of management methodologies. However, not all would be equally fit for running a game production as a game production tends to involve a lot of changing requirements.

As such, the approach to utilizing these methodologies would be studied and reflected upon in regards to actual applications.

Hereby it was revealed, that combining management and gamification would have only been done a few times before, meaning that there would be a lack of gamified management applications as per VII. E.



## VIII. LITERATURE REVIEW

In this chapter the emphasis will be to investigate applications combining both gamification and management in one product. Hence the products from the VII. E chapter will be presented in more detail and discussed according to their structure as a software and management tool, their usage of gamification and limitations.

### A. ChoreWars

ChoreWars being a simplistic gamified housework application has put a large emphasis on the game elements of the system(Davis & Spearing, 2007). Hence it would be of importance to investigate the more detailed structure of this application, along with the actual usage of gamification and finally the limitations of the product.

#### 1. Structure

ChoreWars even though being a very simplistic system manages to utilize the best of both structural gamification and content gamification. As previously mentioned in section VII. E the user is asked to pick a name and image for the avatar the user will be using in the system. After having created the avatar, the user is asked to pick a few favorite tasks like cleaning the bathroom in order to randomly assign the user some starting attributes. Following the completion of the avatar, the user will be asked to either create a party or join an existing one. The party the user joins is essentially a project and can be customized by the creator of that party with unique tasks and rewards. Essentially the entire structure boils down to a customizable list of current tasks either being repeatable or one time usage. Another interesting aspect of ChoreWars is the ability to receive live feedback through RSS feeds. Finally, the system lets your party see who contributed the most based on completed quests and allows the users to scroll through the entire weeks updates of task completions.



## 2. Use of gamification

In terms of the usage of gamification, having a fantasy avatar is a strong metaphor of empowerment on the job. Additionally being able to level up the character and earn new titles based on task completion is another really great usage of structural gamification.

More so is the implementation of an avatar for every member of the team, since it could potentially engage the users through the usage of a strong



Figure 12: The use of avatars in ChoreWars

content gamification approach to

managing mundane tasks. Another interesting aspect of the task completion phase is the random encounter with procedurally generated enemies. It allows the party leader to customize each task to incorporate a random encounter based on an encounter percentage and an enemy list. However, one notion that seems to be a waste is the combat itself as the combat is only a matter of chance and losing a battle has no influence on anything other than some extra items. Concurrently, the system allows for random gold and item drops also being customized by the party leader, although it is stated in the help section of the website that the items and gold has no purpose in the application, but rather is a motivator to incorporate real world rewards much like those from section VII. E. 3 based on items used or money spent.

## 3. Limitations

For all the well-used gamification elements that ChoreWars manages to employ, the management system behind it is over-simplified leading to a very basic interface with extremely limited management options. Overall the method used to organize work in this application seems to focus more on the Waterfall model than an Agile model. Mainly the system lacks organization elements like prioritization of tasks, being assigned to multiple groups and/or projects and even tracking the progress of a single

task. Furthermore, the system does not allow the users to be assigned to a task, rather just pick one at random, meaning that potentially multiple people could be working on the same task. Finally, since this system is used through a website, being allowed to extend this application is not an option.

## **B. RedCritic tracker**

In comparison to ChorseWars, the RedCritic tracker is a more sophisticated take on gamifying management, as this tool is primarily a management application and only secondly uses the concept of gamification (RedCritic Corp, 2011). Even so, this tool incorporates the two concepts quite well and manages to maintain a lot of important features even though gamifying the management.

### **1. Structure**

In opposition to ChoreWars, RedCritic tracker only employs structural gamification as a means of engaging users with the management system. In the section VII. E it was mentioned that the main structure of RedCritic tracker is to use online virtual Kanban boards for assigning people to teams or moving tasks from in-progress to the done board. Additionally, the team can create multiple projects with their own separate sprints containing their own deadlines and Kanban boards for managing tasks. Furthermore, the system allows an individual to switch between task view, plan view and results view presenting the user with a simple and quick overview of the project. The plan view presents the user with a product backlog and a sprint backlog for easy managing tasks and moving them between sprints and backlogs. In addition, every task has both an owner and an approver who will be notified once they are assigned and throughout the duration of the task. Finally the entire system utilizes communication in every aspect possible as it incorporates live team updates, a messaging system and direct connection to the users E-mail making it a tool with a large emphasis on communication and structure.

### **2. Use of gamification**

The approach RedCritic has taken on the usage of structural gamification is quite simplistic as it mostly revolves around completing tasks and receiving points based on the difficulty of the task. However, the gloriousness of the system begins exposing itself once multiple users are completing tasks in order to acquire one of the remaining days off. The reward shop is completely customizable to fit the company's needs and thus allows the admin of the project to create new rewards, set their prize and how many are left meaning that potentially the admin can control the engagement of the users based on the rewards.

Another aspect RedCrittter uses is the badge system, allowing for some recognition amongst the colleagues for having earned a badge based on the completion time of a task or work done in the weekend. Lastly, the system gamifies the project completion rate by granting the users project points instead of showing a number.

### 3. Limitations

Though the RedCrittter tracker is quite advanced in terms of the overall structure, it takes a long time to move around the interface based on loading time between various subsites. Adding to that is the issue with a segmented interface containing information hidden between panels or on other subsites. This enforces the users to click back and forth in order to solve multiple elements at once though it is more flexible than the ChoreWars. Furthermore, the system asks the user upon creating a project if the project should follow a scrum model and implements the entire structure before the user is even allowed into the project overview. The main issue with this product is the lack of content gamification as the tasks, even though granting rewards upon completion, becomes quite mundane very easily. The entire system feels like a management tool with a gamification element attached to it. In addition, much like the ChoreWars system, this is accessed on a website, meaning that applying additional elements to this management system is not a possible solution.

## C. Types of gamification

On the gamified website(Marczewski, 2015), 47 different approaches to gamification elements are proposed in 8 different categories, namely being: General, Schedules, Socializer, Free spirit, Achiever, Philanthropists, Disruptor and Player. In the following section, each category will be explained and discussed in relation to a management tool.

### a) General

The general category involves elements reaching from designing a tutorial to bring the users up to speed, to the use of feedback systems for continuously having users returning. Additionally, the use of a narrative is proposed to engage the user on an emotional level as well as using a theme for the gamification in general, like a fantasy setting and every piece of information would be thematically appropriate to the theme.

There are also a couple of stressful factors proposed, such as the user of time pressure or for the user to be capable of losing virtual valuables, thus stressing the user to be aware of the game elements within certain time frames.

**b) Schedules**

Scheduling the game elements is another approach and possibly the most approachable as it revolves around rewarding the users' activity like their first time using the service. Additionally, the rewards could also be handled at random occasions or within specific time frames.

**c) Socializer**

This subject relates mostly to interaction between users and places a heavy emphasis on how different types of user interactions can either propose a purpose or bet users against each other for competitive engagement. In this category, teams are proposed as a way to encourage users to work together to solve a problem, which is the exact opposite of the previously mentioned competitive engagement. Furthermore, the aspect of using social media profiles to involve friends and family of the user is proposed to expand the network or attain some sort of status by having leaderboards and ranking systems. Another approach would be to apply social pressure to push a user's opinion in a certain direction.

