



ASSIGNMENT OF BACHELOR'S THESIS

Title:	Practical Implementation of Gamification for Scrum
Student:	Martin Koliba
Supervisor:	Ing. Petra Pavlíková, Ph.D.
Study Programme:	Informatics
Study Branch:	Information Systems and Management
Department:	Department of Software Engineering
Validity:	Until the end of summer semester 2017/18

Instructions

Design a gamified system for the Scrum environment to increase work efficiency and performance.

1. Map and analyse existing solutions and possibilities for gamification of SCRUM / agile environments.
2. In cooperation with a company (YSoft) identify and analyse processes and behaviours of development teams in the Scrum environment that increase work effectiveness.
3. Design game elements, game mechanics, forms of measurement, types of rewards, and a gamified system for the Scrum environment.
4. Based on company feedback suggest the procedure of implementation process and time and resources needed for implementation.

References

Will be provided by the supervisor.

Ing. Michal Valenta, Ph.D.
Head of Department

prof. Ing. Pavel Tvrdík, CSc.
Dean

Prague February 7, 2017

CZECH TECHNICAL UNIVERSITY IN PRAGUE
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF SOFTWARE ENGINEERING



Bachelor's thesis

Practical Implementation of Gamification for Scrum

Martin Koliba

Supervisor: Ing. Petra Pavlíčková, Ph.D.

16th May 2017

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended. In accordance with Article 46(6) of the Act, I hereby grant a nonexclusive authorization (license) to utilize this thesis, including any and all computer programs incorporated therein or attached thereto and all corresponding documentation (hereinafter collectively referred to as the “Work”), to any and all persons that wish to utilize the Work. Such persons are entitled to use the Work for non-profit purposes only, in any way that does not detract from its value. This authorization is not limited in terms of time, location and quantity.

In Prague on 16th May 2017

.....

Czech Technical University in Prague

Faculty of Information Technology

© 2017 Martin Koliba. All rights reserved.

This thesis is school work as defined by Copyright Act of the Czech Republic. It has been submitted at Czech Technical University in Prague, Faculty of Information Technology. The thesis is protected by the Copyright Act and its usage without author's permission is prohibited (with exceptions defined by the Copyright Act).

Citation of this thesis

Koliba, Martin. *Practical Implementation of Gamification for Scrum*. Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2017.

Abstrakt

Záměr této práce je prozkoumat použití gamifikace v rámci frameworku Scrum, shromáždit informace o gamifikaci a zmapovat existující řešení gamifikace pro Scrum. Hlavním cílem práce je vytvoření návrhu gamifikačního systému pro Scrum, navrhnout proces implementace a odhadnout potřebný čas a zdroje pro provedení. Výsledky práce jsou následující: seznamy obchodních cílů a cílového chování, které lze řešit gamifikací, návrh gamifikačního systému, návrh postupu implementace a odhad zdrojů a času potřebných pro realizaci projektu.

Klíčová slova Scrum, Gamifikace, efektivita práce, Agile, zábava v práci

Abstract

The aim of this thesis is to investigate the usage of gamification in the Scrum framework, collect information about gamification and map existing solutions of gamification for Scrum. The main goal of the thesis is to design a gamification system for Scrum, suggest a procedure of implementation of the system and estimate time and resources needed for doing so. The results of the thesis are following: wide lists of business objectives and target behaviours which can be addressed with gamification, description of the design of the gamified system, the procedure of its implementation and the estimation of resources and time needed for realisation of the project.

Keywords Scrum, Gamification, effectivity of work, Agile, fun in work

Contents

Introduction	1
1 Theory	3
1.1 Waterfall	3
1.2 Agile	4
1.3 Scrum	5
1.4 Gamification	9
2 State-of-the-art	21
2.1 Educational use of Gamification in Agile	21
2.2 Gamification for Motivation and Efficiency	22
3 Analysis and design	25
3.1 Business Objectives	25
3.2 Target Behavior	26
3.3 Users	29
3.4 Activity Loops	29
3.5 Is it fun?	31
3.6 Appropriate tools	32
4 Plan of the Realisation	37
4.1 Means of Integration into Jira	37
4.2 Technology	38
4.3 Development Team	38
4.4 Implementation Process	39
4.5 Phase 8 Badges	41
4.6 Phase 9 - Creative Game minimum	41
4.7 Estimates of time and resources needed for implementation	42
Conclusion	45

Bibliography	47
A Acronyms	51

List of Figures

1.1	Waterfall: implementation steps to develop a large computer programs	3
1.2	Gamification User Types Hexad, Supporting Gamification User Types [1]	17
4.1	Gantt diagram of implementation phases	42

List of Tables

3.1	Individual User Type Score	30
3.2	Actions in the system	34

Introduction

Modern software development has become profoundly complicated. The knowledge of various programming languages and understanding of computer science is essential, but for creation of a successful software application often not sufficient. Especially in big teams, it is necessary to know the paradigms of software development, internal team processes and best practices and follow them strictly.

The Scrum with its design which aims on customer satisfaction at the delivery time, not project initialisation, ability to flexibly react to changes, cooperation, retrospection and self-regulation became the popular framework for software development teams.

Even though Scrum possesses plenty of positive properties, employees working in big organisations can lose sense of purpose and fail to keep up with their motivation. Additionally, Scrum combined with company process can make an impression of bureaucratic overhead. The gamification is one of the effective ways of how to enhance employees' motivation. Can gamification be effectively used in Scrum to make everyday work more entertaining and increase motivation and efficiency of work?

The objective of this thesis is to map current gamification solutions and usage of gamification for Scrum framework, identify processes and behavior in Scrum environment. The main goal of the thesis is to design a gamification system for Scrum which will motivate members of development teams to follow best-practices and will increase motivation and effectivity of work. Next objective is to suggest a procedure of implementation of the system and estimate time and resources needed for doing so.

In the 1st chapter I present theoretical information I collected about Scrum and gamification. In 2nd chapter I introduce existing gamification solutions for Scrum and describe their properties. In 3rd chapter business objectives are identified and behavior which can be addressable by gamification. The

INTRODUCTION

description of the gamified system is also provided. Chapter 4 describes how such system could be developed and provides estimates of resources and time needed for implementation.

Theory

1.1 Waterfall

The term ‘Waterfall’ is used for sequential development approaches. Development phases are executed in sequential order with some overlaps [2]. Requirements should be clear before the development starts. Returning to previous phases is usually expensive. It can take long time between the specification of requirements and the product delivery to a customer. Waterfall was created for the development of large computer programs [3] when other programming methodologies had not been described yet.

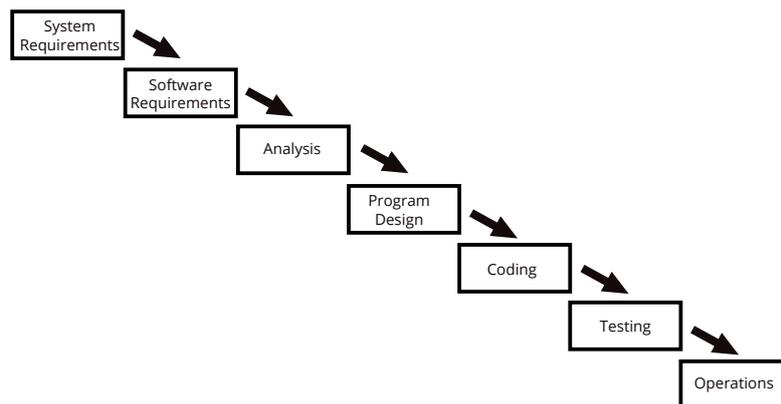


Figure 1.1: Waterfall: implementation steps to develop a large computer programs

1.2 Agile

Major changes often occur concerning many projects in the requirements, scope, and technology that are outside the development team's control.

Conforming to plan is no longer the primary goal; instead, satisfying customers—at the time of delivery, not at the project initiation—took precedence.[...] Because we cannot fully eliminate these changes, driving down the cost of responding to them is the only viable strategy. [4]

The Agile is an alternative to traditional inflexible plan-driven methodologies

In 2001, seventeen software developers representing alternative software development methods or sympathetic to the need of them, formulate and signet what is today known as Agile 'Software Development' Manifesto. [5]

They wrote [6]:

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

The Manifesto is based on twelve principles.

1.2.1 Principles of Agile Manifesto

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly. [7]

1.3 Scrum

Scrum is an iterative and incremental process framework. Its values are compatible with Agile.

