Using Gamification in Mobile Activation Learning System

V. Emets, J. Rogowski, J. Krasiukianis
Institute of Information Technology
Lodz University of Technology
Wólczańska Str. 215, Lodz, Poland
volodymyr.yemyets@p.lodz.pl, jan.rogowski@p.lodz.pl, jacek.krasiukianis@gmail.com

Використання Гейміфікації в Мобільній Системі Активізації Навчання

В. Емець, Я. Роговський, Я. Красюкянис
Інститут інформаційних технологій
Технологічний університет м. Лодзь
вул. Вульчаньська 215, м. Лодзь, Польща
volodymyr.yemyets@p.lodz.pl, jan.rogowski@p.lodz.pl, jacek.krasiukianis@gmail.com

***Abstract*—This paper describes the application of gamification in the mobile and web applications serving as a gamified learning system for use both at home and in a classroom. The use of the system can lead to the pupils activation during the whole process of learning.**

***Анотація*—У даній статті описується застосування гейміфікації в мобільних і веб-додатках, які служать в якості гейміфікованої системи навчання для використання як вдома, так і в класі. Використання системи може призвести до активізації учнів протягом усього процесу навчання.**

Keywords—gamification; pupils activation

Ключові слова—гейміфікація; активізація учнів

I. INTRODUCTION

The purpose of the paper is to present the idea behind gamification as an innovative method of motivation and to prove the viability of a teaching system that is taking advantage of gamification of learning and persuasive technologies. Studies show that game-based learning not only can improve motivation to learn [1], but also keep students' concentration at a higher level [2].

Gamification describes a broad trend of applying of game design elements: mechanics and experience in a non-game context in order to improve user digital engagement and motivation to achieve one's goals. Key terms of the definition consist of *mechanics* (badges, points, leaderboards and other elements common to many games), *experience* (playing space, a storyline etc.), *digital engagement* (interaction with digital devices as opposed to people), *motivation* (to change behaviors, develop skills, stimulate innovation), g*oals* (those of the player should be aligned with those of the organization).

Gamification attempts to increase engagement by the following means:

Accelerated feedback cycles Real-world feedback loops, such as annual appraisals, are too slow and too far apart in time. Gamification rises the frequency of feedback.

Clear goals and rules of play Real-world goals are often ill-defined and rules are applied selectively. Gamification sets clear goals and enforces clear rules of play.

A compelling narrative Most real-world activities are hardly compelling. Gamification provides a narrative to fill that hole.

Challenging yet achievable tasks Most real-world challenges are simply too complex and long-term. Gamification offers the player more lesser challenges to keep them interested.

Techniques used in gamification include:

Rewards for accomplishing desired tasks and friendly competition were the first techniques used in gamification. Rewards can be points, levels, badges, the filling of progress bars, virtual currency, a place on a leaderboard, unlocking a higher difficulty level. A simple way of encouraging competition is to make each player's progress and results visible to the others, for example by means of a leaderboard [3].

Actual gameplay elements. Another rather direct approach to gamification is to incorporate elements of actual gameplay to make virtually everything seem like a game. Such elements include: meaningful choices, informative tutorials, adaptive increases in challenge, narrative.

Big data can play a big part in a gamified system. Its main purpose is to capture the data on the behavior of users, such as employees and clients. Having this data allows an organization to adapt by offering more engaging experience, which results in better performance (in case of employees) and increased loyalty (in case of customers).

The concept of motivating people through fun and play is not recent by any means. In 1982, Thomas W. Malone published a paper on "Heuristics for designing enjoyable user interfaces" [4]. In 1999, Stepher W. Draper published "Analysing fun as a candidate software requirement" [5]. Further research on fun and play in serious scenarios was a natural follow-up, leading to coining the term *funology*, which is an idea that software interfaces can be actually fun to use, going as far as eliciting emotions and feelings in users. The noughties saw the first attempt to gamify software. The actual term *gamification* was coined in 2002 by Nick Pelling, but attracted little attention. In 2008, blogger Bret Terrill mentioned the word *gameification* used to describe the application of game mechanics to web design [6]. The term gained popularity approximately two years later, not before being simplified by dropping the 'e'. 2010 saw the foundation of Badgeville, a SaaS (Software as a Service) company oriented around gamification, a gamified to-do list application Epic Win, a book on *Games-Based Marketing* by Gabe Zichermann and a TED Talk by Jane McGonigal [5]. Finally, in 2011, the gamification become widely recognized in industry and academia alike, with the establishment of the annual Gamification Summit and the Gamification Research Network. In 2012, gamification began to lose its initial momentum. The idea of merely accumulating badges seemed to gradually get old. Gartner predicted that in two years time, 80% of gamified applications will not meet business goals mainly because of their poor design [7]. However, Barry Kirk of Bunchball defied their prediction claiming that even bad gamification is still viable [8]. The claim was incidentally backed up by Badgeville, which raised $25 million [9].