**d) Free spirit**

The free spirit category has a huge emphasis on the user themselves, with the possibility to explore the systems and uncover new content for their specific profile. Additionally, it is proposed to allow for the user to create their own content or at least add their own touch with meaningful choices and personalized assets. Another approach would be to add Easter eggs for the users to discover as they use the application, possibly linking these with the rare content.

**e) Achiever**

This category relates to completionism and how overcoming specific elements will keep the users motivated as they will feel like they have achieved something. One achievement could be supplying certificates or further information based on the challenges overcome.

Another approach could be to give the users a set of challenges to accomplish or even a progression system utilizing levels to keep the users reminded of how far they have gotten. Systems like these could potentially help controlling a user's focus on the task they have been given.

**f) Philanthropists**

The philanthropist category is about helping other users and sharing information, items or virtual content. By having a user in a supportive role, the user will feel needed and can possibly mentor other users about the systems or the product itself. Another aspect mentioned within this category was the ability to give, share or trade items, as this could help the users build relationships and allow a user to help another user grow.

**g) Disruptor**

Disruption is a category focusing on bending or breaking the conventional limits of the systems the product utilizes, thus the goal is to allow the users to run wild in terms of the products systems. One such way is to allow the users to go beyond the original scope of the system or even expand upon it and improve the system. It is also mentioned to allow for the users to stay anonymous and let them express their opinions about the product. Finally it was mentioned that breaking the rules of the system could sometimes be a great way to let the users experience new elements and keep them invested in the product.

**h) Player**

The final category is mostly a gathering of different mechanics to be added for the users to enjoy themselves and their current progress. The first one mentioned was a progression system, as mentioned in an earlier section, this can be used to keep the users motivated to go one step further in the hopes of reaching that next level. Following the progression system, awarding items is mentioned as a reward for accomplishing specific tasks and encouraging the user to continue completing tasks. Once again, the ability to track the users ranking in comparison to others is proposed with the use of leaderboards or badges rewarded for completing specific tasks. Additionally the concept of applying a virtual economy to the products system is also emphasized as rewarding virtual coins for completing a task is a simple gimmick. Finally, the concept of using a game of chance in the system is proposed as it requires nothing from the user, but can potentially grant the user a large reward.

## **D. Gamifications effect on motivation**

As utilizing gamification for an increased motivation is the aim for this investigation, analyzing another experiment with somewhat parallel goals could be of significance. Hence this section will focus on inspecting the experiment of applying gamification with the aim of motivating senior users to interact with a system.

### **1. Goal**

Based on previous investigations about the relationship between gamification and positive experiences along with motivation, a new study was conducted (Wagner & Minge, 2015). The study aimed to analyze whether subjective enjoyment and motivation in a social game can be correlated for seniors and somewhat lessen the feeling of helplessness when dealing with technology. The experiment focused mainly on seniors playing a game with or without social aspects in order to determine if it had a positive influence on their perception of the experience.

### **2. Limitation**

The main limitation for their investigation was with the selection of software as they had chosen an online version of the game Yahtzee. Unfortunately, dealing only with a gamified communication system could potentially not have revealed the participants relation with any actual communication software such as skype. Even so, the study still managed to indicate a heightened motivation for the interaction with the system regardless.

### **3. Experimental setup**

The test was conducted with three groups where each test involved two participants. The first test had to play the Yahtzee game alone without anyone to interact with. The second group had to play the game knowingly against each other on a shared screen but without any interaction. The final group had to play the game like the second group, but with auditory and visual feedback from their opponent. The experiment was validated using questionnaires after the first game and again after the second. The questionnaires contained scales and questions relating to the participants emotions and perceived engagement.

For validating the participants' mood, a Self-Assessment Manikin valence scale was used as a 9 points scale. In addition, an IMI test was conducted to get an understanding of the participants' perceived motivational goals. This was done by asking the participants to indicate their agreement on a Likert scale with multiple questions. This in turn, indicated at certain emotions e.g. "I felt it was important for me to win the game" indicates dominance and "I was really enjoying the game" indicates enjoyment.

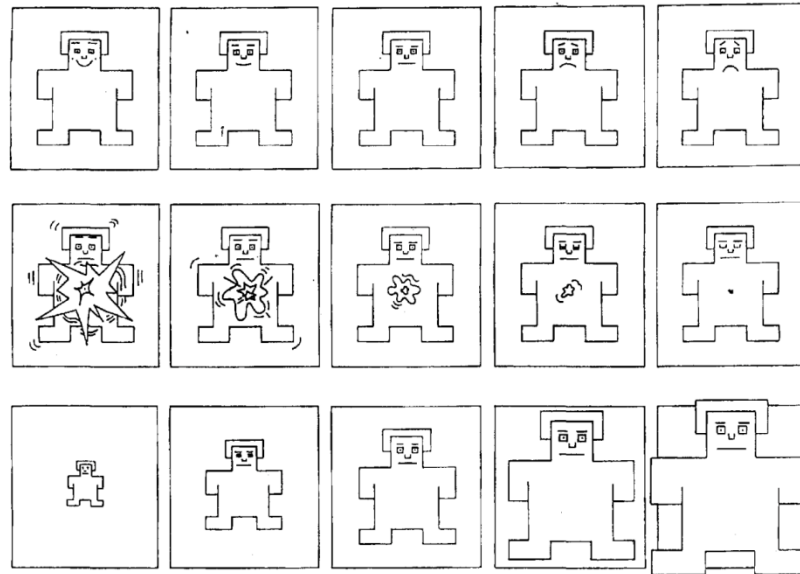


Figure 13: Arranged from top to bottom is the Valence, Arousal and Dominance scale

#### 4. Shortcomings

As for the approach to the experiment, the game Yahtzee was chosen possibly due to the already existing implementation fitting perfectly with the experimental setup. However, it stands to reason that utilizing a simple game, which most participants had prior experience with, could have biased the test as most participants would potentially have expectations of the games flow.

## IX. FINAL PROBLEM STATEMENT

With the knowledge and understanding of game productions phases and flows, it was decided to delimit the focus on game productions to their production phase as this was the most hectic and prolonged phase containing every issue prone to arise during a production VII. A. 2. As such the focus will be aimed mainly at making the production phase motivating and engaging. However, as game productions tend to change their plan during the process of development, this dictates the use of an Agile management model, preferably Scrum as it contains more requirements and will thus be easier for developing a workflow VII. C. 2. a). Additionally, the point of engagement will be through the implementation of gamification as gamification is a tool used for motivating and engaging users through a product VII. D. 1. This can either be done through the usage of structural or content gamification, although in order to utilize gamification the most effectively both should be applied. As for the management system to be tested, since none of the other products are modifiable it has been decided to develop a custom management system with a gamification element VIII. In addition, applications with the same objective, employing a flexible variation of a virtual Kanban board seems to be the most optimal approach as both Shotgun software and RedCritic Tracker utilizes this approach, easily achieving an approachable overview of the entire project or segment of such VIII. B. 1. In regards to the gamification experiment investigated, it would seem that utilizing communication between various parts of the team is another key to achieving engagement and motivation across a production VIII. D.

**“To what extent is it possible to utilize Gamification as a tool for motivating users and inducing a sense of teamwork across a game productions Production phase through a task tracking tool utilizing elements from Agile development?”**



## **X. EXPERIMENTAL DESIGN**

This chapter will aim at designing the experiment itself, both from a technical perspective as well as an experimental one. As such this section will focus mainly on narrowing the Final Problem Statement along with the research done in the investigation chapter to a list of requirements used to design the application. Once the requirements for the application have been establish, the experimental design will be discussed and decided upon.