‘A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.’ [8]

‘The Scrum framework consists of Scrum Teams and their associated roles, events, artefacts, and rules.’ [8]

The three pillars of the Scrum are following: transparency, inspection, and adaptation.

1.3.1 Team roles in Scrum

1.3.1.1 Product Owner

The Product Owner represents interests of the customer, stakeholders and is responsible for the vision of the project, maximising the value of the product and is the sole person responsible for managing the Product Backlog (the prioritise list of tasks), including its content, availability, and ordering. [8]

The Product Owner serves as an interface between stakeholders and the Development Team. ‘No one is allowed to tell the Development Team to work from a different set of requirements, and the Development Team isn’t allowed to act on what anyone else says.’ [8]

Product backlog management includes:

- Clearly expressing Product Backlog items;
- Ordering the items in the Product Backlog to best achieve goals and missions;
- Optimising the value of the work the Development Team performs;
- Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next; and,
- Ensuring the Development Team understands items in the Product Backlog to the level needed. [8]

1.3.1.2 The Development Team

The Development Team is self-organizing, multifunction group of professionals with all necessary skills needed to create and deliver product. ‘No one tells the Development Team how to turn Product Backlog into Increments of potentially releasable functionality.’ Their aim is to create a potentially releasable Increment product at the end of each Sprint. [8]

1.3.1.3 The Scrum Master

‘The Scrum Master is responsible for ensuring Scrum is understood and enacted.[...] The Scrum Master helps those outside the Scrum Team understand which of their interactions with the Scrum Team are helpful and which aren’t.’ The Scrum Master does not participate in decision making concerning the product implementation. The Scrum Master is leading and coaching in Scrum adoption. [8]

1.3.2 Events in Scrum

‘Prescribed events are used in Scrum to create regularity and to minimize the need for meetings not defined in Scrum.’ [8]

1.3.2.1 Sprint

Sprint is an one month (or less) limited time period in which working and potentially releasable product Increment is developed. During the ‘Sprint no changes are made that would endanger the Sprint Goal, quality goals do not decrease and, scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.’ [8] Sprints consist of the Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.

1.3.2.2 Sprint Planning

Sprint Planning is a time-boxed event in the beginning of the Sprint on which the work to be done in the Sprint is planned. This plan is created by the entire Scrum Team. It is up to the Sprint Planning to decide what can be done in Sprint, how it will get done; task time estimated as well as the goal of the Sprint. [8]

1.3.2.3 Daily Scrum

The Daily Scrum is an everyday 15-minute meeting of the Development Team to synchronize activities, create a plan for the next day and inspect progress toward the Sprint Goal. Team inspect the work since the last Daily Scrum and forecast the work that can be done before next one. [8]

During the meeting, the Development Team members answer following questions:

- What did I do yesterday that helped the Development Team meet the Sprint Goal?
- What will I do today to help the Development Team meet the Sprint Goal?
- Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal? [8]

1.3.2.4 Sprint Review

A Sprint Review is held at the end of the Sprint. The Scrum Team and stakeholders inspect the Increment, discuss achievements and problems, evaluate what could be done to optimize value and adapt the Product Backlog if needed. [8]

1.3.2.5 Sprint Retrospective

The Sprint Retrospective allows the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint. The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning. [8]

1.3.3 Artifacts

‘Scrum’s artifacts represent work or value to provide transparency and opportunities for inspection and adaptation. Artifacts defined by Scrum are

specifically designed to maximize transparency of the key information so that everybody has the same understanding of the artifact.’ [8]

1.3.3.1 Product Backlog

Product Backlog is an ordered list of everything that might be needed in the product and is the single source of requirements for any changes to be made to the product.[...] The Product Backlog lists all features, functions, requirements, enhancements, and fixes that constitute the changes to be made to the product in future releases. [8]

Product Backlog items have their own description, order, estimate and value. It is never complete and evolves as the product and the environment in which it will be used change. The earliest version of it may contain only the initially known and best-understood requirements. Changes in business requirements, market conditions, or technology, cause changes in the Product Backlog, making it a live artifact. [8]

1.3.3.2 Sprint Backlog

The Sprint Backlog is the set of Product Backlog items selected for the Sprint plus a plan for delivering the product Increment and realizing the Sprint Goal. The Sprint Backlog is a forecast made by the Development Team about what functionality will be in the next Increment and the work needed to deliver that functionality into a “Done” Increment.

The Development Team modifies the content of Sprint Backlog during the Sprint as they work through the plan and learn more about the work needed to be done. ‘As new work is required, the Development Team adds it to the Sprint Backlog. As work is performed or completed, the estimated remaining work is updated. When elements of the plan are deemed unnecessary, they are removed.’ [8]

1.3.3.3 Increment

‘The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints.[...] It must be in useable condition regardless of whether the Product Owner decides to actually release it.’ [8]

1.3.3.4 Burn-Down Chart

Burn-Down Charts are used for sprint tracking. ‘Burn-Down Chart shows the remaining effort in day-wise number of hours. Burn-down charts aid the Scrum team to keep track of their progress and what needs to be done to meet the sprint goal.’ [9] ‘The burndown is a chart that shows how quickly you and your team are burning through your customer’s user stories. It shows the total effort against the amount of work we deliver each iteration.’ [10]

1.3.4 Course of the Scrum

1. A product owner creates a prioritized wish list called a product backlog.
2. During sprint planning, the team pulls a small chunk from the top of the wish list, a sprint backlog, and decides how to implement the part of the wish list into the product.
3. The team has a certain amount of time — a sprint (usually two to four weeks) — to complete its work, but the members meet on everyday basis to assess their progress (daily Scrum).
4. Along the way, the ScrumMaster keeps the team focused on its goal.
5. At the end of the sprint, the work should be potentially shippable: ready to hand to a customer, to be put on a store shelf, to be shown to a stakeholder.
6. The sprint ends with the sprint review and retrospective.
7. As the next sprint begins, the team chooses another chunk of the product backlog and begins working again. [11]

1.4 Gamification

The gamification phenomenon has become a common element in many software applications and business cases used as an effective tool to increase user engagement, motivation and drive user experience.

1.4.1 Definition

Kevin Werbach and Dan Hunter defined gamification in their book [12] as:

‘The use of game elements and game-design techniques in non-game contexts’

The dictionary definition is [13]:

noun

1. THEORY

1. the process of turning an activity or task into a game or something resembling a game

Gamification is sometimes confused with serious games or mathematical game theory. However they are completely different area.

Now when we defined lets move on game elements commonly used in gamification.

1.4.2 Motivation

By the dictionary [14] motivation is noun with meaning:

1. the act or an instance of motivating, or providing with a reason to act in a certain way
2. the state or condition of being motivated or having a strong reason to act or accomplish something

Gamification can be use to motivate people:

- to do what they originally wanted to do, but lack motivation
- to make actions
- to develop habit
- to behave certain way

‘Psychologists have been studying how to get people to do things for quite some time. In the second half of the twentieth century, the dominant theory was known as behaviorism.’ [12]

1.4.2.1 Behaviorism

Behaviorism is the systematic approach to understanding human and animal behaviour based on external responses to stimuli. The famous study in this field was made by Ivan Pavlov on his slavering dogs. This and similar studies focus on the effect of rewards and punishment to reinforce a behavior. In nutshell carrot or stick method. [12]

Behaviorist thinking suggested that extrinsic motivation was the way to encourage people to do things. A reward or punishment, systematically applied, would condition and reinforce responses in anticipation of further rewards or punishments. [12]

1.4.2.2 Self Determination Theory

Self-determination theory was introduced by Edward Deci, Richard Ryan [15] and goes against behaviourist approach. ‘Human beings can be proactive and engaged or, alternatively, passive and alienated, largely as a function of the social conditions in which they develop and function.’ [15] Their research focused on conditions that support natural growth and well-being. The three innate psychological needs were identified: competence, relatedness, and autonomy.

1.4.2.2.1 ‘Competence or means being effective in dealing with the external environment’ [12]

1.4.2.2.2 ‘Relatedness involves social connection and the universal desire to interact with and be involved with family, friends, and others. It can also manifest itself as a desire for higher purpose.’ [12]

1.4.2.2.3 ‘Autonomy is the innate need to feel in command of one’s life and to be doing that which is meaningful and in harmony with one’s values’ [12]

1.4.2.3 Intrinsic Motivation

Intrinsically motivated people are doing the activity for its own sake. Doing activity itself is rewarding. [16]

1.4.2.4 Extrinsic Motivation

Extrinsic motivation is about reward not enjoyment of action itself. [16]

1.4.2.5 Reward Demotivation

In some cases, rewards can actually demotivate in certain cases. If people were intrinsically motivated in an activity and we start to give them reward for doing it, they are less likely to do it after we remove the reward. Extrinsic motivation reward crowd out intrinsic motivation. [16] In the study, [17] researchers showed on groups of children doing a creative activity, drawing, that afterwards, they introduce reward for the activity children show less subsequent intrinsic interest in the activity. It is called Overjustification effect.