II. GAMIFICATION IN SCHOOLS

This section will focus on the incorporation of gamification into school curricula, specifically, the examples of successful deployments.

A. Quest to Learn

Quest to Learn is a public school in New York for children between six and twelve years old. A defining feature of the school is its innovative approach to education, developed by teachers cooperating with game designers of the Institude of Play and funded by the MacArthur Foundation [10]. Every trimester, each teacher teams up with a game designer and a curriculum designer. They prepare gamified curricula which are relevant in modern day, establish concrete learning goals and put special emphasis on subjects which students tend to have difficulty with [11]. Not only the students are learning at Quest to Learn. The teachers and game designers gather feedback from the courses and improve them.

Quest to Learn does not distinguish between traditional school subjects, and integrates new technologies into every course. For example, math is taught in two units: The Way Things Work, together with physics, chemistry and biology, and Codeworlds, mostly concerned with the relationship between language and code, intertwined with programming. In every unit, especially Sports for the Mind, students design games and use digital media. Through interdisciplinary courses students become researchers, historians and biologists, by means of podcasts, movies and games.

B. Gaming the Classroom

Gaming the Classroom is a blog created by Lee Sheldon, a teacher at Indiana University, and Jenna Hoffstein, a graduate. The blog describes a game design course taught by Sheldon that has been gamified to imitate an MMORPG (massively multiplayer online role-playing game). The whole course resembles a multiplayer game. Every student needs to choose a handle, an avatar (a portrait of the in-game character) and decide what kind of game they wish to design. Students then join into groups, called guilds, based on that preference and their skill, and sit together in the classroom. Taking tests and exams is referred to as fighting monsters (also known as mobs), preparing reports and analyses is classed as crafting, and research and presentations are the equivalent of completing quests.

C. Programming class at Lincoln Lutheran

Lincoln Lutheran is a middle and high school in Missouri. Lloyd Sommerer is a programming teacher who gamifies his class by introducing game mechanics to it. He first thought of experimenting with gamification in August 2013, in order to provide an external motivation for students who lacked it internally [12]. Leaderboards were the first gamification mechanics incorporated by Sommerer. The leaderboards were arranged in such a way that the home page showed the top 5 students in each category, and individual pages showed top 10 students. Everybody could also view their positions on each leaderboard on their personal page. Students gain levels by reaching certain experience points milestones.

III. SYSTEM PROJECT

Incorporation of gamification into school curricula is already happening. However, what they usually lack is computerized, standard infrastructure which would allow for much wider adoption. A system designed to aid in lecturing and motivating students. The system is supposed to be adopted by schools, and as such must meet several criteria: be simple enough to be easily deployable, provide a clear, user-friendly interface as to require as little additional staff training as possible, support several platforms to reduce deployment costs.

The system can be used in two ways: in a classroom and at home. In a classroom, the system allows to assign the tasks to students and ensure that students actually do them. At home, students use the system to access assigned tasks remotely. The system also saves teachers time by checking the homework automatically.

The system usage scenarios are as follows.

1) Student

The student is supposed to only access the mobile application and can perform the following actions: receive

tasks to solve, solve the tasks in application, submit solutions to the server, receive points for solving the tasks.

The actions are connected and executed in sequence, as presented in Figure 1.

2) Parent

From the perspective of the system, the role of a parent is mainly connected with the points students collect. Although merely accumulating them might feel somewhat rewarding, the main purpose of the points is to be exchanged for rewards. What rewards are available and how many points they cost is intentionally not covered by the system, but instead left at the sole discretion of a parent. Moreover, taking advantage of the automation and integration of the system it is very easy to distribute email notifications to parents about the progress of their children. Parent's interactions with the system are presented in Figure 2.