### **A. Introduction**

With the scope of our investigation narrowed to focusing only on gamifications effect on a user's motivation and sense of teamwork through a game productions production phase managed through applied elements of Agile development, the focus of the study becomes evidently clearer. As such, the aim is to form a list of requirements used to determine the most important features to implement along with the less important and finally the features which would be of significance for the implementation, but not necessary for the product to function as intended for the test. Furthermore, this section will focus on the experimental design and how the testing scenario should be performed.

### **B. Application design**

This section will focus on the design of the application used to validate the final problem statement. As such, the primary focus will be on how to join an agile based management application with gamification elements without intervening in the process of task tracking, sprint and product management. Following, various topics and guidelines for gamification will be analyzed and utilized according to agile management methodology. This analyzation will provide an overview of the agile configuration and help in building the structure for the actual management application.

#### **1. Gamification design**

Following sections VII and VIII it becomes evident that the concept of gamification is of much use when utilized correctly. Furthermore, in VIII. C it was discovered that there exists multiple types of gamification elements depending on the approach. Since the aim of this project is focused on motivation, teamwork and planning two obvious choices would be scheduled gamification and

socialized gamification, although these do not exclude other types of gamification these will be the primary focus areas of the gamification design. Also to be noted is the importance of having the gamification element be visually distinguishable from the management system itself in order for the users to take note it(Lux, 2014). This relates to section VII. D. 2 as putting emphasis on the mechanics could potentially help guide the users focus towards the gamification actions. Furthermore, as stated in section VII. D. 1 in order to get the full potential from gamification it is of significance to use both structural and content gamification. As such, designing both a structural gamification element for the system along with a content gamification element would be the best approach.

**a) Structural gamification**

In section VII. D. 1, it was discovered that structural gamification would be used mainly to keep user coming back on a regular basis and that it uses progression, real-time feedback, incremental goals and rewards making it ideal for a task tracking process, specifically in a sprint. Additionally it fits well with the scheduled gamification element as both will have the user returning at specific intervals, such as upon task completion to see the incremented progress of the sprint.

**b) Content gamification**

Content gamification was another type of gamification which aims mostly on transforming the environment into a more engaging thematic substitute of the same system. This can be done through the usage of story elements, interactivity and characters as was the purpose of ChoreWars in section VIII. A. 2. As such it would be the most appropriate in conjunction with both the socialized, philanthropist and player gamification elements meaning that every user could potentially have a customizable avatar. Additionally, the segmented developer teams could be gamified adventure party's working together towards a completing a sprint. Furthermore, some users could be granted specific roles and functionalities to alter the teams' perception of their work. This could potentially help the asocial developers get slightly more integrated with the rest of the team based on their performance within the system as every user would be notified based on that specific users task completion. Finally it could alter the perception of the managers on a production and change the focus from being the guy delivering all the work, to the guys supporting the team on their journey.

## 2. Agile design

As it has now become evident that the application would be capable of encompassing both structural and content gamification elements, arranging the structure of the management application in accordance with section VII. C. 2 is the next step. Accordingly, the application should be able to handle multiple projects with each their own set of users, tasks, teams and sprints. This would allow users to change individual deadlines or even segment their entire team into two lesser sub teams. Furthermore, for the application to follow an agile approach, it should contain lists of tasks structured according to the user's preferences. Additionally, as some users might be part of multiple development fields, a user should be capable of participating in multiple teams. Finally, tasks should contain a time estimate based on their level of difficulty.

## C. Requirements

This section will primarily focus on listing the requirements separated into two categories based on their integration with the management system or the gamification element in accordance with the research done prior to this chapter. Additionally the requirements will be listed based on their area of implementation either being user interface, functionality or feedback.

### 1. Important features for the management application

This section holds the requirements for the management application and contains only two categories being the user interface and functionality

#### a) User interface

- Should be able to effortlessly identify elements of interest such as buttons, windows and images
- Should provide an easy overview of assigned tasks
- should provide an easy overview of projects
- should provide an easy overview of users
- should provide an easy overview of project status
- should provide an easy overview of relevant tasks
- should provide an easy overview of teams
- should have a login screen

- should have a personalized desktop
- should have a segment for user data, such as profile image, name and title

#### **b) Functionality**

- Should allow a user to be part of multiple teams
- Should allow a user to create new tasks
- Should allow a user to create projects
- Should allow a user to create additional teams
- Should allow a user to assign users to teams
- Should have a hierarchal structure with a project containing sprints, teams and users
- Should have a hierarchal structure with a sprint containing tasks
- Should be able to complete tasks
- Should be able to edit existing tasks, teams and users
- Should be able to move an object while dragging with the mouse
- Should show content information on mouse over

### **2. Important features for the Gamification element**

This section holds the requirements for the gamification element to be added to the previously described management system. This section contains three categories being the user interface, functionality and feedback

#### **a) User interface**

- Should contain a small frame used to visualize the gamification interpretation of the system
- Should be effortlessly identifiable as an element which is not an immediate part of the management system
- Should contain visuals of the team itself

#### **b) Functionality**

- Should work based on functionality from the management system
- Should allow the users to pick an avatar
- Should allow the users to collect points

### c) Feedback

- Should provide feedback based on task completion
- Should provide feedback on receiving coins
- Should provide feedback when a user joins the team
- Should provide feedback when a team completes their sprint tasks

## D. Prototyping

With the knowledge of the products requirements, developing a prototype for the finite design of the application is possible. Thus the focus of this chapter will be to develop a design based on the section X. C. Additionally, the application will be implemented in two prototype variations before implementing all of the features from section X. C.

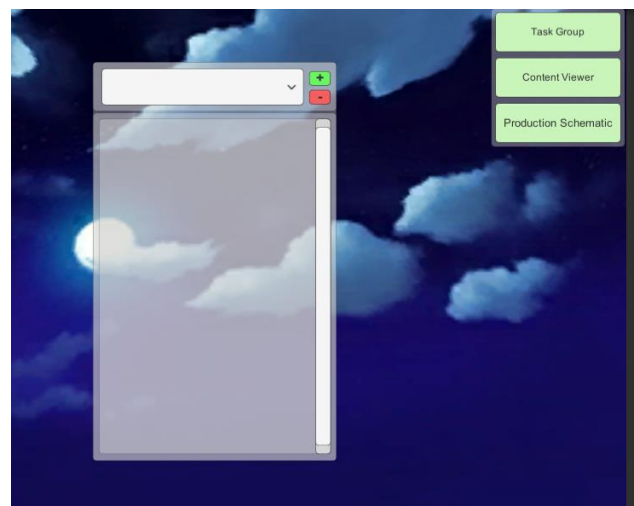
### 1. Prototype design

The design was developed mainly based on the design principles from section RedCritic, but with the addition of the gamification elements inspired by ChoreWars. The led to an interesting setup with a gamification frame used to control the gamification of the users characters. Additionally, the design was focused on teamwork, so the gamification frame was used to portray every character in the user's current team. Since ChoreWars used characters and enemies in a combat scenario for content gamification purposes, but to little success, the application will take another approach. Thus this application will be designed based on a combat scenario between each team and the sprint, being the enemy. The health value of the monster will be proportional to the amount of estimated work left during a sprint in order to potentially engage the users across the team based on team effort. Finally, these actions will be provided through a live stream showing off the other users' task completion as attacks against the enemy and thus provide feedback for the sprint progress. Furthermore this could reward the users' upon a successful sprint completion before a deadline and thus motivate the users' to finish work within a certain deadline to achieve more. The concept of gamifying task completion with the usage of structural gamification will hopefully influence the behavior of the users and have them report task completions more often. As for the management part of the system, it would seem that segmenting information is the optimal approach according to section VIII. B. 1. As such, it would be beneficial to have a dashboard used primarily for workers and one used primarily to manage the team.