1.4.3 Rewards

The dictionary definition of reward is [18]:

‘something given or received in return or recompense for service, merit, hardship, etc.’

Another definition in context of gamification is [19]:

‘A reward is a benefit given to the player for some action or achievement.’

1.4.3.1 SAPS

SAPS is motivation framework. It divides rewards into four categories. It means status, access, power and stuff. Its order is important because it is ordered from the most desired one to least desired one.

1.4.3.1.1 ‘Status is the relative position of an individual in relation to others, especially in a social group.’ [20]

1.4.3.1.2 Access is about unique experiences, to unlock something or getting somewhere what is special. Examples of access can be new content, availability to enter into VIP space like airport lounge, meet somebody important, get priority before others.

1.4.3.1.3 Power ‘Prizing power to your players offers a modicum of control over other players in the game. A good player might be asked to serve as a moderator on a forum, for example.’ [20]

1.4.3.1.4 Stuff can be any physical or virtual thing or item. It is considered as least desirable reward. The problem of stuff is that people start to calculate its value and profitability.

‘The primary goal of SAPS is to highlight that cash (or the stuff) is not the ultimate reward for your customer’ [21]

1.4.4 Reward Structure

In the course of gamification on Coursera in chapter Behaviorism Werbach mentioned typology of rewards based on Cognitive Evaluation Theory [16]:

- **Tangible/Intangible:** Tangible rewards are physical items e.g. money, medal. Intangible rewards are things which are not real in any tangible sense. We cannot take them into a hand. Intangible are e.g. virtual badge, levelling up, status, access power, recognition.
- **Expected/Unexpected:** When a reward is expected we know it is coming. Unexpected people does not expect to get or does not know it exists. People love surprises and unexpected rewards are more powerful than expected rewards.
- **Contingency:**

- non-contingent: User gets reward automatically.
- Engagement-contingent: User get reward for starting a task.
- Completion-contingent: User has to complete a task to get a reward. It does not matter on a result.
- Performance-contingent: User has to finish a task with certain level of quality or reach certain level of performance to get task

1.4.4.1 Reward Schedule

‘Reward Schedules refers to when the reward is offered, as opposed to what it is, or what it’s based on.’ [16]

- Continuous: User get reward for each incidence of actions. Least interesting schedule for users
- Fixed Ratio: Rewards is given in fix ratio of a actions occurrence. Every
- Fixed Interval: User is rewarded in fixed intervals of time.
- Variable rewards: Reward is on no fixed schedule. Most interesting reward schedule, brings surprise.

1.4.5 Game Elements

Most common elements used in gamification are points badges a leaderboards. However they are not only elements used in gamification.

In the book For the Win [12] are elements divided into three categories:

- Dynamics
- Mechanics
- Components

1.4.5.1 Dynamics

‘Dynamics are the big picture aspects of the gamified systems system that you have to consider and manage but which can never directly enter into the game.’ [19]

The Dynamics are [19]:

1. Constraints

1. THEORY

2. Emotions
3. Narrative
4. Progression
5. Relationships

1.4.5.2 Mechanics

‘Mechanics are the basic processes that drive the actions and generate player engagement’ [19]

There are ten types of mechanics [19]:

1. Challenges
2. Chance
3. Competition
4. Cooperation
5. Feedback
6. Resource Acquisition
7. Rewards
8. Transactions
9. Turns
10. Win States

1.4.5.3 Components

‘Components are the specific instantiations of mechanics and dynamics.’ [19]

There are 15 common components in games [19]:

1. Achievements
2. Avatars
3. Badges
4. Boss Fights
5. Collections

6. Combat
7. Content Unlocking
8. Gifting
9. Leaderboards
10. Levels
11. Points
12. Quests
13. Social Graph
14. Teams
15. Virtual Goods

1.4.6 Players-Users

Understanding to players, their behaviour and motivation are crucial is for successful designing not only in gamification and game design.

Multiple player types models were developed to help the designer to describe and design successful applications.

In this thesis, I use term user and player interchangeably. Both mean same person who uses the system, application, etc.

1.4.6.1 Bartels Playertype

Probably most spread player type framework was made by Richard Bartle [22]. It was made to categories player of massive multiplayer online games. He divided players into four categories:

1.4.6.1.1 Killer ‘Killers like to provoke and cause drama and/or impose them over other players in the scope provided by the virtual world. Trolls, hackers, cheaters, and attention farmers belong in this category, along with the most ferocious and skillful PvP (player versus player) opponents.’ [23] They act upon players.

1.4.6.1.2 Achiever ‘Achievers are competitive and enjoy beating difficult challenges whether they are set by the game or by themselves. The more challenging the goal, the most rewarded they tend to feel.’ [23] They act within the word

1.4.6.1.3 Socialiser ‘Socialisers like to explore the world - not just its geography but also the finer details of the game mechanics. These players may end up knowing how the game works and behave better than the game creators themselves. They know all the mechanics, short-cuts, tricks, and glitches that there are to know in the game and thrive on discovering more.’ [23] They interact upon players.

1.4.6.1.4 Explorer ‘Explorers are often more interested in having relations with the other players than playing the game itself. They help to spread knowledge and a human feel, and are often involved in the community aspect of the game.’ [23] They interact within the word.

1.4.6.2 Marczewski’s Gamification User Types Hexad

Marczewski in his article [24] point out that gamification differs from MMOG and Bartle’s player types are flawed metaphor for gamification. Marczewski redefined and extend user types to take into account specifics of the gamification and players who are willing to play and who are not. Because not all users can be engaged with the extrinsic things like badges and trophies.

Marczewski’s users types are defined as:

1.4.6.2.1 Socialiser is same as in the original Bartle’s Player Type.

1.4.6.2.2 Free Spirit

Free Spirits like to have agency. They have two basic subtypes, Creators and Explorers. Explorers don’t want to be restricted in how they go through their personal journey, to explore the system. They are also likely to find the most holes in a system. Creators want to build new things. They will have the fanciest avatars and create the most personal content. They seek self-expression and autonomy. [25]

1.4.6.2.3 Achiever

Achievers are the ones who want to be the best at things or, at least, be achieving things within the system. They want to get 100% on the internal learning system. They do this for themselves and are probably not that bothered with then showing off to others about it. (This differs from the original definition, but I could not think of a better word!!). Will compete with others, but as a way to become better than others. The system provides the platform, other “players” are just things to be overcome and mastered. May

also be motivated by status as a representation of their personal achievement. They need a system that will enrich them and lead them towards mastery. [25]

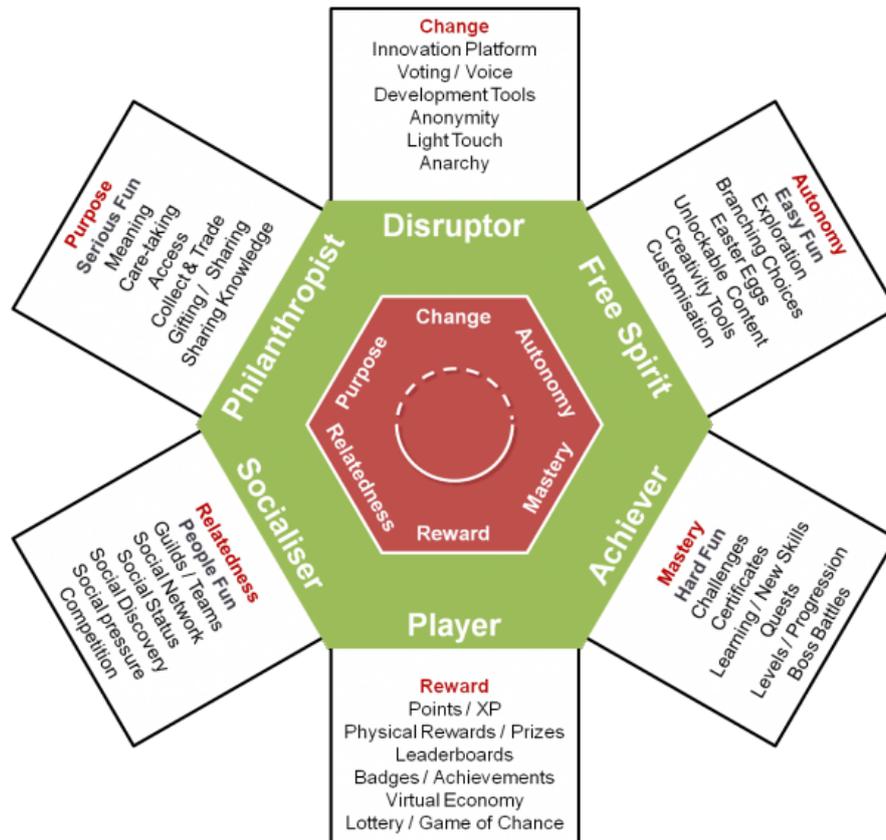


Figure 1.2: Gamification User Types Hexad, Supporting Gamification User Types [1]