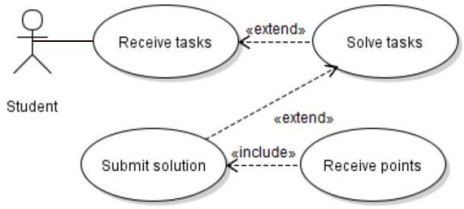

Fig. 1. Use cases of a student

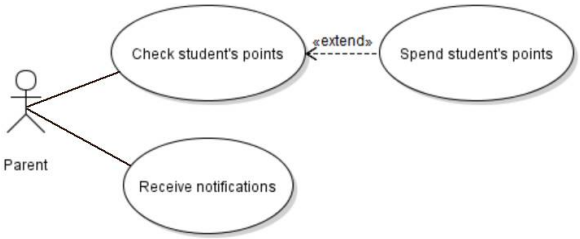

Fig. 2. Use cases of a parent

3) Teacher

Instead of presenting in real-time on a blackboard, the teachers are supposed to prepare video and text content beforehand and distribute it using the system. They also have the access to the leaderboards, which rank students who have successfully completed assigned tasks. Teacher's usage scenarios are presented in Figure 3.

The system is composed of two modules: the mobile application and the web component. The mobile application is intended to be used primarily by students, but also contains features designed for parents. A teacher is supposed to manage the course by using the web application. The applications do not communicate directly, but instead rely on the common server to pass the data between them. This ensures modularity and extensibility.

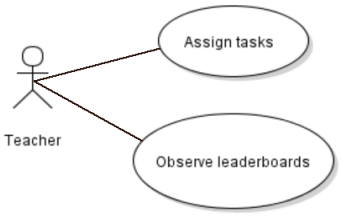

Fig. 3. Use cases of a teacher

The web component is written in Java programming language and runs on the Google App Engine platform. Google App Engine is a cloud computing solution, used for both development and hosting of web applications (platform as a service). It has been selected because of its ease of use, as well as freemium pricing model.

The web component serves two primary purposes: providing teachers with a simplified CMS (content management system), currently implemented using the standard combination of HTML5, CSS3 and JavaScript, and exposing a RESTful API, which allows clients to consume content (in form of learning material and problems to solve) and to submit solutions. REST (representational state transfer) is a set of specifications describing the interactions between web-enabled components. It has become the de facto standard architectural style for web communication, often viewed as a successor of service-oriented architecture. Data is exchanged using JSON (JavaScript Object Notation) format, due to its superiority to XML in terms of simplicity and bandwidth consumption.

Given the actual system is a prototype, no persistent storage mechanism has been implemented, with the server holding all data in memory. A possible solution to this would be to incorporate Objectify, a simple data access API for the Google App Engine, or a separate, fully fledged database (seeing that separation is a desirable trait in system design, whereas vendor lock-in is not).

The other component, the mobile application, is also written in Java and developed for the Android operating system. Android has been chosen as the target platform because of its popularity, accessibility, flexibility and openness - most of its components are free and open source software. The application supports all Android versions from 4.4.2 onwards, thus being available to virtually all Android users.

The primary purpose of the application is to present up-to-date learning material to students. The API exposed by the web component is consumed by Retrofit, a popular open source REST handling library, paired with Gson for mapping JSON submissions and responses.

The storage mechanism of the mobile application is greatly simplified. Because the learning materials are not persisted on the device, only the SharedPreferences storage is used. The Android operating system offers native support for SQLite3 databases, making them an obvious choice for data persistence. Currently, learning material consist of only text and URLs to video content, making binary file storage unnecessary. In a case when video content is to be directly fetched from the server and stored on the device, it should be saved separately to

external storage, as binary files should not be placed directly inside an SQLite database.

Because the mobile application is supposed to be used much more extensively by many more users, significantly more focus has been put on designing it. The source code is organized according to the Clean Architecture. All software dependencies are managed by Dagger 2, a dependency injection framework developed by Google as a successor to Dagger. View layer is bound to the logic layer using ButterKnife, a lightweight boilerplate code generator utilizing custom annotations. Such design offers maximum compliance with SOLID principles and as a result makes the code significantly more reliable, maintainable and extensible.

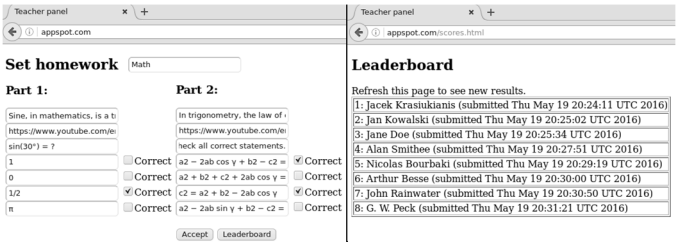

Fig. 4. Task assignment panel and Leaderboard presented in web application

For the system to be of any use to a student, that student must be registered. As such, the students are best registered before the start of a course, their system accounts linked to the school accounts from the very beginning. This also allows the mobile application to skip the registration step, making it simpler and thus easier to use.