Additionally, providing a clear overview of both profile data and projects is of significance so that the user always knows which project is worked on. As such there seems to be a need for a segment containing previously mentioned information in the system, and a field which could contain multiple tabs for showing project information. Another aspect of the system is the issue of showing content, however as a larger part of the screen seems to be needed for both the gamification segment and the projects and data segment, the most optimal solution would be to have the interface utilizing a drag and drop system for moving content around, much like the system RedCitter uses for its tasks. Additionally, this functionality could be extended to frames containing various types of information such as lists of completed work, assigned tasks or even users on a specific team.

## 2. Unity UI system

The decision was made to implement the application using the Unity3D game engine, as distributing the system for testing would be simple with the flexibility of the Unity build settings. Based on the previous section X. D. 1 the system would need to feature a flexible yet easy to use interface while still providing the necessary functionality of other management tools. This would be done by segmenting all the information into one out of three windows containing either draggable content, a production schematic or a content visualizer. As the content visualizers would update with the newest content produced by the team, it would add to the users feeling of working in a team rather than feeling secluded from a team working in another location,



thus making this an essential element of the product.

Figure 14: Unity UI Prototype

The issue was to get access to the content of another Unity project as Unity does not contain the functionality for importing assets in a standalone build. This meant that in order to get visualization for 3D graphics, the system would need to contain a custom importer for each file type.

Adding to the difficulty and time consumption of that task, was the fact that in order to import fbx type files which is the default file type for Autodesk models, the system would need to use a library in C++ possibly meaning running another external process.

Instead of writing custom importers it was decided to extend the Unity editor to implement the management system. This approach seemed to be the ideal solution as most designers and programmers would get feedback from the system in their working environment.

### 3. Unity editor system

Having developed most of the basic functionality for the system, integrating the system into the editor could only have limited problems. Since the unity engine handle the implementation of the assets this way, creating a content viewer would be seemingly easier. The Unity editor was customized to have an additional dropdown menu which would be capable of performing various actions such as opening the main management window. The impressive feature using this approach was the ability to dock the window inside of the editor, allowing for a fast workflow for designers and programmers.

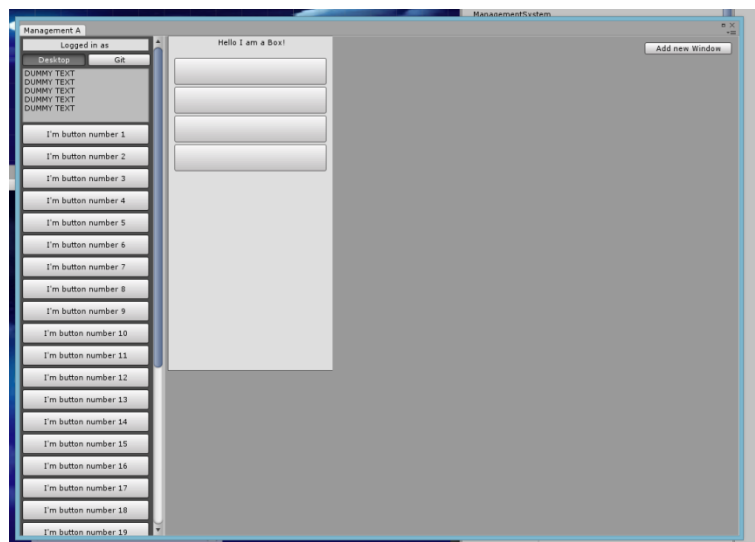


Figure 15: Unity Editor Prototype

There were a couple of issues with this approach though. A main issue was the lack of flexibility as every single menu would have to be organized within one out of three different types of UI scripting meaning that managing multiple UI elements for creating the interface became quite messy and disorganized. Additionally, an interesting bug was encountered during the development of this setup when utilizing a struct for draggable windows in the unity editor.

The draggable window would reset back to its original position upon release, which seemed to be a problem related to the influence of a script extending the Unity editor.

A possible solution to this was to move it into a separate script and then create an instance of that object within the management system script. However, upon having spent a vast majority of the implementation time fixing layouts, it was decided to go back to working with the Unity UI system for implementing the product.

## **E. Usability testing**

As the complexity of the developed product increased proportionally to the implementation of the management system itself, the entire system underwent a limited number of usability tests. These tests were mainly used to find faults with the system, check the literacy of the system and finally to investigate if the system had the desired effect on the participants. The following segments will describe the approach to the testing scenario followed by the observations and what influence it had on the project.

### **1. Approach**

As the goal of this section was to analyze the user-friendliness of the developed prototype a list of tasks within the system was agreed upon before starting the test. During the test, the participant would be given full control of the interface with most of the features implemented. The participant was informed by a test conductor to state his mind and explain as many of his thoughts in regards to the system as possible. Furthermore, the conductor would state the task for the participant to complete and observe as the test participant would try to complete the said task. The conductor's job would be to take notes of the participants' actions within the system and observe whether or not the test participant could complete the task at hand. After all the tasks had been attempted, the conductor would ask the test participants to express their thoughts on the experience and the system itself.

### **2. Observation**

While observing the usability tests, it was discovered that the systems controls were not all utilized as some users never discovered the ability to delete objects or close menus. Although this could prove to be quite a problem for testing the products functionalities, it should have no influence on the



gamification element of the system. Generally the participants were quite adept at learning how to use the interface and amongst all the usability tests completed, the lowest task completion proved to be above 90%. Furthermore, with the exception of a single participant all the testers understood the gamification element as being a visualization of the production. Additionally, it would seem that the users with prior knowledge of management had a better understanding of the systems functionalities than those without.

### 3. Consideration

Based on the usability tests conducted, the product proved to be in need of a tutorial to ensure that all users would be equally skilled at the application. Additionally, the most important aspect of the system would be for the users to have an understanding of agile management, preferably scrum. As such, a guide for the system was introduced to the application to ensure that every user would have an equal understanding of scrum and the functionalities of the system. Moreover, numerous system failures were discovered through the usability tests. However, each failure was properly noted and resolved before the final test was initiated.

## F. Experimental Design

Having developed and tested the design of the application, it became evidently significant to investigate how to test the hypothesis based on the implementation of the application. As such, this chapter will focus on the design of the experiment itself.