1.4.6.2.4 Philanthropist

Philanthropists want to feel that they are part of something bigger. They want to give to others but expect nothing in return. These are the ones who will answer endless questions on forums, just because they like to feel they are helping. They want a system that allows them to enrich others and feel a sense of purpose. [25]

1.4.6.2.5 Player ‘Players are motivated by Rewards. They will do what is needed of them to collect rewards from a system. They are in it for themselves.’ [25]

1.4.6.2.6 Disruptor ‘Disruptors are motivated by Change. In general, they want to disrupt your system, either directly or through other users to force positive or negative change.’ [25]

First four in the list are intrinsic types; Achiever, Socialiser, Philanthropist and Free Spirit. They are motivated by Relatedness, Autonomy, Mastery and Purpose. [1], [15]

1.4.7 Werbach’s 6D Framework

Kevin Werbach [12] created framework for gamification design which helps and lead through design process of gamification system.

‘Gamification is best implemented in six steps, each of which starts—like the word “design”—with the letter D:’ [12]

1. Define business objectives
2. Delineate target behaviors
3. Describe your players
4. Devise activity cycles
5. Don’t forget the fun!
6. Deploy the appropriate tools [12]

1.4.7.1 Define business objectives

‘For effective gamification, it’s critical to have a well-developed understanding of your goals.’ [12] Before we start to design a gamification system we should answer following questions. ‘Why are you gamifying? How do you hope to benefit your business, or achieve some other goal such as motivating people to change their behavior?’ [16] If we do not answer these question the gamification project will probably fail. Our task is to create list of objectives and rank it [12].

After creating the list of objectives, go through the list and cross off anything that is a mean to end rather than an end result [12]. In this phase, we want to define important goals, not mechanics.

‘Getting users to accumulate points and badges isn’t a reason to implement a gamified system.’ [12]

In the end is recommended to explain how the organization will benefit from each objective .

Whole process of the Object definition can be summarised into four for tasks:

1. List objectives
2. Rank objectives
3. Delete mechanics
4. Justify Objectives [12]

1.4.7.2 Delineate target behaviors

In this step, we have to a identify what we want our players to do and how we will measure them [12]. ‘Behaviors and metrics are best considered together. Target behaviours should be concrete and specific.’ [12]

‘The behaviors you are looking for should promote the ultimate business objectives you previously defined, though the relationship may be indirect.’ [12] Once we have listed all desired behaviour, we have to develop our metric for success, define ‘win states’, forms of analytics and how we will measure them [12].

1.4.7.3 Describe your players

We are developing system for people. We should know who these people are. What is their relationship with us? We can use demographics to describe them and some player type models or another type of framework. Based on players we will later choose game elements which are likely to be effective on this particular population. Recommended way is to create personas - a virtual representation of user groups with character models. We should consider player’s life cycle too. As players advance through the system their experience is likely to change. [12]

‘What was at first new and challenging is now effortless.’ [12]

1.4.7.4 Devise activity loops

Games are not a simple linear list of steps. ‘[...]they operate through series of loops and branching trees.’ [12]

There are two kinds of cycles to develop: engagement loops and progression stairs. Engagement loops describe, at a micro level, what your players do, why they do it, and what the system does

in response. Progression stairs give a macro perspective on the player's journey. [12]

1.4.7.4.1 Engagement Loops: [12] Motivation leads to player actions which trigger feedback from system or other users and feedbacks motivates the user to take further actions.

1.4.7.4.2 Progression Stairs: [12] Progression of player can be seen as series 'of short-term missions and long-term goals'. At the beginning, a system has to be simple, then 'difficulty ideally should increase at variable rates[...]' Periods of steadily increasing difficulty should be followed by relative ease.

1.4.7.5 Don't forget the fun!

The gamified system should be fun and function even without any extrinsic rewards. 'Identify which aspects of the game could continue to motivate players to participate even without rewards.' [16]

1.4.7.6 Deploy the appropriate tools

In this stage we have should have a good idea how the system should work. Now we have to explain how the system will look like and specify game elements in detail - how it will be deployed and what overall experience of players will be. [16]

State-of-the-art

One of the tasks of these theses is to map existing solutions for gamification of a scrum.

To search these solutions I used multiple search engines such as Google, Google scholar, search on the web portal of the National Library of Technology and studied research literature on the topic.

In 2014 group of researchers [26] made a systematic mapping of studies focused on gamification in software engineering. They found 29 studies published between January 2011 and June 2014. The majority of them were introduced in conferences or as workshops. They found that ‘[...]38% of the studies consider only the simplest gamification element, namely rewarding user’s behaviors with points, which could be called “pointification”, instead of gamification.’ They also stated that the proposals did not properly incorporated and integrated gamification with tools ecosystem of organisations.

From found existing solutions I chose the ones I considered relevant and divided them into two categories.

2.1 Educational use of Gamification in Agile

There are a lot of studies and proposals how to use gamification in education because it is good tool make people interest in or learn during playing a game.

For the Scrum were created plenty of games. There is a big catalogue of games to train and learn Agile practices and software engineering processes available on <http://tastycupcakes.org/>.

Worth to mention are:

- The Daily SCRUM Game [27]

- Scrum Simulation with LEGO Bricks [28]
- Scrum Knowsy [29]
- Scrum Resort Brochure [30]

2.2 Gamification for Motivation and Efficiency

To drive motivation and increase efficiency in software development and Scrum there were investigated and created multiple solutions in academic sphere and business.

Martin Češka investigated the similar topic in his diploma theses Gamification in the SCRUM Software Development Framework [31]. In his study are described properties of Scrum and gamification and proposed and implemented a prototype of a gamified system. However, this prototype is relatively simple and lack integration into tools used by software development teams.

Another academical project is Scrum Hero [32]. It started as university experiment gamifying game development. In fist iteration authors focus on increase precision task's time estimations, through making bets on other team members estimates. Later it was recreated into the mobile app.

In few recent years appeared on the market multiple commercial solutions in forms of plugins to project management tools, integrations.

2.2.1 Jirafee

Jiraffe [33] is plugging to JIRA, flexible issue tracking and project management tool with agile tooling for software teams. Except for classic gamification elements such us points, badges and leaderboards it should allow you to personalise project with themes to create from it for example pirate story.

Pricing per year [34]:

- 10 users: \$10
- 25 users: \$25
- 50 users: \$100
- 100 users: \$250
- 250 users: \$750

On Atlassian market, it has few reviews which are mostly positive. The price is low. It is the interesting, but the relatively simple solution, I am skeptical how long it will keep users engaged.

2.2.2 GetBadges

GetBadges is Polish startup developing an application with the same name. It utilises gamification through integration into more than 20 apps and tools widely used by developers. It allows its players to play and compete against to release stress and bring synergy. GetBadges can be integrated into Issue trackers, Code repositories, Code review apps, Continuous integrations, Customer relationship management apps, Helpdesk, Team communication chats and other. In total it supports 22. [35]

After connecting to applications GetBadges start to collect data and use communications services such as Slack to send notifications. The administrator has the ability to define and set SMART goals individually to each team. The application allows and encourages users to give feedback and thank each other. For completing task users get experience points. After reaching a certain amount of points they level up. Users have in the app their own profile and bar showing them a number of XP needed to level up. It gives users sense of progression. Leaderboards allow to players to compare their performance. All finished tasks and in-game events related to an in-game resource called Adrenaline. It can be used in badge dungeon to fight monsters. [36]

It supports integration into JIRA. It reacts on following Jira events:

- closed a user story (+ 50XP)
- closed a task (+ 50XP)
- reopened a task (+ 50XP)
- added a user story (+ 50XP)
- opened a new task (+ 50XP)
- found a bug (+ 50XP)
- fixed a bug (+ 50XP)
- reopened a bug (+ 50XP)
- deleted a task (+ 25XP)
- deleted a bug (+ 25XP)

Pricing is [37]:

- Free for private game and 1 project
- Small company: \$672/year, up to 10 players

2. STATE-OF-THE-ART

- Company: \$1152/year, up to 20 players
- Enterprise: \$2400/year, up to 50 players

It is interesting and complex solution with reasonable pricing which I consider be worth to try. However I am little skeptical if it will be still engaging for users after they slay few monsters in badge dungeon.

