GameCourse - Adaptive UI

Mónica Sánchez Lemos Alves
monica.alves@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

November 2023

Abstract

Gamification, defined as ”the use of game elements in non-gaming contexts”, has gained popularity in many fields, especially in education. It has been beneficial to students to improve their user experience, in terms of motivation and performance. However, each user is different so there is a need to escape the ”one-size-fits-all” approach and provide an adapted experience for each one. Moreover, deciding which game elements to use and how to present them is crucial for gamification to be effective. GameCourse is a gamification system used in the context of Multimedia Content Production (MPC) course, taught at Instituto Superior Técnico. The current system counts on adaptive functionalities to improve users' experience depending on each student's profile. The purpose of this thesis is to explore new ways to adapt to different player profiles, as well as to update GameCourse user interface, ensuring support for both current and future functionalities. To achieve this, we created different game element versions and defined how they would be seen differently depending on the type of student profile. Moreover, the creation of a new Notification System was integrated to advance the level of personalized experience, delivering individual messages to students based on their actions in the system. From the results obtained, we concluded that we were successful in most of the new game elements for adaptation, and the rest of UI modifications.
Keywords: Gamification; Adaptation; Personalized Experience; User Interface; User profile.

1. Introduction

Video games first appeared in the late 20th century and gained popularity due to their easy engagement with the audience. They started to attract male customers, and only later female customers and families [23]. However, they were not the only ones interested in this invention. Educational researchers have also shown interest in this type of entertainment. It is because it requires a player to get involved in an active learning process to beat levels or master the game. Students could improve their learning behavior and engagement by combining game mechanics with learning content. That is why researchers have theorized this concept of *gamification* in the educational field.

Gamification can be defined as ”the use of game elements in non-gaming contexts” [6]. Using gamification can have positive impacts whether it is for academic achievement or conceptual and application knowledge. However, sometimes gamification can be too general, and ineffective with the learning outcomes [25]. In an attempt to reduce these negative results, it is essential to consider a user as an individual and to avoid the ”one-size-fits-all” approach [14]. Instead, by considering certain aspects like the user's characteristics (demographics/preferences) or the system's context (geographic location), gamification can be adapted to each own, and consequently, users can have more personal experiences.

For the same purpose, Instituto Superior Técnico has developed its own gamified course, called Multimedia Content Production (MCP), which is taught through a system called *GameCourse*. The platform (developed internally) encouraged students to participate in return for rewards such as badges or experience points. During the Covid-19 pandemic, when traditional classes became an online activity, this platform had a big impact on students. An internal result of the MCP course demonstrated that students perceived classes as more interesting and fun and that using *GameCourse*'s platform provided a healthy environment for students to learn, as well as an improved relationship between students and educational institutions.

1.1. Objectives

The goal of this thesis is to upgrade GameCourse's Adaptive UI for **improving both system interface and adaptation processes to show different pages according to specific student profiles**. Performing these changes will ameliorate user ex-

perience to a more customized one, and ensure a consistent interface throughout the pages, ushering in a new version of growth for GameCourse.

2. Related Work

This section presents research about gamification, and in particular, adaptation approaches to make gamification personalized to its users.

2.1. Gamer Profiling models

Everyone is different, their motivations, traits, and personality characteristics define how to act when facing tasks and obstacles, and how to achieve desires and needs [18]. Therefore, when the goal is to create an adaptive gamified system, considering various profile types is crucial to achieving a customized version for each person [28].

Various models have already been proposed and validated to represent different user profiles, for instance, **BrainHex**'s [17] or **Hexad**'s [16]. **BrainHex** was inspired by prior models, such as **Bartle** [3], and uses an observational approach. It is also based on demographic game design models (DGD1, DGD2) [17, 4] and insights from neurological findings to create the profiling process.

This model identifies 7 profile types, that are categorized into three motivational orientations (Figure 1): ***Seeker***, ***Survivor***, and ***Socialiser*** are aesthetic-oriented (related to social activities). Seeker is motivated by finding new things, Survivor by intense excitement of escaping from a terrifying threat, and Socialiser by talking/helping/being around other people; ***Daredevil***, ***Mastermind*** and ***Conqueror*** are skill-oriented (related to action). Daredevil represents excitement of taking risks and living on the edge, Mastermind represents fun of solving problems that require strategy, and Conqueror represents satisfaction of fighting against the odds for victory; ***Achiever*** is goal-oriented, meaning that enjoys completing collections, solving puzzles, and formulating problem-solving strategies.

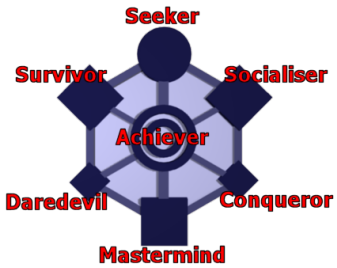

Figure 1: BrainHex model[1]: users' profile illustration.