### 1. Online experiment

The aim of the first experiment was to find a correlation between the gamification elements and the test participants' perceived sense of teamwork and motivation. As such, a between groups test was conducted with the aim of receiving a bare minimum of 20 participants in each group. The test was conducted using an identical questionnaire for each group which contained a Self-Assessment Manikin valence and arousal scale in order to get an understanding of the participants' perceived mood and motivation during the test (Bradley & Lang, 1994). In addition, an IMI-C was used in order to get an understanding of the relationship the user had with the virtual coworkers (Kiesler & Schmidt, 2011; McAuley, Duncan, & Tammen, n.d.). The application was distributed online on social medias with game development groups or societies (e.g. facebook). The test would be available for a couple of days

although a strict time limit was never decided upon. In opposition to section VII. F acquiring data for this test was slightly slow as when the test had been available for a few days, only a handful of people had actually taken their time to test. To get an idea of how to proceed most effectively, the few gathered test results were compared and it was discovered that neither the control group nor the test group had had any understanding of the system. Neither had the control nor the test group noticed the fact that the system actually contained virtual coworkers and thus rendered both the data from the SAM and the IMI-C completely irrelevant. In addition, many participants posted comments stating their frustration with the management system in general and how they had ended the test earlier than anticipated. As such, it would seem apparent that another more structured test would need to be conducted with the aim of directing the participants' attention towards the gamification elements of the system. For more information see Appendix 2

## 2. Focus group

As the previous test had returned no indications of any kind, the following test would need to be more controlled by a facilitator. As such, it was considered whether it would be the best approach to test the hypothesis using focus group interviews or a Wizard of Oz test (Green, Wei-Haas, Orqvirmtion Nan Address, & N, 1985). However, due to the limitations of both the implementations state and planning it wouldn't be possible to conduct a Wizard of Oz test. Thus it was determined to conduct a focus group interview with a number of game development teams. Multiple people from various teams, ranging from newly formed to multiple years of experience, were contacted in order to get as many interviewees as possible for the focus group interview. The interview took place in a small secluded room with a couple of chairs, a table and a laptop for running the application. The total test would take about 30 minutes where the structure would be for the conductor to ask a question followed by the interviewees answering for a certain duration followed by another question. Additionally it was the conductors' responsibility to ensure that the interviewees would stay on track rather than start speaking about matters not related to the investigation. The process would start by having the conductor asking questions related to the interviewees current management system. This was followed by an introduction to the application and further questions based around how the application would affect the teams' engagement with work. During the interview, the entire conversation was recorded using the laptop in order to further analyze at a later point in time. In conclusion this method of approach would prove much more useful because of the explanation of the product.

## G. Summary

During this chapter various subjects have been analyzed and reviewed in order to get a more thorough understanding of the final problem statement:

**“To what extent is it possible to utilize Gamification as a tool for motivating users and inducing a sense of teamwork across a game productions Production phase through a task tracking tool utilizing elements from Agile development?”**

As such, the subject of the application itself was discussed both as a management tool and as means of gamification. Furthermore the concepts of structural and content gamification were discussed with the aim of arriving at several design requirements for the further development of the application. This led to a list of requirements for features to be implemented in the management application in order to engage the users most effectively. In addition, a couple of approaches to the management systems implementation were investigated as the requirements dictated certain functionalities only achievable with a specific method. Furthermore, several usability tests were conducted in order to get an understanding of the literacy of the basic interactions within the management application. This was mainly to ensure that the environment would not pose a threat to the testing results. However, based on the usability tests it was discovered that there was indeed a need for a tutorial for first time users of the system in order to avoid bias. Moreover, the primary test to be conducted would prove itself to be inadequate even though the usage of SAMs and IMI-C scales. As such, it was decided to do a focus group interview instead with a pre- and post-application discussion.

## **XI. IMPLEMENTATION**

In this chapter the implementation will be described according to approach, software usage and technical difficulties. As the implementation was completed utilizing a number of pre implementations for testing technical accordance, each step will be described based on the goal of the implementation, the conflicts, the solution and the actual implementation.

### **A. Introduction**

This chapter will describe the final implementation based on the technical design, the conflicts and their solutions. In addition, it should be noted that this chapter only contains a superficial explanation of the implementation as the prototype system is comprised of more than 40 scripts and a complex UI GameObject structure utilizing the Unity engine. For additional insight in the development of the product, see Appendix 4 and 5.

### **B. Pre implementations**

Before developing the final product for testing the hypothesis, several lesser features had to be considered in terms of possible solutions. As such, each of the following sections will propose a technical issue, along with possible solutions and finally complications in regards to the rest of the system.

#### **1. Git implementation**

Based on section VII. F one feature which would be inevitable for the system was version control system integration. However, utilizing version control is often done through a UI or by typing commands into a console.

This meant that the system would have to get access to another piece of software and communicate information through that connection. This caused a lot of problems as it required knowledge of both git, Unity, C# and computer operative systems. The main problem was to find a way to open a communication channel with the git server for sending and receiving data. The solution was to implement a system which could open a new computer process running git with a set of commands and

then streaming the data out of the git process and into the unity environment. The only issue with this solution would be the need to open and close and new process every time information had to be sent or retrieved.

In order to utilize this for reading the data correctly, an additional functionality had to be implemented, namely being the functionality to decode the received data. When this system was initially used, the data gotten was encoded with color codes used by the git process to distinguish between elements of the data, such as username, time of commit, branching structure and more. This issue was fixed by converting the encoded color data to a color code unity could utilize for printing the information.

## C. Application implementation

With the knowledge of both the Unity UI system and Unity editor scripting, the final iteration of the implementation proceeded. Being an evidently large implementation, this section will focus on lesser parts of the implementation and string together the overall understanding of the management system before applying the gamification elements to finalize the development.

### 1. Master script

The first and foremost element stringing together the entire system is a script inheriting from MonoBehaviour, named ManagementSystem, this script was mainly used to control the interface, both additionally worked as a pier between the UI interface and the entire underlying management structure. Being a key element in controlling the interfaces, this script contained public variables for various menus, buttons and graphics as many elements would need to be instantiated at runtime.

```

64 private User activeUser;
65
66 public GameObject subtasksPanel;
67
68 public enum MenuState{
69     NotLoggedIn,
70     Desktop = 1,
71     Team = 2,
72     Git = 3
73 }
74
75 public static MenuState currentState = MenuState.NotLoggedIn;
76
77 public Text totalTasksText;
78
79 // Use this for initialization
80 void Start () {
81     projectsPanel = GameObject.FindGameObjectWithTag ("Projects");
82     usersPanel = GameObject.FindGameObjectWithTag ("Users");
83
84     SetState (1);
85
86     allUsers = new UserGroup ("Company");
87
88     AddNewProject ("Project Testaburger", new System.DateTime(2016,5,26), 7);
89
90     selectedProject = allProjects [0];
91     Debug.Log(selectedProject.GetRemainingPrintPercentage (new System.DateTime(2016,5,26)));
92     //selectedProject.gameObject.GetComponent<Image> ().color = new Color (0.5f, 1, 0.5f, 1);
93
94     canvas = GameObject.FindGameObjectWithTag ("Canvas");
95     myCamera = Camera.main;
96
97     GameObject temp = Instantiate (TaskWindow, Vector3.zero, Quaternion.identity) as GameObject;
98     temp.GetComponentInChildren<ContentController> ().SetTaskGroup (selectedProject.taskGroups[0]);
99     temp.transform.SetParent (canvas.transform.GetChild ((int)currentState));
100     temp.transform.localPosition = new Vector3 (400, -5, 0);
101     temp.GetComponentInChildren<DragSlot> ().DropdownValue = 2;
102
103     GameObject temp2 = Instantiate (TaskWindow, Vector3.zero, Quaternion.identity) as GameObject;
104     temp2.GetComponentInChildren<ContentController> ().SetTaskGroup (selectedProject.taskGroups[0]);
105     temp2.transform.SetParent (canvas.transform.GetChild ((int)currentState));
106     temp2.transform.localPosition = new Vector3 (5, -5, 0);
107     temp2.GetComponentInChildren<DragSlot> ().DropdownValue = 1;
108
109 }
110
111 public void AddTaskWindow(){
112     GameObject temp = Instantiate (TaskWindow, Vector3.zero, Quaternion.identity) as GameObject;
113     temp.GetComponentInChildren<ContentController> ().SetTaskGroup (selectedProject.taskGroups[0]);
114     temp.transform.SetParent (canvas.transform.GetChild ((int)currentState));
115     temp.transform.localPosition = new Vector3 (5, -5, 0);
116 }
117
118 }
119
120 public TaskGroup FindMyTaskGroup(Dropdown myDropDown){
121     return selectedProject.taskGroups [myDropDown.value];
122 }
123
124 public UserGroup FindMyUserGroup(Dropdown myDropDown){
125     return selectedProject.userGroups [myDropDown.value];
126 }