Analysis and design

As a part of my work on the thesis and cooperation with the company, I did an observation of Scrum processes of one development team. I attended Daily Scrum meetings, Sprint Review, Sprint Retrospective and Sprint Planning. I interviewed the developers, testers and team leader about how they are doing processes, why they are doing them and if they have an idea how to improve them. Later I examined the ways the team use the issue tracker and a project management tool named Jira, which can be utilised as a rich source of information about the team performance and data for analytics and metrics in the gamified system.

I decided to design the gamified system integrated with Jira, widespread tool used by development teams which is also used in the company I cooperated with.

To design gamified system I used Werbach's Six Step framework mentioned in the section 1.4.7.

Based on my observation and interviews I identified business objectives and a target behaviour which could be addressed with gamification.

3.1 Business Objectives

Based on the assignment of thesis, literature, existing solutions, and my observations in the company I defined following business objectives which gamification could address.

1. Increasing work Efficiency
2. Increasing accuracy of planning work scope for Sprint
3. Increasing accuracy of a task's time estimates
4. Following the team's established practices
5. Onboarding of new people into company processes

However, onboarding into company is a process with its own specifics and rules which highly differentiate from everyday routine and could stand as a topic of a whole separate work. Because of these reasons, I will continue with first four objectives.

3.2 Target Behavior

As mentioned in 1.4.7.2 target behaviour should directly or indirectly support business objectives of the gamified system.

I grouped identified target behaviours based on business objective they support. If a behaviour supports multiple objectives, I assigned it to an objective I consider it support most. For each target behaviour, I defined a ‘win state’ and form of analytics to measure it. I define win state as a state which, if reached by the user, is considered by the system as a success and the system rewards the user or do not punish him/her.

BO 1: Increase work Efficiency

3.2.0.0.1 Target behavior: Reduce context switching

3.2.0.0.1.1 Win States: User did not work on multiple unfinished tasks that day.

3.2.0.0.1.2 Analytics: Time logged on tasks, status of tasks.

3.2.0.0.2 Target behavior: Try to do a new feature every sprint.

3.2.0.0.2.1 Win States: Finish user story during sprint

3.2.0.0.2.2 Analytics: Number of user stories in Done state during sprint

3.2.0.0.3 Target behavior: Remove tasks from sprint backlog if team discover they cannot finish in current sprint.

3.2.0.0.3.1 Win States: Remove a task from backlog, If there is more than 10% of unfinished estimated work than team’s capacity to finish in average velocity.

3.2.0.0.3.2 Analytics: Total sum of remaining estimates on unfinished tasks, teams capacity of work to finish in current sprint based on average velocity from last 5 sprints

When multiple teams are working on a project they expect the task in sprint backlog will be delivered on end of the sprint. However, if a critical issue appears in the middle of the sprint, the plan for sprint is ruined and the team is unable to finish what has been planned. Under such circumstances it is better to remove tasks, which are not expected to be finished from sprint backlog so other teams are informed about the change.

BO 2: Increasing accuracy of planning work scope for Sprint

3.2.0.0.4 Target behavior: Finish all task in sprint backlog

3.2.0.0.4.1 Win States: There are no unfinished tasks on the end of sprint

3.2.0.0.4.2 Analytics: Number of tasks in 'ToDo' list, Number of tasks assigned to team member in 'In progress' list

3.2.0.0.5 Target behavior: All tasks in sprint backlog get estimate

3.2.0.0.5.1 Win States: All tasks in sprint backlog get a time estimate before the start of the sprint.

3.2.0.0.5.2 Analytics: Time estimate of task on the beginning of the sprint

3.2.0.0.6 Target behavior: All bugs should have estimates.

3.2.0.0.6.1 Win States: User give time estimate to a bug

3.2.0.0.6.2 Analytics: Time estimate on bug in Jira

Even though it is sometimes almost impossible to predict how much time will be needed to fix some bugs, in long term, certain predictability exists of significant amount of bugs by provided by the experience team.

3.2.0.0.7 Target behavior: Put into sprint backlog amount of work which team is able to finish.

3. ANALYSIS AND DESIGN

3.2.0.0.7.1 Win States: Total sum of tasks' estimates in sprint backlog after sprint planning do not exceed average amount of work in five previous sprints more than 10%

3.2.0.0.7.2 Analytics: Sum of estimates for next sprint, the average amount of work done in last five sprints

BO 3: Increase accuracy of a task's time estimates

3.2.0.0.8 Target behavior: Precise prediction of the time required to accomplish task

3.2.0.0.8.1 Win States: Under X% Difference between predicted and spent time

3.2.0.0.8.2 Analytics: Time predicted / time spent

3.2.0.0.9 Target behavior: Break User and technical stories into task which take max 1 day

3.2.0.0.9.1 Win States: Task and issue have time estimate less than a day.

3.2.0.0.9.2 Analytics: Time estimate on tasks

BO 4: Follow the team's established practices

3.2.0.0.10 Target behavior: Report work time into the work log same day as the work was done.

3.2.0.0.10.1 Win States: Date of insertion into the worklog is same with a date work was reported to

3.2.0.0.10.2 Analytics: Date if insertion into worklog,

3.2.0.0.11 Target behavior: Arrive on meeting on time

3.2.0.0.11.1 Win States: Users are on meeting on time

3.2.0.0.11.2 Analytics: Self reports of users and their colleagues

Currently, the team sets practice any latecomer pays 10CZK into team building fund. They reported that it lead to reduction of late arrivals. However, it did not reduce it entirely and still there are many people often coming late for meetings.

I think it would be worth to try to incorporate positive reward in gamification system. Werbach in his course [16] mention field study [38] of a group of daycare centres where parents had to pick their kids up before a certain time. Some parents were late and researchers' aim was to determine whether the introduction of penalty will reduce the number of late-coming parents. The result was the number of late-coming parents increased. Before the introduction of fine, parents were probably forced to pick-up their children on time by social pressure to spend more time with children and not trouble the workers at the daycare to work overtime. The study suggests that after introduction of fine coming late become a commodity with the price tag.

3.3 Users

To understand and better describe users I choose to use Andrzej Marczewski's Gamification User Types [24], [25]. To determine user types of potential users I asked members of the observed team to fill quantitative user type test questionnaire. The user type test is available at Gamified.uk/UserTypeTest. The test is a 24-item survey made by Marczewski to assess an individual's Hexad user type. The test participants choose one from the seven scaled answers for each statement. The possible answers are: Strongly Disagree, Disagree, Somewhat Disagree, Neither, Somewhat Agree, Agree, Strongly Agree. After submitting answers, the graph and table with percentage distribution among the six user types are generated.

Results of my survey show there are three Philanthropists, two Free Spirits, one Achiever/Free Spirit and one Socialiser. Individual user type scores can be seen in table 3.1.

Average values show participants of test score high in categories Achiever, Free Spirit, Philanthropist. On these categories I focused when designing the system and choosing proper gamification elements.

3.4 Activity Loops

3.4.1 Feedback and Activity Cycle

The system offers several kinds of feedback to to users encourage their further actions. Of course, the system will utilise a classic triad: points, badges and leaderboards. The feedback can be divided into two categories. First is in-

Table 3.1: Individual User Type Score

	Achiever	Free Spirit	Disruptor	Philanthropist	Socialiser	Player
User1	20	21	16	13	16	13
User2	19	19	17	17	15	12
User3	15	18	13	20	19	15
User4	18	17	14	21	18	14
User5	21	17	12	22	16	13
User6	16	18	13	19	20	16
User7	20	21	18	15	14	12
Average	18,4	18,7	14,7	18,1	16,9	13,6

formative feedback giving players information about how they are doing, how their colleagues are doing and whats is happening in the system. For these purposes, it will use a dashboard, a news wall, an Experience Point Bar, a Health Bar and leaderboards. The rewards are second feedback category. As rewards, I use points, badges, status, access and abilities.