On the other hand, **Hexad model** focuses on users' motivation, rather than focusing on user experiences [16], like BrainHex. It distinguishes six types of profiles, depending on their motivation origin (Figure 2): intrinsically motivated, that is, motivated by self-realization, such as ***Socialisers*** (driven by *Relatedness*), ***Free Spirits*** (driven by *Autonomy*), ***Achievers*** (driven by *Mastery*), and ***Philanthropists*** (driven by *Purpose and Meaning*); extrinsically motivated, that is, motivated by external rewards, which is the case for ***Players***; motivated by *Change*, disrupting the system either directly or through other users, which is the case for ***Disruptors***.

Figure 2: Hexad model [28]

These two models employ a series of questions to assess how users are represented by gamification user types [27, 7]. BrainHex's are similar to a psychometric survey regarding each player's preferences, while Hexad's applies a set of statements that users ought to answer through a 7-point Likert scale and the highest summed score within each statement indicates the user type.

2.2. Impacts on Motivation

Self-Determination Theory (SDT) proposes that user types differ based on how they can be motivated by either extrinsic (e.g. goals, purposes, rewards) or intrinsic (e.g. enjoyment of the task itself) motivation [13]. A gamification design framework called *Octalysis*, proposed by Yu-kai Chou, states that there are 8 core drives in our brains, and "if there are none of these Core Drives behind a desired action, there is no motivation, and no behavior happens" [24].

The Core Drives are the following: **Epic Meaning and Calling** (when feeling a greater purpose); **Development and Accomplishment** (drive for progress, skill development, mastery, and challenges); **Empowerment of Creativity and Feedback** (when engaging in a creative process); **Ownership and Possession** (when motivated by a sense of ownership/control); **Social Influence and Relatedness** (drive for social factors like mentoring or acceptance); **Scarcity and Impatience** (desire for rarity, exclusivity, or immediate unavailability); **Unpredictability and Curiosity** (when engaging with uncertainty); **Loss and Avoidance** (drive to

[1] https://blog.brainhex.com/what-does-my-brainhex-icon-mean.html

avoid something negative from happening).

These Core Drives aim to achieve both intrinsic and extrinsic motivation as well as engagement in users. So, when designing game-based systems, the first attempt is to captivate users' attention through external rewards and slowly change that strive towards more intrinsic motivation, which will lead to long-term engagement [9]. However, users must not be taken as a whole. Each is unique and has his/her motivations, preferences, and traits that dictate how they act toward the system [18]. Thus, is crucial to consider each one of them as an individual.

2.3. Adaptation Process

Choosing an effective adaptation process can determine the success of a gamified application. However, there is no unique way of doing so, and different approaches can lead to different results [5, 18]. To achieve the best possible outcome and mitigate its negative effects, researchers textitasize the importance of personalizing gamification and providing tailored experiences to users.

Personalization can be defined as a process of collecting user information during interaction with the user, which is then used to adjust appropriate services to the user's needs [8]. Some studies defend the idea of "inside-out personalization" [19], in which the tasks should dictate which game elements to use. The procedure originates from the system itself, by first analyzing the type of activity being executed, and only then providing game elements that suit it. It avoids having different outcomes depending on each user, and when the right elements are picked for each activity, it enhances the potential positive results.

Other studies defend that the adaptation should not be based on the system tasks, but rather on the users themselves ("user-based" perspective) [21, 11]. Centered around the SDT [13], Marczweski [15] proposed the idea of different player types, which offered a new perception of how to adapt a user experience based on their preferences. Hence the importance of analyzing users' characteristics (like demographics) and using that to determine which designs will work best. This information can be gathered by providing questionnaires to users, and then adapting [12, 20], or letting users choose their gamification elements, such as *points*, or *leaderboards*. However, even if the selection is accurate and favorable for most cases, it might not always match the features that motivate them for a given activity [26].

3. GameCourse

GameCourse is an online platform that originated in the context of Multimedia Content Production (MCP), a gamified MSc course that has been taught at Instituto Superior Técnico for over a decade now. In it, teachers can create and manage courses, and students can learn and participate in the activities set by a specific course. This platform includes traditional face-to-face classes with gamification features and successfully encourages students to be more proactive while learning [2, 1].

For this purpose, different game elements have been introduced. GameCourse counts with Experience Points (**XP**), **progress levels**, **leaderboard**, **badges**, **streaks**, **virtual currency**, **wildcards**, and a **skill tree**. Students are awarded once they perform and complete the course's activities, which afterward translates into grade points. It also sends a **weekly report**, so students are aware of their progress and celebrate their latest achievements.