```

Figure 16: A screenshot of the ManagementSystem script

Furthermore, as the ManagementSystem script would be the one initiating the entire system, this script also contained functionality for creating a new project along with setting multiple values through public functions.

## 2. Core structure

The main structure of the application would contain a project, with a list of users, a list of teams, a list of grouped tasks, a start and end date, the length of sprints in days and finally a project name. Each team would consist of users from the projects user list and have at least a one task group associated with it. This meant that once a user would be added to the team, that user would be able to get notifications on tasks within associated task groups (e.g. a user is added to the 3D team and now receives tasks with both 3D modelling and 3D animation). Additionally, every task group would contain its own list of tasks with the option of moving, removing or adding additional tasks. As a single task might exceed the largest time estimate, each task was implemented to support subtasks. This was mainly to keep task management organized and ensure a better overview of the process. Each subtask consisted of a description, a time estimate and a boolean value determining if the sub step had been completed or not.

## 3. Draggable content

One of the requirements was for the interface to utilize a drag and drop functionality to simplify the interaction with the system. As such, Unitys Eventsystem was used in order to detect when the mouse buttons would be pressed and released along with the movement of the mouse. Furthermore, a static variable was used to keep track of the currently dragged GameObject. This ensured that there would always only be one object actively being dragged. Additionally, another script was created with the purpose of being a container for dragged objects, this could for instance be a window used to hold task objects or a menu used to hold windows.

## 4. UI system

Having created the basic functionality explained in the prototyping section X. D, the rest of the system was built on top of that drag and drop functionality. However, setting up the entire menu and making a more visually distinguishable interface was another of the requirements. As such, the three menus mentioned in section X. D. 2 would be redesigned to have distinguishable visuals and functionality to complement their use cases. Another element in the UI system which needed a definite solution would

be the anchor points used for scaling the interface, as creating a scalable UI with a dynamic UI system could have proved quite troublesome although in a finite product this would have been the standard. Furthermore, in numerous arrangements, the UI would have to follow a specific layout or ignore it. Hence the use of layout elements and layout groups was highly useful in order to organize every element of the interface.

## 5. Prefabs

Yet another important element to the system would be that of prefabs as close to every single element used within the interface of the system is based on prefabs. In unity, it is possible to create prefabricated GameObjects which can be instantiated at numerous occasions both while working in the editor but also during testing of the application. Since most of the menus had similar functionalities, it proved possible to utilize the same objects for task groups and teams. Similarly, users, projects and tasks would be created from the same objects as with the previous group, they shared most of the basic functionality.

## 6. Gamification

Having described most of the implementation for the management system, the only element which remains is the gamification elements and their integration with the system. From the beginning of the implementation, the main focus was to establish a management tool comparable to those used by most professionals. As such a huge emphasis was put on developing the actual management system. In addition, the gamification tool would preferably be a plug and play implementation to the entire system



Figure 17: The Gamification element



as previously stated, it should avoid interfering with the management system itself, but rather add to the perceived engagement of the system. This would also prove easier to manage when developing both a control test and a hypothesis test as both would need to use the same interface, only with the addition of the gamification elements.

As such, the entire gamification element was created as an extra panel in the system, which could be enabled or disabled based on input. The entire logic was created in a separate Gamification script would be the controller of the gamification process in the application by communicating with the ManagementSystem script. Furthermore, in order to give the participants the illusion of working with a team, a couple of timers were implemented for having users being added to the participant's team during their testing session. Additionally, multiple tasks could potentially be completed by the participant's virtual teammates in order to give the participant the impression that the other users were actually working on the production.

## 7. Animation setup

Finally, in regards to the gamification simulation numerous animations would be needed in order to make the gamification element believable. As such, a simple but clever animation handling system was created by developing a public enumeration with all possible animation states. The animation handler had one public method which took a value from the public enumeration and type casted the enumeration to a string to pass into the animators trigger system. Every trigger within the animator would then be named according to the enumeration and call the associated animation before returning to the idle animation. This setup ensured that no matter which character, either user or enemy, they would all contain the exact same animations and they would all be called the same way. Furthermore, an editor script was created for making bug finding of the character animations easier to find. The editor script created a button in the inspector for each animation state, meaning that by starting the application and clicking a button, the specific animation associated with the button would be played. Finally, for the sprint win sequence, additional scripts were created to make coins spin and rotate along with fading the enemy upon defeat.



## **XII. FINDINGS**

This chapter will focus on the results of the focus group interview and try to make sense of what was mentioned during the interview by various interviewees both in the first part of the interview as well as the last part.

### **A. Pre-application**

The aim of the pre-application interview was to gain an understanding of the management structure the team used on a daily basis. As such several questions were asked to get a better understanding of the management system, their approach and possible issues and/or solutions they had encountered. For more information on this test, see Appendix 3.

#### **1. How do you plan your productions?**

The group from the focus interview were using weekly meetings most of all, with the addition of verbal and written communication. They put a lot of emphasis on how important it was for them to communicate and how much of an issue it had been for the team to have a single person split from the rest. Additionally it was mentioned that they used a google sheets setup for their task tracking and how limited they tried to keep it in terms of task details. One of the production managers added how much of an impact speed had had on the tool as it had been quite slow previously, but had now been optimized for faster tracking. In addition, it was mentioned by one of the content producers that he had started using the tool immediately after the update.

For a while the team focused on tasks and how people sometimes forget to complete their tasks in the tool even though their work was complete. Furthermore, it was mentioned that for a content producer the most important feature would be the ability to quickly login, check updates and work that had to be fixed and then logout again. Additionally, the content producer put a larger emphasis on the importance of seeing the progress of the game rather than being kept in the blind.

## **2. What are the largest issues with that process?**

When asked about the largest issues on their production, they immediately answered version control problems, such as data suddenly missing or having to revert and fix the same error again and again. Another issue was the lack of communication as mentioned previously, with people separated from the team.

Another issue was the focus of the meetings as their meetings were often planned for two hours but typically took six. Finally, it was mentioned by the content producer that there was a problem with people having multiple roles, as they would have to check multiple tabs in order to find all their tasks.