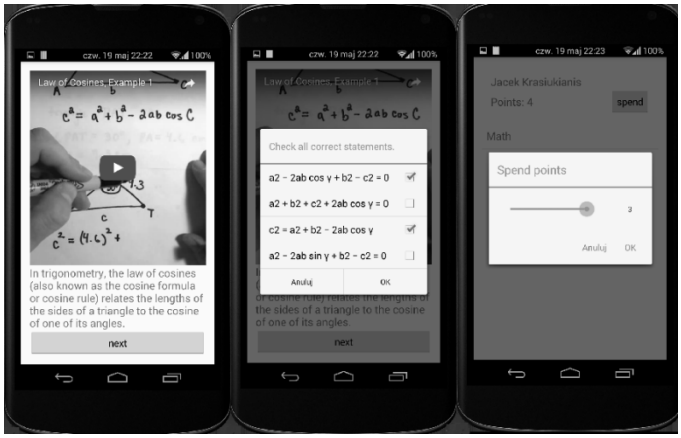

Fig. 5. Screens of the mobile application

In order to assign the tasks for students, a teacher visits a website hosted over the Internet or the school intranet and accesses the panel presented in Figure 4. The tasks are grouped in homeworks. Each task consists of: textual information, URL to video content, multiple-choice type question.

To access the application a student logs in providing a username. After that, main screen of the mobile application is presented. A student provides the name of a homework and initiates the process of solving it. Then, a separate screen is presented for each task the homework is composed of (see Figure 5). After answering the last question, the solutions provided by a student are uploaded to the server. The server verifies the answers and rates the submission. Beside the feedback, a student receives a number of points as a reward. The screen also feature the option to share the achievement on Facebook. If the submission is 100% correct, the server places it on the leaderboard.

IV. Conclusion

The system may offer several advantages. The first, most clear advantage is that a significant part of the menial work a teacher must do (checking homework, managing grades, etc.) is eliminated. The delivery of the content is also automated - a teacher needs to prepare the content once, and then, during classes, merely answer questions (which can serve as a feedback as to which parts of the content need clarification). Naturally, this workflow remains simply a suggestion and a teacher remains in complete control over the course. The system is very modular in nature and highly extensible. It can be fitted with support for any school intranet or content management system already in place, or act as a module for it.

References

[1] T.Y. Liu, Y.L. Chu, "Using ubiquitous games in an English listening and speaking course: Impact on learning outcomes and motivation", Computers & Education, vol. 55(2), pp. 630–643, 2010.

[2] K. Squire, "Video Games in Education", International Journal of Intelligent Games and Simulation, vol. 2(1), pp. 49–62, 2003.

[3] Badgeville wiki, https://badgeville.com/wiki/.

[4] T. W. Malone, "Heuristics for designing enjoyable user interfaces: Lessons from computer games", ACM Digital Library, 1982.

[5] Z. Fitz-Walter, "A brief history of gamification", http://zefcan.com/2013/01/a-brief-history-of-gamification/, 2013.

[6] B. Terrill, "My Coverage of Lobby of the Social Gaming Summit. Bret on Social Games", http://www.bretterrill.com/2008/06/my-coverage-of-lobby-of-social-gaming.html, 2008.

[7] C. Pettey, R. Meulen, "Gartner Says by 2014, 80 Percent of Current Gamified Applications Will Fail to Meet Business Objectives Primarily Due to Poor Design", http://www.gartner.com/newsroom/id/2251015, 2012.

[8] B. Kirk, "Defying the Odds of Gartner's Gamification Prediction", http://www.bunchball.com/blog/post/944/defying-odds-gartners-gamification-prediction, 2012.

[9] R. Herring, "Gamification Platform Badgeville Raises $25M", http://www.redherring.com/internet/gamification-platform-badgeville-raises-25m/, 2012.

[10] About Q2L, http://www.q2l.org/about/.

[11] B. Waniewski, "Meet The Game Designers Who Are On A Quest To Make NYC Public School More Fun", Fast Company, http://www.fastcompany.com/3003920/meet-game-designers-who-are-quest-make-nyc-public-school-more-fun, 2012.

[12] L. Sommerer, "Gamification of our Programming Class. Lincolnlutheran.org", http://www.lincolnlutheran.org/programming/.

[13] V.F. Emets, J. Rogowski, J. Krasiukianis, "Mobile Activation Learning System Using Gamification Approach", in Advances in Intelligent Systems and Computing, vol. 512, N. Shakhovska, Ed. Springer, 2017, pp. 101-114.