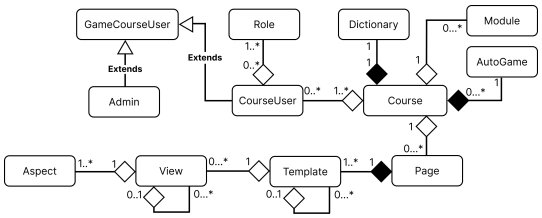

Figure 3: GameCourse simplified class diagram

GameCourse includes three types of users: users of the system (*GameCourseUsers*), users inside a course's context (*CourseUsers*), and users who have full access to the system (*Admin*). All CourseUsers have a specific *Role* (additional roles can be created, but default ones are "Teacher", "Student" and "Watcher"). Moreover, courses have a set of *Modules*, that change their behavior depending on which are enabled. The modules include a set of tools that help refining the functionality of the course, such as the "Profiling" module.

The "Profiling" module is responsible for clustering students into different player profiles - *Achievers*, *Regulars*, *Halfhearted*, and *Underachievers* - depending on their participation with GameCourse. It is usually run weekly, to have students' most accurate player profiles.

Other modules include distinct data sources that are used to import external information and integrate it into the system, such as s Moodle[1], ClassCheck[3], and GoogleSheets[3]. Interactions between the GameCourse system and the data sources modules are assisted by the system's own Expression Language (EL), provided by a *Dictionary*. The EL is structured with multiple libraries, functions, and variables. The functions are always available, some associated with the system's basic concepts and others associated with certain modules. The

[1] https://moodle.org/about
[2] https://classcheck.pcm.rnl.tecnico.ulisboa.pt/
[3] https://www.google.com/sheets/about/

EL can make use of the module's specific functions to manage different game elements.

GameCourse also has *Views*, where the state of the game is displayed to users. Each view can be composed of more views, creating more complex ones, and is highly configurable. A view is rendered by *Pages*, linked to *Templates* that describe how a view is displayed. Furthermore, each view is configured with the use of the EL and can have several *Aspects*. Aspects support GameCourse view customization and are the building block of how gamification is being applied. There can be multiple aspects for the same view, depending on the user's roles, and the information being displayed. This allows views to have various versions, and pages to be rendered differently depending on the type of user accessing them. Views, and ultimately pages, are built and customized in the View Editor (through the use of the system's dictionary).

Additionally, GameCourse counts with a Rule System, which is triggered by *AutoGame* whenever it imports new external data. AutoGame is a gamified rule-based system that uses a set of active rules to act on data and applies logic to calculate course awards and deliver them to students. The Rule System is composed of three entities: Sections, Rules, and Tags. Sections refer to groups that organize rules. Rules consist of text files with **Name**, **Description**, **When clause**, and **Then clause**. Tags (formed by name and color) are used to categorize rules within each section.

Lastly, rules creation and editions occur in the Rule Editor, which also counts on the dictionary's vocabulary and courses' metadata. The latter has global variables that can be used for all rules in a specific course. Since they are global, their value remains consistent throughout rule execution.

Focusing on GameCourse's UI, there have been some changes worth mentioning. In the past, GameCourse used outdated client-side technology no longer supported by Google. Therefore, a modern version of AngularJS[1] was implemented. To maintain style consistency throughout the platform, the Tailwind[2] styling framework was integrated. This contributed to a simple yet attractive interface composed of a top and side navigation bar and its respective content that is presented at the center of the screen. The top bar displays the user's profile, theme mode (light or dark), and a shortcut that redirects to the main page. The sidebar allows a deeper exploration of the system.

Once inside a course (Figure 4), and if the user is a "Teacher" and of type Admin, this sidebar allows navigation through "Users", "Course Operation", and "User Interface". Inside "Users", CourseUsers of that course can be visualized and managed, as well as the existing Roles. "Course Operation" shows all components of the course's behavior. In specific, "AutoGame", "Rule Editor", and "Modules". As for "User Interface", it shows all components regarding the platform's edition, namely "Pages", and "Themes" (pre-made course custom styles with the use of DaisyUI[3]).

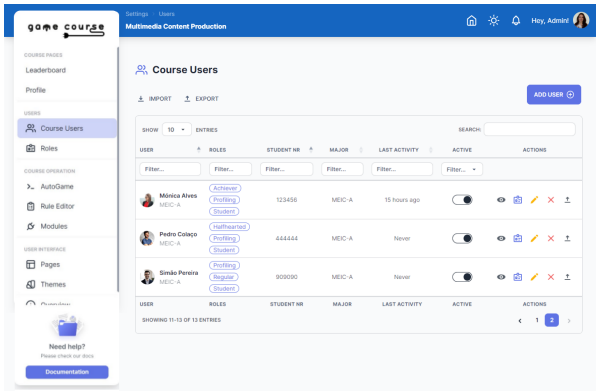

Figure 4: "Course Users" page UI

3.1. UI Case Study

Before working on new features and integrating them into the system, we took the time to identify existing GameCourse limitations and organize/plan this thesis work.