## **3. What are the most essential features for a management tool?**

The entire team seemed to agree that the first and foremost priorities would be speed and accessibility as lacking either would make the task completion process troublesome and ultimately lead to people not using the system. Another point they had was with how some applications tend to have multiple layers of content or tabs in order to perform a single action. To them, having a simple interface with only the necessary information was a priority.

## **B. Post-application**

The aim of the post-application interview was to gain an understanding of the motivation to use the developed product. As such, several questions were asked to get a better understanding of what could have influenced their opinions and if there could have possibly been an answer to the final problem statement.

### **1. Could gamifying management tools potentially engage users more?**

When interviewing regarding the product, it would seem that there was a uniform agreement that this could potentially motivate users to utilize the management tool more frequently. One of the production managers' arguments was: "All game developers play games right?" Another production manager mentioned that as long as it had no interference on the management itself it would be a welcome addition to the management tool. The content producer seemed overly fascinated by the idea and started coming up with further game design elements for visualizing specific fields of competences as additional enemies. Additionally it was mentioned that it was a fun way of visualizing the production

management and how it would probably motivate people for reporting task completion. Furthermore, the participants referred to the concept of juice it or lose it as mentioned in section VII. D. 2 and claimed that the system would work even better had more juice been applied. A final note from a production manager was however that the tool seemed too large for his team, but he could potentially see it used for larger production teams.

### **C. Usability testing observations**

During the usability tests described in section X. E there were some additional comments about the gamification element of the prototype. Mainly, the participants were amazed by the interactions between the management system and the gamification elements and claimed that such a system would engage coworkers to pull their share of the work. Additionally, some participants were really invested in the customization element of the gamification system, where the participants could pick between multiple avatars.

### XIII. DISCUSSION

During this investigation the aim was to discover if gamification could be used to motivate users and induce teamwork during a game productions production phase. Currently, a final conclusion has yet to be decided upon as still many questions remain. One of the problems addressed was rooted in the team compositions of game productions, although it would seem it is not the composition itself but rather managing multidisciplinary individuals' tasks. Likewise another problem encountered would be that of not using a management tool or forgetting to mark tasks as complete in a task tracking tool. This problem though, might be possible to at least decrease the frequency of by adding a gamification element to the task completion process of a task tracking tool. Similarly, was the problem of having a multidisciplinary team with various interpretations of an interface possible to solve through the implementation of a tutorial for a product. In addition, keeping the team motivated and engaged in the development is a task which is hard to manage, however by the use of gamification it would seem that the struggle can potentially be lessened slightly.

Although the data acquired was less than expected, the findings of this investigation are in alignment with those of other literature reviewed during this investigation. As it was stated in section XII. B it would seem that game developers are expecting that the implementation of gamification could potentially strengthen the motivation of users, thus potentially changing their behavior regarding the reporting of completion of tasks. Which, when compared to the literature reviewed in this investigation is in accordance. In regards to the induction of teamwork, it would seem that the question remains yet to be answered. Additionally, allowing the users to distinguish themselves through the use of customizable avatars based on content gamification seems to be a welcome change and it can be speculated according to section VIII. C that this might pose either a feeling of collaboration or a feeling of competition.

One result to be mindful of is the impact of speed and efficiency for a management system as most of the research done throughout this investigation indicated that it was one of the deciding factors when choosing a management system. This indicates that when utilizing gamification as an element of motivation in a management tool it becomes of outmost significance that it does not interfere with the workflow of the management tool.

Finally, these outcomes suggest that there could potentially be a use for gamification in management as engaging and motivating employees to work harder means more content in less time, potentially saving companies some money on productions which could hypothetically mean an avoid bankruptcy.

## XIV. CONCLUSION

The final problem statement was as follows:

**“To what extent is it possible to utilize Gamification as a tool for motivating users and inducing a sense of teamwork across a game productions Production phase through a task tracking tool utilizing elements from Agile development?”**

It might be concluded that there was a tendency for the agile management application developed to motivate users to report their tasks completion. This can be indicated based on multiple verbal responses during various interviews. However that only answers half the question as the results in regards to inducing teamwork were too indecisive for a conclusion. As such, giving a concluding answer to the final problem statement would require more thorough testing during an actual game production with a non-virtual team.

## **XV. FUTURE WORK**

The final problem statement remains inconclusive, although there seems to be potential for motivating users to work harder by the usage of gamification in a management application. This might be further uncovered through a test using a real world scenario rather than a simulated one. This means first and foremost further developing the application to enable numerous game development teams to use it for their production phase. In technical terms, this would require for the product to synchronize data across multiple instances of the application, in addition to more graphical assets for the gamification elements. Additionally, customizable avatars should also be developed in order for the user to distinguish themselves. Preferably, this test should utilize a focus group interview per day and have the team discuss their perception of other users' completion rate. The teams should still be separated into a control and a test group in order to determine the perceived effect. Additionally, using an IMI-C to measure the perceived relationship with the coworkers could prove beneficial in order to determine any changes between the two groups.

A secondary approach could be to conduct a Wizard of Oz experiment and thus making it seem as if there was interaction with the team. This approach would only require slight modifications to the product as well, but would still be a simulation and thus might also uncover a bare minimum of indications.

## XVI. BIBLIOGRAPHY

- Bradley, M. M., & Lang, P. J. (1994). MEASURING EMOTION: THE SELF-ASSESSMENT MANIKIN AND THE SEMANTIC DIFFERENTIAL. *I. B&w Thu. & Exp. Psvchrar*, 25, 49–59.
- Bunchball inc. (2016). What are Game Mechanics? Retrieved May 23, 2016, from <http://www.bunchball.com/gamification/game-mechanics>
- Burger, R. (2015). The Ultimate Guide to Project Management Gamification. Retrieved May 23, 2016, from <http://blog.capterra.com/the-ultimate-guide-to-project-management-gamification/>
- Chan, K. (2013). Waterfall vs. Agile. Retrieved May 23, 2016, from <http://www.onedesk.com/waterfall-vs-agile/#comment-10104>
- Chandler, H. M. (2009). *The game production handbook*. Infinity Science Press.
- Chapple, C. (2014). 8 Production pipeline tools to keep your game on track. Retrieved May 23, 2016, from <http://www.develop-online.net/tools-and-tech/8-production-pipeline-tools-to-keep-your-game-on-track/0198114>
- Davis, K., & Spearing, C. J. (2007). ChoreWars. Retrieved from <http://www.chorewars.com/>
- Doulin, A. (2010). Building A Strong Indie Game Development Team. Retrieved May 23, 2016, from [http://www.gamasutra.com/blogs/AlistairDoulin/20100107/86323/Building\\_A\\_Strong\\_Indie\\_Game\\_Development\\_Team.php](http://www.gamasutra.com/blogs/AlistairDoulin/20100107/86323/Building_A_Strong_Indie_Game_Development_Team.php)
- Galanakis, R. (2014). Agile Game Development is Hard. Retrieved May 23, 2016, from [http://www.gamasutra.com/blogs/RobGalanakis/20140219/211185/Agile\\_Game\\_Development\\_is\\_Hard.php](http://www.gamasutra.com/blogs/RobGalanakis/20140219/211185/Agile_Game_Development_is_Hard.php)
- Green, P., Wei-Haas, L., Orqvairmtion Nan Address, P., & N, S. A. (1985). *The Wizard of Oz: A Tool for Rapid Development of User Interfaces*. Michigan. Retrieved from <https://deepblue.lib.umich.edu/bitstream/handle/2027.42/174/71952.0001.001.pdf?sequence=2>
- Groh, F. (2012). Gamification: State of the Art Definition and Utilization. *Research Trends in Media Informatics*, 4, 8.