When users enter the system they will see the dashboard and the news wall. On the dashboard, there will be a list of activities for which users can get Experience Points (XP) or lose Health Points (HP) and list of public badges. This information should direct users' attention towards activities for which they can earn points and badges. Once user finished one of the actions such as 'Finish User or Technical Story', the system will award user with XP points and put the notification into users news wall.

Awarded XP points are automatically counted in Experience Point Bar, indicating how many XP user earned and the amount of XP required for reaching next level, giving user a sense of a progression and motivating some users to reach next level.

The system punishes users for a non-desirable behaviour by losing a small amount of HP from Health Bar which should lead to users will try to avoid an occurrence of punishable actions.

Sometimes users get additional reward in a form of a healing potion as an unexpected bonus. The healing potion gives users ability to heal some HP. When the user loses some HP he or she may be motivated to make rewarded actions to find the potion or to level up to which reset HP in Health bar to maximum. The potion can be used to heal a hurt colleague character in the system too. This altruistic act will make the receiver happy as well as a pleased philanthropist.

When a user makes a time estimate on a task in Jira the system shows other team members an offer to make a bet on the precision of the time estimate. Team members can win XP if they made the precise bet or lose XP. In bets,

users will quickly discover who is good at precise prediction and will learn how precise their prediction is. The positive side effect here is that users tend to be interested in other team members' task and are better informed of what is happening in the team.

Above the gamification system, there will be a Creative Game giving users the ability to transform earned XP into actions in the game. For each XP a user will get the ability to draw one pixel on a virtual canvas which should motivate them to engage into more positive actions. In the Creative Game, Free Spirits can show their creativity, Socialisers make teams and coalitions, Philanthropists help others in achieving a common goal, Achievers try to control as big area of the canvas as possible and Disruptors troll others.

3.4.2 Progression

When the user enters the main page of the system for the first time they go through a quick bubble-like tutorial, a list of activities for which they can get XP and level up is presented to them and a short list of public badges which they can collect. In this phase, the player gets used to basic system functions, game mechanics and rules.

On the second level, the ability to heal starts to randomly drop as the bonus reward.

After reaching the third level, the ability to make bets on time estimates of other team members unlocks the possibility to compete against each other in the precision of time estimates on task.

After reaching the fifth level players get access to the Creative Game where they are allowed to paint one pixel on canvas for each earned XP . When the users fill the whole canvas the picture is saved into an archive and canvas is then cleared allowing users to start a new piece of cooperative art.

For longer user engagement, it could be useful if users could create their own badges and vote about them. Scrum/game masters could create also secret badges as easter eggs.

3.5 Is it fun?

Is it fun? Would the system work without extrinsic rewards?

In the system, there are not any tangible rewards used . The system utilises motivation mostly on intrinsic and non-tangible rewards. The betting on time estimates is the activity which can be a real fun for people who enjoy competition and challenges. The participation in the Creative Game is based purely on individual intrinsic motivation and does not bring any additional

rewards to a user. After the modification certain parts of the system could work even if points and badges would be removed.

So the answer is the system can prove itself entertaining for some people and its parts would work without extrinsic rewards.

3.6 Appropriate tools

In this section, game mechanics, game elements of the gamified system are described in detail. The plan for despoilment is set in next chapter.

3.6.1 Points

As mentioned before the system uses two types of points: Experience Points and Health Points.

3.6.1.0.1 Experience Points XP are given to players as a reward for making positive actions, getting into ‘win states’, and winning bets.

3.6.1.0.2 Health Points For negative actions and states, a user will lose HP. The user can get HP back by leveling up, using health potion or being health by another user.

3.6.2 Point Bars

There are two point bars in the system.

3.6.2.0.1 Experience Point Bar indicates how many XP user earned and the amount of XP required for reaching next level.

3.6.2.0.2 Health Bar shows how much health user has in the system. When a user does something punishable he or she loses some Health Points

3.6.3 Badges

In the system, there is a list of public badges with known criteria to achieve them. The secret badges function as surprises and easter eggs. The game master should have the ability to create and integrate new badges into the system. The system should contain these badges:

- Beginner: the first issue resolved
- First time estimate

- Bug Hunter of the Month: Awarded every month to a worker in company with the highest number of bugs entered into the Jira.
- Task issue of the month: Complete highest number of issues during month

Additionally administrators will be able to add new badges.

3.6.4 Leaderboards

The system uses two types of leaderboards. First showing results within the team. Second showing score within the company.

Within the team, I would use board showing lists of users with a number of XP earned and a number of closed issues, during the current sprint and list of users with a number of XP earned within last five sprints.

Within the company, I suggest making lists of people based on the amount of XP earned within last five sprints, the bug hunter of the month for reporting bugs into the Jira, the bug killer of a month for resolving bugs and the issue slayer list for completing the biggest number of issues and tasks.

3.6.5 Dashboard

Dashboard shows a list of actions for which users can get rewards and punishments, a list of public badges and actions which user can do such as offer to engage in a bet.

3.6.6 News Wall

News Wall shows results of user's actions in a chronological order and system notifications.

3.6.7 Actions

There are multiple actions in the system - their list is presented in the table 3.2 with reward or punishment which a user can get for an action and the behaviour an action support.

3.6.8 States

In the system, users can reach various win states and lose states based on their effort. When this happens they are either rewarded or punished.

1. User worked on 3 unfinished task within the day

3. ANALYSIS AND DESIGN

Table 3.2: Actions in the system

Action	reward	Supported Behaviour
Finish User or Technical story in sprint	+XP	3.2.0.0.2
Remove task from sprint backlog if there is more than 10% estimated work than team capacity in average velocity	+XP	3.2.0.0.3, 3.2.0.0.6
Give time estimate to a issue	+XP	3.2.0.0.5
Make supportive bet on time estimate		3.2.0.0.8
Make counter bet on time estimate and create new estimate		3.2.0.0.8
Create task	+XP	
Create task with time estimate under 1 MD	+XP	3.2.0.0.9
Draw pixel on canvas	-XP	
User report on time arrival on meeting	+XP	3.2.0.0.11
User report late arrival on meetin	-HP	3.2.0.0.11
Team leader report late arrival of a user on meeting	-HP	3.2.0.0.11
Report time spent time on task into worklog in same day as work was done	+XP	3.2.0.0.10
Report time spent time on task into worklog day after work was done.	+XP	3.2.0.0.10
Report time spent time on task into worklog two days or more after work was done.	-HP	3.2.0.0.10

- Punishment: -HP
 - Supported behaviour: 3.2.0.0.1
2. All tasks in sprint backlog are finished and moved Done column on the end of sprint
 - Reward: All team members +XP
 - Supported behaviour: 3.2.0.0.4
 3. All tasks and issues in sprint backlog have time estimate on start of sprint
 - Reward: All team members +XP
 - Supported behaviour: 3.2.0.0.5
 4. A task is finished and all users who had worked on it reported the time spent into the worklog. The system compares original task's time estimate with time spent on the task and if the difference is less than 10% the system rewards user. The system waits two days more once the

task is closed, for work time reports to evaluate user who created the original time estimate and bets on time estimate of the task.

- Reward: The creator of original time estimate +XP
 - Supported behaviour: 3.2.0.0.8
5. A user makes supportive bet on time estimate and it was within 10% deviation from reality.
 - Reward: +XP
 - Supported behaviour: 3.2.0.0.8
 6. A user makes a supportive bet on time estimate and it deviated more than 10% from time spent on the task.
 - Punishment: -XP
 - Supported behaviour: 3.2.0.0.8
 7. A user makes the counter bet on task's time estimate and user's counter estimate was within 10% deviation from reality.
 - Reward: +XP
 - Supported behaviour: 3.2.0.0.8
 8. A user makes the counter bet on task's time estimate and original estimate was precise.
 - Punishment: -XP
 - Supported behaviour: 3.2.0.0.8
 9. Suma of time estimates of tasks in sprint backlog deviates less than 15% from average value of work done in last five sprints
 - Reward: +XP
 - Supported behaviour: 3.2.0.0.7

3.6.9 Creative Game

The Creative Game is the game above gamified system. It is designed as a big canvas where the users can draw one pixel for every XP they earned from the fifth level. Once users draw on all pixels on the canvas the picture will be saved into the archive gallery, the canvas will be cleared and users get the opportunity to start their creative process again. Users are be allowed to draw on pixels where other users has already drawn yet the cost to draw on these pixels is two-time higher.