Focusing on the *Profiling Module*, it lacked an updated interface and did not match the rest of the system's aesthetic. We also identified an opportunity to enhance cluster assignment interactions. Previously, the profiling's configuration page was composed of two parts: "**Overview**", and "**Run the Profiler**". The former contained a diagram with a visual representation of the flow between different player profiles and the number of students clustered in each profile. The latter contained settings to run the *profiler* and the rendered of the results, which included two tables, one with all clusters results up until that point, and the other for the newest (latest) results when running the module.

This meant that whenever course admins wanted to analyze a student's progression and see if that student needed to be assigned to a new cluster, they were obliged to check the student's trajectory in the first table, and only then go to the second table to decide which was going to be his/her new player profile. We meant to change this last part so admins could access all of the results in one single table, avoiding the need to go across tables to search for information.

As for the Rule System, we saw the need to refactor the previous rule editor, which is the place where rules are edited. The previous version provided very little documentation and lacked user-friendliness. Hence, our goal was to update the ed-

[1] https://angularjs.org/
[2] https://tailwindui.com/
[3] https://daisyui.com

itor to support more guidance to users, and for it to match the ongoing aesthetic. Similarly to the Rule System, we also saw the need to refactor the previous *View Editor*, which is the place where views are edited. The previous editor was fully functional but lacked a user-friendly interface, making it difficult to interact with, even for users who already knew the system beforehand. Hence, we intended to update it with a simpler interface, which was also coherent and easy to use.

Regarding GameCourse's customizable UI, we wanted to update the system so it would support the adaptation process we intended to develop. This included creating different game element versions to be displayed differently depending on the type of student accessing the system. Moreover, we intended to create a Notification System to send personalized messages to students with suggestions regarding their ongoing performance.

4. Development

This section provides an overview of what was developed during the project. It describes updated features as well as newly introduced ones.

4.1. Profiling Module

Although this module had already been developed and used previously [10], it was one of the main components for this year's adaptation process in GameCourse. By using it, we could monitor how students changed during the course, and how they could be categorized into different player profiles, allowing us to have more tools for the adaptation process. Therefore, its interface needed to be updated as well as improved.

The new structure is now composed of two parts. "***Profiler***" and "***Overview***". The former has the module's configuration, the information regarding its last run, current status, and actions for running the profiler. The "***Overview***" section, on the other hand, includes a graph from Highcharts[1] JavaScript library, used for visualizing students' flow between clusters, and a table with all the results from the profiler.

Once the profiler is run, a table appears displaying not only all of the students' profiles but the latest results as well, in which course admins can select a new cluster for a student if needed, without needing to search the information across tables.

4.2. Adaptation

As mentioned before, we wanted to consider each student as an individual, with their traits and characteristics, and avoid having the "one-size-fits-all" approach. To achieve this, we used different versions of game elements adapted to each of the student's profiles. In previous years, GameCourse used BrainHex's questionnaire to assist said profiles, but this year we also included Hexad's.

To better understand those preferences, we created multiples *aspects*, which displayed different versions of game elements to students. We first performed a "best guess" adaptation, meaning that after running the *Profiling Module* for the first time, we assigned one of the game elements' versions for every student. This decision was already made in previous years, so we only integrated it into the system. Tables 1 and 2 show the game elements and their new versions, as well as what game element version matched which of the students' profiles. It is noteworthy to specify that *Regular* and *Achievers* are both highly participative, so we chose to give them the same versions.

At the time, we faced a challenge regarding the views, which were being restructured by Joana Sesinando [22], and some of them had bugs that needed to be fixed. As a consequence, we decided to delay the adaptation process, to prevent students from noticing views' bug fixing instead of new game elements' versions, which would jeopardize our adaptation process. Once we gave students their respective game elements' versions, we waited one week before starting the process. After that, we inquired them in three steps: first, we asked them if they noticed any changes to the UI during this period. Then, we asked them to specify what they thought had changed, and finally, we used a Likert scale to see whether they liked those perceived changes or not.

After this questionnaire, we also gave them the possibility to state their preferences, and therefore, change to another game element version if desired. However, this was not the only time when students were able to express their choices, as their game elements were updated weekly through the *Profiling Module*. This allowed us to gather data and see if our "best guess" adaptation was successful, and if correlations between game element's versions and students' profiles were correct.

In terms of UI, this adaptation process was divided into perspectives for students and admins/teachers. In the scenario where the user was a student, the page contained a table with all the available questionnaires. Once all questionnaires were answered, a new section appeared a the top of the screen, in which students could select their preferences.

In the scenario where the user was a teacher or an admin, the page would show all the configurations, which were organized in an *Adaptation table*. This table contained all available game elements for adaptation. Both teachers and course admins could enable or disable game elements, making the adaptation process active or inactive

[1] https://www.highcharts.com/

for said elements. Besides that, the table also contained statistics of the adaptation questionnaires answered by students.