- Heller, N. (2014). Games developed entirely by one person. Retrieved May 23, 2016, from <http://www.gamesradar.com/10-games-developed-entirely-one-person/>
- Jonasson, M., & Purho, P. (2012). *Juice it or lose it - a talk by Martin Jonasson & Petri Purho*. YouTube. Retrieved from <https://www.youtube.com/watch?v=FyoaCDmgnxg>
- Kapp, K. M. (2014). *The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education* (1st ed.). San Francisco: Wiley.
- Kapp, K. M., Blair, L., & Mesch, R. (2014). *The Gamification of Learning and Instruction Fieldbook: Ideas into Practice* (1st ed.). San Francisco: Wiley.
- Kiesler, D. J., & Schmidt, J. A. (2011). *The Impact Message Inventory - Circumplex (IMI-C) Manual*. Retrieved from [http://www.safranlab.net/uploads/7/6/4/6/7646935/impact\\_message\\_inventory\\_manual.pdf](http://www.safranlab.net/uploads/7/6/4/6/7646935/impact_message_inventory_manual.pdf)
- LeanKit Inc. (2015). What is a Kanban board? Retrieved May 23, 2016, from <http://leankit.com/learn/kanban/kanban-board/>
- Lux. (2014). Visual Hierarchy for Game Developers: A Practical Guide to Making Important Stuff Seem Important. Retrieved May 23, 2016, from [http://purplepwny.com/blog/visual\\_hierarchy\\_for\\_game\\_developers\\_a\\_practical\\_guide\\_to\\_making\\_important\\_stuff\\_seem\\_important.html](http://purplepwny.com/blog/visual_hierarchy_for_game_developers_a_practical_guide_to_making_important_stuff_seem_important.html)
- Marczewski, A. (2015). 47 Gamification elements, mechanics and ideas. Retrieved May 25, 2016, from <http://www.gamified.uk/2015/02/04/47-gamification-elements-mechanics-and-ideas/>
- McAuley, E., Duncan, T., & Tammen, V. V. (n.d.). Psychometric Properties of the Intrinsic Motivation Inventory in a Competitive Sport Setting: A Confirmatory Factor Analysis. *Research Quarterly for Exercise and Sport*, 60(1), 11. Retrieved from [https://selfdeterminationtheory.org/SDT/documents/1989\\_McAuleyDuncanandTammen\\_PsychometricPropertiesofIMIinSport.pdf](https://selfdeterminationtheory.org/SDT/documents/1989_McAuleyDuncanandTammen_PsychometricPropertiesofIMIinSport.pdf)
- RedCritic Corp. (2011). RedCritic Tracker. Dallas: RedCritic Corp. Retrieved from <https://www.redcritictracker.com/>

- Schaufeli, W. (2011). Work Engagement: What do we know? *Int. OHP Workshop*. Timisoara. Retrieved from [http://www.psihologietm.ro/OHPworkshop/schaufeli\\_work\\_engagement\\_1.pdf](http://www.psihologietm.ro/OHPworkshop/schaufeli_work_engagement_1.pdf)
- Shotgun Software Inc. (2015). Shotgun. Shotgun Software Inc. Retrieved from <https://www.shotgunsoftware.com/>
- Skou, D. K. (2012). *Gamification Et teoretisk orienteret speciale*. Aalborg. Retrieved from [http://projekter.aau.dk/projekter/files/63422058/Gamification\\_et\\_teoretisk\\_orienteret\\_speciale.pdf](http://projekter.aau.dk/projekter/files/63422058/Gamification_et_teoretisk_orienteret_speciale.pdf)
- Stanley, R. (2014). Top 25 Best Examples of Gamification in Business. Retrieved May 23, 2016, from <http://blogs.clicksoftware.com/index/top-25-best-examples-of-gamification-in-business/>
- Velasquez, L. (2014). Employee Engagement vs Satisfaction vs Motivation. Retrieved May 23, 2016, from <https://www.linkedin.com/pulse/20140625224424-19395817-employee-engagement-vs-satisfaction-vs-motivation>
- Wagner, I., & Minge, M. (2015). *The Gods play dice together: The influence of social elements of gamification on seniors' user experience*. Berlin. Retrieved from [https://www.researchgate.net/profile/Michael\\_Minge/publication/280562426\\_The\\_Gods\\_play\\_dice\\_together\\_The\\_influence\\_of\\_social\\_elements\\_of\\_gamification\\_on\\_seniors'\\_user\\_experience/links/55b9d14d08ae9289a09014fe.pdf](https://www.researchgate.net/profile/Michael_Minge/publication/280562426_The_Gods_play_dice_together_The_influence_of_social_elements_of_gamification_on_seniors'_user_experience/links/55b9d14d08ae9289a09014fe.pdf)
- Weber, R. (2014). On Reflections: First interview with the Ubisoft studio's new MD. Retrieved May 23, 2016, from <http://www.gamesindustry.biz/articles/2014-02-26-on-reflections-first-interview-with-the-ubisoft-studios-new-md>

## XVII.FIGURE BIBLIOGRAPHY

Figure 1: Showing the difference between game graphics

<http://gameranx.com/features/id/14125/article/video-games-then-and-now/6/>

Figure 4: Various iterations of a 3D game model

<https://blog.sketchfab.com/wp-content/uploads/2015/03/ff5de26c3690a3e8.jpg>

Figure 5: An example of Game development roles

<http://www.gameschoolprep.com/wp-content/uploads/2014/05/year6.png>

Figure 6: The Agile and Waterfall model

<http://www.blueprintsys.com/wp-content/uploads/agile-waterfall.png>

Figure 7: Work engagement

[https://askusfirst.files.wordpress.com/2010/08/istock\\_000010903737small.jpg](https://askusfirst.files.wordpress.com/2010/08/istock_000010903737small.jpg)

Figure 8: An example of basic Gamification

<http://elearningindustry.com/wp-content/uploads/2016/01/low-cost-gamification-for-adult-learners-001-1024x843.jpg>

Figure 9: The essence of Game Design

[http://www.gamasutra.com/db\\_area/images/feature/3738/cycle.jpg](http://www.gamasutra.com/db_area/images/feature/3738/cycle.jpg)

Figure 10: A screenshot of ChoreWars

<http://nerdsworthacademy.com/wp-content/uploads/2011/05/party-adventures.png>

Figure 11: A screenshot from RedCritter Tracker

<https://gigaom.com/wp-content/uploads/sites/1/2011/07/redcritterteam.jpg>

Figure 12: The use of avatars in ChoreWars

[http://suefrantz.com/wp-content/uploads/2011/01/010611\\_1537\\_ChoreWarsTu1.png](http://suefrantz.com/wp-content/uploads/2011/01/010611_1537_ChoreWarsTu1.png)

Figure 13: Arranged from top to bottom is the Valence, Arousal and Dominance scale

<http://www.theoriesofmind.com/wp-content/uploads/2015/03/Screen-Shot-2015-03-25-at-10.14.39-AM.png>