Proper size of the canvas has to be a subject of testing so the process drawing does not take ages.

Plan of the Realisation

In this chapter, the procedure of implementation will be suggested with an estimate of time and resources.

4.1 Means of Integration into Jira

The Jira allows several ways of integration.

Add-on, also known as a plugin, adds a functionality to Jira. Add-ons allow to add either a single feature or the rich set of features leading to creation a very different product.

There are two types of plugins for Jira. Atlassian Connect add-ons are web applications that operate remotely over HTTP and run only on Jira Cloud. However many companies use self-hosted Jira Server. The second type is Plugins2. 'A Plugins2 plugin is a single JAR containing code, a plugin descriptor (XML) and usually some Velocity template files to render HTML.' [39]

JIRA has Java APIs and REST APIs providing the way to get data, interact, create and modify information in JIRA such as sprints, boards, epics, etc.[39]

The gamified system could be integrated into the JIRA as Plugins2 add-on or as the separate application using REST API.

The system introduced in the previous chapter is quite complex and require separate user accounts. In the case of separate application, there would be necessary to create user accounts and link them with JIRA accounts or integrate JIRA and the system with another type of authentication.

My suggestion is to build the system as JIRA as Plugins2 add-on because it will be possible to use it with JIRA Server and authentication will JIRA solve for us.

4.2 Technology

JIRA require Java platform on a server to run.

I suggest to write back end of the system in Java programming language. What to use on frontend development team can choose from multiple variants such as Velocity templates, soy template, Atlassian User Interface or stand alone solution free of any Javascript depend on JIRA such as Javascript library React.

4.3 Development Team

To successfully implement and deploy gamified system the team need to consists of people with expertise in following areas:

- Understanding business objectives of the system
- Understand target users and basics of psychology
- User Experience design
- Game design
- Analytics
- Frontend and backend development
- Graphical design

I do not expect that system I suggested in the previous chapter is flawless. Gamification is about experience and psychology. There will be necessary to measure and analyse users behaviour in the system. Observe how they use and interact with the system. Watch if users cheat and how they cheat. The tweaking system, its game elements, rules and rewards will be necessary.

Based on my assessment the team should consist at least of following people:

- Product owner with understanding business objectives, companies inner processes and processes inside teams.
- UX and game designer
- Frontend developer
- Backend developer

4.4 Implementation Process

I created the list of steps and system elements and grouped them together based on the how they way function. Then I ordered them in order how I evaluate their for the system, complexity, and dependency on previous functionality and divided them into phases of implementation.

4.4.1 Phase 1

- Research and Analysis
 - User research and analysis on bigger sample of users.
 - Global analysis
- Setting up environment
- Implementing plug-in skeleton
- Implementation of separate user accounts
- Testing of authentication

4.4.2 Phase 2 - Basic Environment and Functions

- Implementation of interfaces
 - Dashboard
 - Action cards in Dashboard
 - XP Bar
- Implementation of basic Actions
 - Finish User or Technical story in sprint
 - Create task
 - Give time estimate to a issue
 - Create task with time estimate under 1 man day
- Implementation of Finished Task Counter

4.4.3 Phase 3 - News Wall

- Interface of the News Wall next to Dashboard
- Implementation of the notifications for previously implemented actions
- Usability testing

4.4.4 Phase 4

- Team leaderboards
- Company leaderboards

4.4.5 Faze 5 - Betting mechanism

- Notification about news time estimate on which can user bet in news wall
- Card in Dashboard shows all open bets and bets waiting on results
- Interterface to make supportive bet
- Interterface to make counter bet and counter time estimate

4.4.6 Faze 6

The health functionality

- Health Bar
- Random dropping of Health Potion with rewards
- Interface and functionality for using potion
- Interface and functionality for healing colleague
- Usability testing

Reporting arrival on meeting

- Implementation card in Dashboard for reporting arrivals
- Implementation of the action: User report on time arrival on meeting
- Implementation of the action: User report late arrival on meeting
- Interface for team leader to report late arrival of the user

4.4.7 Phase 7 - Additional actions and states

- Implementation of action: Remove task from sprint backlog if there is more than 10% estimated work than teamcapacity in average velocity
- Implementation of States:

- User worked on 3 unfinished task within the day
- All tasks in sprint backlog are finished and moved Done column on the end of sprint
- All tasks and issues in sprint backlog have time estimate on start of sprint
- Suma of time estimates of tasks in sprint backlog deviates less than 15% from average amount of work done in last five sprints
- A task is finished and all users who work on it reported spent time into the worklog. The system compares original task's time estimate with time spent on the task and if the difference is less than 10% the system reward user. The system waits two days after closing task for work time reports to evaluate user who created original time estimate and bets on time estimate of the task.
- Reporting time into work log, implementation of actions:
 - Report time spent time on task into worklogin same day as work was done
 - Report time spent time on task into worklogday after work was done.
 - Report time spent time on task into worklogtwo days or more after work was done.
- Usability testing

4.5 Phase 8 Badges

- Cards with publicly know badges in Dashboard
- Interface of user's collection of badges
- Interface and functionality for adding badge into the system
- Usability testing

4.6 Phase 9 - Creative Game minimum

- Interface of canvas view
 - View of whole canvas
 - Zoomed view for drawing
- Colour palette for changing colours

4. PLAN OF THE REALISATION

- Counter of counter of left XP with expiration
- Usability testing

4.6.1 Phase 10

- Counter of unpainted pixels
- View of map of unpainted pixels
- Gallery for finished canvases
- Saving a canvas into gallery
- Reseting canvas

4.6.2 Phase 11

- Interface for creating teams with common goal in Creative Game
- List of team members of the team
- Link on chat room on slack or other chat application
- Usability testing

4.7 Estimates of time and resources needed for implementation

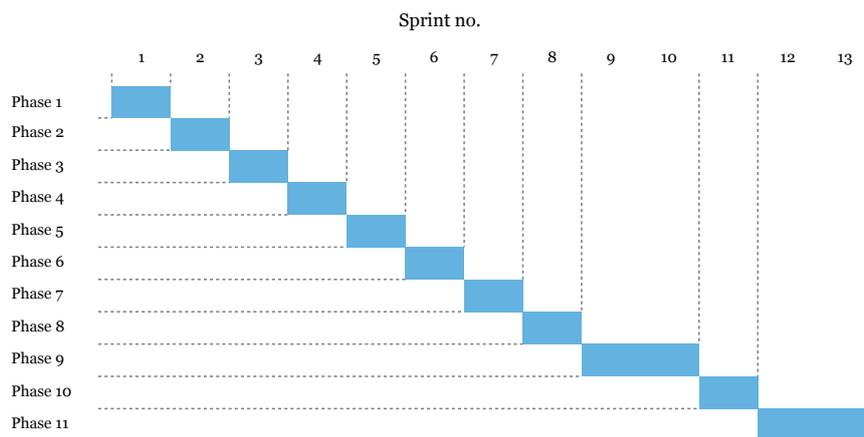


Figure 4.1: Gantt diagram of implementation phases

4.7. Estimates of time and resources needed for implementation

I made time estimate for each phase and enter them into Gantt chart 4.1. I expect the suggested team in section 4.3 consisting from Product Owner, UX/Game designer and two developers will work in two weeks long sprints.

In my estimate, whole implementation procedure will take 13 sprints. That is 6 months or 26 weeks or 130 working days. In the group of three people, it is approximately 390 man days. Plus there is Product Owner. However, I expect Product owner will not work on this project on full-time. I estimate the Product Owner will work on the product on average maximally two days per week. That is 52 man days. In total it is 442 man days which is 3536 hours of work.

If one hour of work would cost company on average 450 CZK per hour, the total cost of the project would be 1 591 200 CZK.

Conclusion

Gamification proved to be effective tool do drive behaviour in various areas if is done properly and existing business examples show gamification can be successfully used in software engineering.

The aim of this thesis is to investigate the usage of gamification in the Scrum framework, collect information about gamification, map existing solutions. The main goal of the thesis is to design a gamification system for Scrum, suggest a procedure of implementation of the system and estimate time and resources needed for doing so.

In the theses are collected information about gamification and Scrum. In the second chapter are listed multiple existing solutions on the topic and their review.

Based on obtained knowledge of gamification, an analyst of Scrum processes and observation of the Scrum team, proposal of the design of the gamification system has been done using the Werbach's gamification framework. I took into consideration important psychological findings and gamification best practices. During designing process were made wide lists of business objectives and target behaviours which can be addressed with gamification, research of user types of the potential users, description of activity loops and detail description of game mechanics and game elements within the system. The last chapter contains proposal of the implementation process, a minimal team needed for implementation and the estimation of time schedule and cost for the realisation of the project.