Game Element	Version	Description
Leaderboard	LB001	Entire leaderboard with all course's students
	LB002	Snapped leaderboard (only 3 students above + 3 students below)
Badges	B001	Badges sorted alphabetically
	B002	Badges sorted with achieved first
Profile	P001	Contains graphs comparing user with everyone else in the course
	P002	Contains graphs comparing users with others with the same profile
	P003	Contains graphs with user's progress (no comparison)

Table 1: Game elements' and their versions.

Role	Game elements' versions
Achiever *and* ***Regular***	LB001, B001, P001
Halfhearted	LB002, B002, P002
Underachiever	LB002, B002, P003

Table 2: Different student's roles and the game element's versions they are assigned to.

4.3. Rule System

The Rule System has been developed in GameCourse for several years now and is responsible for dictating the behavior of each course through different sets of statements called *Rules*. Rules are used to retrieve actions from students and award them based on those actions. Said actions are the inputs of the Rule System and are generated in any of the places for which the data collection plug-ins are activated, such as posting on Moodle or rating a post. The awards, on the other hand, are the outputs of the Rule System and depend on the modules that are enabled in GameCourse. For example, only if the *Badges* module is enabled, the Rule System will be able to award badges, etc.

Regarding its mode of operation, the Rule System relied on text files to store rules since the beginning, which were later executed by Autogame. Each test file was constituted of a *Section*, that contained rules inside. This meant that whenever the rules were going to be executed or edited, the Rule System had to first access each of the files, and then parse it for it to be ready. This was a significantly time-consuming process that could be avoided. Because of that, one of the main objectives of the Rule System refactoring process was to replace the manipulation of text files for DB tables that could be easily accessed.

However, this replacement was not so straightforward, due to the file accessing and manipulation being deeply rooted across the majority of AutoGame Python scripts. Adding this to the fact that our priority became to create a visual editor for rules that all users could access, we ended up solely implementing the manipulation of sections, rules, and tags to be done through the DB, whereas the triggering and execution of the rules themselves - and therefore sections - were still left to work via files.

Regarding the new editor, we created three pages: The *Sections* page, which contains a list of all course sections for manipulation; The *Section's Rules* page, which contains all rules within a section and can be edited/created; The *Rule Edition* page, where rules are created/edited.

The last page displays several input fields for all rules' parameters. Whenever users enter this page, they are greeted with a small introduction to the Rule System. It can be used as a "Getting Started" tutorial, as it includes the main aspects and definitions of the System. Once users close the tutorial - which is always in the display if needed - the page is presented with fields for name, description, tags, and when and then clauses. The latter clauses use a code input component, by employing the *Codemirror*[1] framework. This element was meant for users to write the body of the rule by applying EL and GameRules functions. It was already used in the past but lacked visual feedback and guidance, so we refactored it and improved it.

4.3.1 Testing Environment

Once we refactored the Rule System, we created a new environment to test it online with real data. Hence, we transferred the information from the current MCP course (students, modules configurations, etc) into the new environment, except the information regarding *participation* and *awards* table. We ran the plugin modules to import new data into *participation* table, which then triggered autogame to award students and populate *award* table.

4.4. View System

The primary aim of the View System was to create a user-friendly, coherent, and above all, efficient editor. We had to reorganize our priorities and give precedence to creating a completed prototype, instead of starting to develop code without a detailed plan. We took inspiration from Canva[1] and Wix[2], in that they are both web-based platforms used to create visually engaging content. This gave us concrete ideas of what we wanted to achieve and how it would be integrated into the architecture.

Our prototype was elaborated using Figma[3] and counts with over 400 screens, and more than 2000 interactions. The Page Editor prototype is accessible for more details. This model consists of two

[1] https://codemirror.net/
[1] https://www.canva.com/
[2] https://www.wix.com/
[3] https://www.figma.com/

main screens: ”Pages” screen, and ”Page edition” screen. The former is used for the general manipulation of pages, and the latter for the edition of specific pages, or the creation of new ones.

4.5. Notification System

We wanted to create a Notification System to succinctly send personalized messages to students. Since we wanted to notify them about different things, we created four types of messages, depending on when and how they originated.

Adaptation questionnaires informed students about specific game elements that were opened for the adaptation process; **Dependencies missing** informed students that they needed to achieve a skill dependency from the Skill Tree before receiving a certain award; **Failed payment for skills** informed students about their lack of virtual currency to buy another skill submission attempt; **Aproaching a badge milestone** informed students about a badge that they were close to achieving.

5. Evaluation

This section describes the evaluation methodology of our work and details the results obtained.

5.1. Usability Testing

To evaluate the new features, we asked participants to perform a set of tasks and gathered some information for later analysis. We performed an A/B inter-group testing, meaning that some users tested the new version and others the old one. We asked them to perform 8 tasks for the new system, and 4 for the old. The disparity is caused by the adaptation process tasks, which only exist in the new version. We collected the following information: amount of time that took to perform each task; number and type of errors committed; whether the task was completed and successful;

Each task had a time limit that helped us determine if users completed the tasks within a reasonable amount of time. Tasks 1, 2, 4, (and also 5, 6, 7, and 8 in the new system) had a 4-minute time limit, whereas task 3 had a 20-minute time limit, in that it had a higher degree of complexity and required some thinking on behalf of participants.