The proposed gamified system contains gamification elements designed in the way that they should not crowd out the intrinsic motivation of its users. In combination with the proposed Creative Game and expandable system of badges, it should lead into long lasting user engagement.

This thesis makes ground for possible implementation of the system. The

CONCLUSION

validation of the suggested system elements and mechanics will be needed to evaluate if they fulfill expected results. The system also will have to be tested and balanced. The systematical evaluation and comparison of existing solution would be useful to help spread gamification into software engineering.

Bibliography

- [1] A Player Type Framework for Gamification Design. [cit. 2017-5-13]. Available from: <https://www.gamified.uk/user-types/>
- [2] Royce, W. W. Managing the development of large software systems: concepts and techniques. *Proc. IEEE WESTCON, Los Angeles*, August 1970: pp. 1–9, reprinted in Proceedings of the Ninth International Conference on Software Engineering, March 1987, pp. 328–338. Available from: <http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf>
- [3] Benington, H. D. Production of Large Computer Programs. *IEEE Annals of the History of Computing*, volume 5, no. 4, 1983: p. 350–361, doi: 10.1109/mahc.1983.10102.
- [4] Highsmith, J.; Cockburn, A. Agile software development: the business of innovation. *Computer*, volume 34, no. 9, Sep 2001: p. 120–127, doi:10.1109/2.947100. Available from: <http://sunset.usc.edu/events/2002/arr/agile.pdf>
- [5] Highsmith, J. History: The Agile Manifesto. 2001, [cit. 2017-4-3]. Available from: <http://agilemanifesto.org/history.html>
- [6] Manifesto for Agile Software Development. [cit. 2017-4-3]. Available from: <http://agilemanifesto.org/>
- [7] Principles behind the Agile Manifesto. [cit. 2017-4-3]. Available from: <http://agilemanifesto.org/principles.html>
- [8] Schwaber, K.; Sutherland, J. The Scrum Guide™. Jul 2013, [cit. 2017-4-20]. Available from: <http://www.scrumguides.org/scrum-guide.html>

BIBLIOGRAPHY

- [9] tutorialspoint.com. Scrum - Burn-Down Charts. [cit. 2017-5-14]. Available from: https://www.tutorialspoint.com/scrum/scrum_burn_down_charts.htm
- [10] Burndown Charts. [cit. 2017-5-1]. Available from: <http://www.agilenutshell.com/burndown>
- [11] Learn About Scrum. [cit. 2017-4-5]. Available from: <https://www.scrumalliance.org/why-scrum>
- [12] Werbach, K.; Hunter, D. *For the win: how game thinking can revolutionize your business*. Wharton Digital Pr., 2012, ISBN 978-1-61363-022-8.
- [13] Dictionary.com Unabridged. May 2017, [cit. 2017-5-10]. Available from: <http://www.dictionary.com/browse/gamification>
- [14] Dictionary.com Unabridged. May 2017, [cit. 2017-5-13]. Available from: <http://www.dictionary.com/browse/motivation>
- [15] Ryan, R. M.; Deci, E. L. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, volume 55, no. 1, 2000: p. 68–78, doi:10.1037//0003-066x.55.1.68.
- [16] WERBACH, K. Gamification. Available from: <https://www.coursera.org/learn/gamification>
- [17] Lepper, M. R.; Greene, D.; et al. Undermining children’s intrinsic interest with extrinsic reward: A test of the ”overjustification” hypothesis. *Journal of Personality and Social Psychology*, volume 28, no. 1, 1973: p. 129–137, doi:10.1037/h0035519. Available from: http://web.mit.edu/curhan/www/docs/Articles/15341_Readings/Motivation/Lepper_et_al_Undermining_Childrens_Intrinsic_Interest.pdf
- [18] Dictionary.com Unabridged. May 2017, [cit. 2017-5-12]. Available from: <http://www.dictionary.com/browse/reward>
- [19] Werbach, K.; Hunter, D. *The Gamification Toolkit: Dynamics, Mechanics, and Components for the Win*. EBL-Schweitzer, Wharton Digital Press, 2015, ISBN 9781613630686. Available from: <https://books.google.cz/books?id=RDAMCAAQBAJ>
- [20] Zichermann, G.; Cunningham, C. *Gamification by design: implementing game mechanics in web and mobile apps*. O’Reilly, 2011, ISBN 978-1449397678.
- [21] SAPS (Status, Access, Power and Stuff). [cit. 2017-5-10]. Available from: <https://influitive.com/dictionary/saps-status-access-power-stuff/>

-
- [22] Bartl, R. HEARTS, CLUBS, DIAMONDS, SPADES: PLAYERS WHO SUIT MUDES. [cit. 2017-5-13]. Available from: <http://mud.co.uk/richard/hcde.htm>
- [23] Kyatric. Bartle's Taxonomy of Player Types (And Why It Doesn't Apply to Everything). Feb 2013, [cit. 2017-5-13]. Available from: <https://gamedevelopment.tutsplus.com/articles/bartles-taxonomy-of-player-types-and-why-it-doesnt-apply-to-everything--gamedev-4173>
- [24] Marczewski, A. Gamification User Types. Mar 2013, [cit. 2017-5-9]. Available from: http://www.gamasutra.com/blogs/AndrzejMarczewski/20130318/188620/Gamification_User_Types.php
- [25] Marczewski, A. *Even ninja monkeys like to play: gamification, game thinking and motivational design*. Gamified UK, 2015, ISBN 978-1514745663.
- [26] Pedreira, O.; García, F.; et al. Gamification in software engineering – A systematic mapping. *Information and Software Technology*, volume 57, 2015: p. 157–168, doi:10.1016/j.infsof.2014.08.007.
- [27] Troughton, R. Scrumheads: The Daily Scrum Game. [cit. 2017-5-12]. Available from: <http://tastycupcakes.org/2014/07/scrumheads-the-daily-scrum-game/>
- [28] Krivitsky, A. Scrum Simulation with LEGO Bricks. [cit. 2017-5-12]. Available from: <http://tastycupcakes.org/2012/04/scrum-simulation-with-lego-bricks/>
- [29] Scrum Knowsy. [cit. 2017-5-12]. Available from: <http://www.scrumknowsy.com/>
- [30] McGreal, D. Resort Brochure. [cit. 2017-5-12]. Available from: <http://tastycupcakes.org/2009/06/resort-brochure/>
- [31] Češka, M. *Gamification in the SCRUM Software Development Framework*. Dissertation thesis, 2015. Available from: is.muni.cz/th/430640/fi_m/430640_MasterThesis_FINAL.pdf
- [32] Todd Deery - Scrum Hero: The Gamification of Scrum. Nov 2014, in: YouTube [online]. [cit. 2017-5-13]. Chanel of the user Gamification Co. Available from: <https://www.youtube.com/watch?v=4LBbdr6ZwK4>
- [33] bugpotion.com. Bug Potion. [cit. 2017-5-13]. Available from: <http://bugpotion.com/>

BIBLIOGRAPHY

- [34] Jiraffe. [cit. 2017-5-13]. Available from: <https://marketplace.atlassian.com/plugins/com.bugpotion.jiraffe/server/pricing>
- [35] Motivation system for teams. [cit. 2017-5-13]. Available from: <https://getbadges.io/>
- [36] Gietkowska, J. 5 Ways for Employee Motivation Using GetBadges. Jan 2017, [cit. 2017-5-13]. Available from: <https://getbadges.io/blog/5-ways-for-employee-motivation-using-getbadges>
- [37] GetBadges Reviews: Overview, Pricing and Features. [cit. 2017-5-13]. Available from: <https://reviews.financesonline.com/p/getbadges/>
- [38] Gneezy, U.; Rustichini, A. A Fine is a Price. *The Journal of Legal Studies*, volume 29, no. 1, 2000: p. 1–17, doi:10.1086/468061. Available from: <http://rady.ucsd.edu/faculty/directory/gneezy/pub/docs/fine.pdf>
- [39] Atlassian Developers. Feb 2016, [cit. 2017-5-10]. Available from: <https://developer.atlassian.com/jiradev/jira-platform/building-jira-add-ons>

Acronyms

MMOG Massively multilayer online game

HTTP Hypertext Transfer Protocol

API Application Programming Interface

JIRA Issue and project tracking software

UX User Experience

XP Experience Point

HP Health Point

REST Representational state transfer