Before starting, we asked them for consent to record both the screen and microphone and to answer a short questionnaire with their basic information. This included age, highest completed level of education, English level of understanding, previous contact with GameCourse, and previous experience with programming in Python. The level of English was important due to one of the tasks containing a substantial amount of written information, and the experience of programming in Python was important due to one of the tasks being the creation of a new rule in the Rule Editor.

Besides that, we gave them a small introduction, as well as the expected time to be spent overall. The tasks were categorized in **Modules' information and manipulation** - ***Skill Tree*** **and** ***Profiling Module*** - (tasks 1 and 2); **Rule Editor** (task 3); **Pages' Edition** (task 4); **Adaptation feature** (only for new system - tasks 5, 6, 7, and 8).

After completing each task, participants answered an *Individual Task Rating* to get their perceived overall task overload. Additionally, at the end of the testing procedure, they were asked to fill out a usability questionnaire, and a User Experience Questionnaire (UEQ), as well as a couple of final questions regarding their experience. Tasks were considered successful if they were provided with the right answers, and completed if they were finished within the expected time limit. Tables 3 and 4 summarize the data collected per task in both old and new systems.

	Task 1	Task 2	Task 3	Task 4
Completed	73.33%	26.67%	66.67%	86.67%
Successful	100%	0%	13.33%	87.67%

Table 3: Old version: Tasks' complete and successful rate.

	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8
Completed	73.33%	66.67%	80%	100%	86.67%	100%	100%	86.67%
Successful	86.67%	26.67%	60%	93.33%	100%	93.33%	40%	100%

Table 4: New version: Tasks' complete and successful rate.

By comparing common tasks between both versions in terms of success, we can see that, overall, the newest version contains more successfully completed tasks than the old one. However, one of the tasks had a considerably high rate of unsuccess in both versions. This is the case for task 2, in which participants entered the ”course users” page and chose one of the users with ”Halfhearted” as the current profile, ignoring their entire progress. Most participants understood that that was not the right answer and explored other pages, but only a small amount was able to find the solution. The old system's UI only made this task more difficult, in that none of the participants found the solution.

Besides that, Task 1 had a slightly worse success rate in the new system. The feedback provided by participants proved that the page did not have ”the best structure”, as it gives the illusion that it contains multiple Skill Trees that need to be accessed individually to edit them, but in fact, all the information is contained on the same page. This did not happen in the old version because the module was called ”Skills”, and did not have a list of existing Skill Trees once accessed, as it was only implemented to have one per course, which avoided such confusion. Lastly, task 7 on the new system was not very successful. All of them did the task correctly in terms of the interaction workflow, but they did not state the UI Leaderboard changes that we expected.

The UEQ measures Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, and Novelty. We used the Short version (UEQ-S), which contained eight items, and analyzed the results with UEQ's official tool[1]. We achieved positive outcomes, given that results show **the old version to be considered significantly worse across all items in comparison to the new one**.

5.2. Adaptation Process Results

The adaptation process gave us substantial data to analyze if the game elements' changes were accurate for students regarding their profiles and preferences. However, the number of students who participated was too low to support the conclusions taken in this experiment, given that **less than 20% of students** answered adaptation questionnaires and stated their preferences.

By analyzing what they had answered, their preferences, and comparing that to the preferences of their final clusters (assigned by the Profiling Module), we can assert that both *Leaderboard* and *Profile* game elements were proven to work, in that **81.81%** and **58.33%** of the students chose the same version preferences as what their "final clusters" were assigned. The *Badges*, on the other hand, only had **37.5%** of students who chose the same versions as what their "final clusters" were assigned. It is difficult to state why, given that their answers to what had changed were not accurate, and so, most of the students did not understand the current version they had when stating new preferences, mistaking it with the fixing of bugs.

Consequently, because the result includes less than half of the students who submitted preferences for the badge game element, we cannot affirm that our adaptation performance was successful in this particular case. While the remaining game elements were shown to be effective, the limited number of students participating prevented us from affirming the accuracy of our adaptation process, even though the results for those users showed favorable outcomes.

Additionally, we compared students' final clusters with their player profiles from Hexad's and BrainHex's questionnaire model. With this correlation, we saw that underachievers yielded consistent percentages in both models, making them inconclusive in determining which profile model was better. However, Hexad's model showed more accuracy for both regulars and halfhearted, whereas BrainHex's model only showed more exactness for achievers. **This confirms that Hexad presented more fidelity in the majority of clusters, and therefore, is considered to be better than BrainHex's model when predicting students' final clusters**.

5.3. Testing Environment Results

Once we set up the testing environment, we ran the data collection plugins modules and executed autogame to populate both **award** and **participation** table. As an attempt to compare the tables from the testing environment with the "real" environment, we created scripts that would assess this process. The scripts essentially executed outer joins between the participation table for the "real" and test environment, and repeated the same process for the award table, resulting in two data sets. One had the rows from the "real" environment that were not included in the testing environment, and one with the rows from the testing environment that were not in the real one.

For the comparison process, we removed the course ID (which was different in both environments but referred to the same course), the entry ID (which was only used for identifying data inside the DB), and the timestamps (which dictate when were *logs* imported into the system, and in this case differ). Lastly, we also replaced students' IDs with their full names, given that the IDs from the two DBs did not coincide. In summary, we used the name of the person, the source that imported the participation, the description and the type of participation, the post, and the rating and evaluator.

The results showed that **80.52%** of the "real" data from the participations table, and **62.31%** from the award table were not included in the testing environment. On the other hand, **32.71%** of the testing data from the participations table, and **11.36%** from the award table were not included in the "real" environment. **These results are alarming** in the sense that we expected to have identical data sets in both contexts.

6. Conclusion

The old GameCourse version implemented gamification as a method to increase the motivation of students, incentivizing them to interact with the system and perform specific tasks with which they would be rewarded. However, it did not consider each one of the student's preferences when providing them with game elements. This year's main goal was to create an adaptation process that would consider each student as an individual.

To assess this process we wanted to use the Profiling Module, which would analyze students' performance and assign them to a specific cluster weekly. This would help us pair game element versions with students' profile types. Moreover, we wanted to use both BrainHex and Hexad models to characterize students in player profile types at the beginning of the course. We intended to use that information to see if the students' game elements' preferences would match the cluster's preferences we predicted with the Profiling Module. Further-

more, we wanted to study which one these models worked best by comparing the Profiling's "final cluster" with the models' answers. Finally, we planned to continue refactoring different parts of the system, so the UI would match the current version's aesthetic while improving the UX.

With those goals in mind, we added new game elements' versions, which depended on the type of student profile, as well as the possibility to edit them specifying their preferences. At the end of the course, we were able to study if our attempt to create an adaptive UI was well received and, above all, successful. Overall, we received positive results across the majority of the game elements, however, there is still room for improvement. Especially for the *Badge* game element, which received the lowest percentage of accuracy results regarding the version students chose when compared to their final cluster. Moreover, by comparing the profiling clusters with the answers from both models, we determined that Hexad's model works better than BrainHex's, in that it shows more precision to the majority of clusters.

Additionally, we also refactored the Profiling Module and the Rule Editor, as well as other pages from the old system. To assess these alterations we conducted a user testing procedure. The collected data demonstrates that the changes were favorably received, confirming that the modifications done throughout this thesis improved GameCourse in terms of its users' interaction. Results also show that the new version was considered supportive, efficient, clear, and exciting, contrasting the results of the old version which classified GameCourse as more obstructive, inefficient, confusing, and boring.

7. System Limitations and Future Work

Even though a considerable amount of work significantly improved the system, several limitations could be overcome in the future. Regarding the Rule System, autogame's rule parsing could be streamlined by utilizing the DB instead of separate text files stored on the server. Furthermore, the addition of a debugger and the "preview rule" actions would be a useful addition to the current editor. Regarding the testing environment, the current work demonstrated that it was not properly functioning. **The next critical and top-priority step is to understand why this occurred**, as this environment was intended for further system development testing and must function properly.

Regarding the view system, the high-fidelity prototype should be integrated into GameCourse's ongoing version. Since the editor's UI is meant to significantly change in the current version, testing how it will be received before implementing it would be ideal. For the notification system, it would be useful to have some administrator configurations, such as which type of notifications can be sent to users, or the addition of new messages.

Besides that, the ongoing UI can still be improved. This is particularly true for the profiling module's results and the skills in the skill tree module. For the former, it has been demonstrated that very few users notice it within the "Profiling module" configuration page, making its relocation beneficial to user-system interaction. The latter could be improved so that each of the existing tree's configurations - i.e. skills and tiers - are moved inside their specific trees, and not in the overall module configuration page.

Finally, regarding the adaptation process, it would be interesting to repeat the same procedure with more students, given that this year only a small number of them participated in the experiment.

References

[1] G. Barata, S. Gama, J. Jorge, and D. Goncalves. Engaging engineering students with gamification. In *2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES)*, pages 1–8. IEEE Transactions on Learning Technologies, 09 2013.

[2] G. Barata, S. Gama, J. Jorge, and D. Gonçalves. Improving participation and learning with gamification. Gamification '13, page 10–17, New York, NY, USA, 2013. Association for Computing Machinery.

[3] R. Bartle. Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of MUD Research*, 1(1):1–27, 06 1996.

[4] C. Bateman and R. Boon. *21st Century Game Design (Game Development Series)*. Charles River Media, Inc., USA, 08 2005.

[5] I. Bouchrika, N. Harrati, V. Wanick, and G. Wills. Exploring the impact of gamification on student engagement and involvement with e-learning systems. *Interactive Learning Environments*, 06 2019.

[6] S. Deterding, D. Dixon, R. Khaled, and L. Nacke. From game design elements to gamefulness: Defining "gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, MindTrek '11, page 9–15, New York, NY, USA, 2011. Association for Computing Machinery.

[7] L. Diamond, G. Tondello, A. Marczewski, L. Nacke, and M. Tscheligi. The hexad gamifi-

cation user types questionnaire: Background and development process. In *Workshop on personalization in serious and persuasive games and gamified interactions*, pages 229–243, London, UK, 10 2015. Association for Computing Machinery.

[8] M. Duke. Personalization of web services: Opportunities and challenges. *Ariadne*, (28), 06 2001.

[9] J. Díaz-Ramírez. Gamification in engineering education – an empirical assessment on learning and game performance. *Heliyon*, 6(9):e04972, 2020.

[10] A. Gonçalves. GameCoursePersonal. Master's thesis, Instituto Superior Técnico, Lisbon, Nov. 2021.

[11] M. Hajarian, A. Bastanfard, J. Mohammadzadeh, and M. Khalilian. A personalized gamification method for increasing user engagement in social networks. *Social Network Analysis and Mining*, 9(1):47, Aug 2019.

[12] E. Lavoué, B. Monterrat, M. Desmarais, and S. George. Adaptive gamification for learning environments. *IEEE Transactions on Learning Technologies*, 12(1):16–28, 04 2018.

[13] L. Legault. *Self-Determination Theory*. Springer International Publishing, 06 2017.

[14] D. Liu, R. Santhanam, and J. Webster. Toward meaningful engagement: A framework for design and research of gamified information systems. *MIS Quarterly*, Forthcoming(4), 11 2016.

[15] A. Marczewski. *Even Ninja Monkeys like to play*, volume 1, page 28. CreateSpace Independent Publishing Platform, 10 2015.

[16] A. Marczewski. *User Types HEXAD*, pages 65–80. 10 2015.

[17] L. Nacke, C. Bateman, and R. Mandryk. Brainhex: A neurobiological gamer typology survey. *Entertainment Computing*, 5, 01 2013.

[18] S. Reyssier, S. Hallifax, A. Serna, J.-C. Marty, S. Simonian, and E. Lavoué. The impact of game elements on learner motivation: Influence of initial motivation and player profile. *IEEE Transactions on Learning Technologies*, 15(1):42–54, 2022.

[19] L. Rodrigues, W. Oliveira, A. Toda, P. Palomino, and S. Isotani. Thinking inside the box: How to tailor gamified educational systems based on learning activities types. In *Brazilian symposium on computers in education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 30, page 823, 11 2019.

[20] A. C. G. Santos, W. Oliveira, J. Hamari, and S. Isotani. Do people's user types change over time? an exploratory study. *5th International GamiFIN Conference 2021 (GamiFIN 2021)*, 2883, 04 2021.

[21] S. Schobel, M. Schmidt-Kraepelin, A. Janson, and A. Sunyaev. Adaptive and personalized gamification designs: Call for action and future research. *AIS Transactions on Human-Computer Interaction*, 13(4):479–494, 12 2021.

[22] J. Sesinando. GameCourse – The Ultimate UI. Master's thesis, Instituto Superior Técnico, Lisbon, June 2023.

[23] A. Shroff, S. Barde, and A. Verma. Improve the learning capability of students at university level by using gamification. In *2021 IEEE International Conference on Technology, Research, and Innovation for Betterment of Society (TRIBES)*, pages 1–4. IEEE, 2021.

[24] F. Teixes. Yu-kai chou (2016). actionable gamification: beyond points, badges and leaderboards. octalysis media: Fremont. ca. *Revista Internacional de Organizaciones*, page 137, 06 2017.

[25] A. Toda, P. H. Valle, and S. Isotani. *The Dark Side of Gamification: An Overview of Negative Effects of Gamification in Education*. Springer International Publishing, 08 2018.

[26] G. Tondello and L. Nacke. Validation of user preferences and effects of personalized gamification on task performance. *Frontiers in Computer Science*, 2:29, 08 2020.

[27] G. Tondello, D. Valtchanov, A. Reetz, R. Wehbe, R. Orji, and L. Nacke. Towards a trait model of video game preferences. *International Journal of Human-Computer Interaction*, pages 1–17, 04 2018.

[28] G. F. Tondello, R. R. Wehbe, L. Diamond, M. Busch, A. Marczewski, and L. E. Nacke. The gamification user types hexad scale. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '16, page 229–243, New York, NY, USA, 2016. Association for Computing Machinery.