

Pingo - An Online Portal for Educational Games with Customizable Content

Denise Notini Nogueira, Luiz Chaimowicz, Raquel Oliveira Prates
 Computer Science Department, Federal University of Minas Gerais
 Belo Horizonte, Minas Gerais - Brasil
 denotini,chaimo,rprates@dcc.ufmg.br

Abstract—Many educational games do not offer mechanisms for teachers to adapt their content to the curriculum applied in the classroom. In general, this is a barrier for the adoption of many educational games in the school context. This paper describes the design and implementation of a children’s educational game system that aims to provide teachers the possibility of customizing game content, trying to reduce the lack of pedagogical content adjustment. The system was implemented as an online portal for educational games, integrating different technologies that combined supports the games content customization. This portal was designed to offer access to three different types of users: game developers, teachers and students. Each one of these groups has unique roles and can perform actions that, in a whole, collaborate to the main educational objective: to assist in teaching and learning using fun and motivational aspects provided by computer games.

I. INTRODUCTION

Educational games provide opportunities for children to interact with a cognitive domain [1]. Through them, new knowledge can be introduced indirectly with also the amusement offered by this kind of media.

The first works that tackled learning using digital games have appeared within the past 20 years [2]. Since then, many theoretical and practical studies on how to use games to convey educational content have been conducted.

The use of educational games has the characteristic that the learning experience is formed by a specific goal: every educational game has a different way of teaching, and what is transmitted confers the uniqueness of the instructional purpose. There is no doubt that it is possible to learn with games, and many works in this area focus on the effectiveness of learning through fun offered by them.

Games evoke behaviors in their players that are favorable to learning, such as persistence, risk-taking, attention to detail and problem solving skills [3].

However, there are practical barriers in using educational digital games frequently cited by researchers and educators (usually related to the constraints within an educational setting): short lessons, physical space, variations in game competence among students, installation, costs and preparation time for teachers to learn the game [2]. There is also a reluctance among teachers and educators to use the games, and a lack of knowledge about how to use them.

A research conducted through literature review of papers that deal with the use of digital games for educational purposes

in the classroom [4] also pointed elements that dissuaded their use by teachers and educators. Through a qualitative analysis of papers, six areas were identified as barriers to adoption of educational games in the classroom and are going to be listed in next section.

Among them, one important barrier identified is the lack of alignment between the objectives of commercial games and curricula adopted by the teachers - problem also identified during the evaluation of a game developed in a previous work [5]. Aiming to minimize this problem, we developed a portal for educational games with customizable content.

This portal, named **Pingo** (portal of instructional games, online), aims to be a repository of games of various disciplines and enables teachers of Brazilian schools to customize their content and use them with students, assisting in strengthening the content taught in classroom.

Customization is an user centered activity that allows changes in the environment by its occupants [6]. This is a process that involves changes in functionality, interface, information content or distinctiveness of a system by its users [7].

Other problems such as low graphics quality and lack of time available in schools for activities involving games were also considered during the system design. The portal tries to reduce them through the technology employed in its games (Unity 3D) and the platform adopted (Wordpress).

Pingo allows designers to submit various educational games, and they can explore their imagination and creativity to develop a variety of games which content can be customized by teachers.

Each one of the games, regardless of their discipline, demands a certain set of parameters for its customization. Therefore, the portal allows game developers to define which contents of their games must be customized by teachers through the system interface. Using these specific informations, the system generates different interfaces for each one of the games, enabling teachers to insert educational content to be reinforced or learned by their students.

In this paper, our focus is to present this portal, emphasizing the offered opportunity of game’s content customization. Next section explains the motivations that led our work and Section III presents related works that discuss solutions to provide content customization tools in games to be applied by teachers. In Section IV, Pingo’s implementation is described, and the used technologies are discussed. Section V discuss the

results we have achieved so far, and in the last section we present the conclusions and possibilities for future work.

II. MOTIVATION

Digital games induce certain conditions in their players that are favorable to learning: satisfaction, desire, defiance, interest, enjoyment, pride in achievement, among others. Using games combined with teaching in the classroom increases student motivation and consequently increases attention and retention of the content [8] apud [9].

The integration between fun and learning causes a greater immersion of players in the educational content presented, making them to assimilate more effectively, and encouraging them to learn from their own mistakes. The simple process of learning objectives, restrictions, and controls of a game is usually a task that requires several cognitive skills.

Because of these reasons, there is a growing number of schools and teachers adopting the educational games as a new playful way to convey knowledge to children, and they appear as powerful tools to enhance learning [10].

With an increasingly early age, children begin to have contact with many different technologies through routine tasks such as using TVs, phones, computers, video games, generating behavioral profiles that encourage teachers to use different media.

Playful exercises have always been present in school. Through fantasy, children are encouraged in their symbolic capacity, and now computers are proving increasingly present as an aid in the teaching process.

The use of console video-games in schools is less studied than the computer ones, since the latter unlike the former have more flexibility at intervention and development, not to mention that they are based at resources (computers) that are often already available in schools. Generally, console video-games typically offer content that can not be easily changed by end users, and for this reason, our work focuses on educational computer games.

Barriers to classroom implementation of educational games were expressed at the same time that interest in these new tools increased [11], [2], [4], [3], [12], [13].

A work surveyed, through literature review, the use of digital games with educational purposes in the classroom [4]. The objective of the study was to identify elements and/or features of educational games that discouraged educators to adopt them. Through a qualitative analysis of papers, six areas were identified as barriers to implementation of educational games in schools:

- 1) Negative perceptions among teachers and educators about the games.
- 2) Low graphics quality mainly games developed in an academic context.
- 3) Lack of adequate hardware in school to run newer gaming software.
- 4) Lack of instructional time in school periods to adequately engage in rich, cognitive video games. for activities involving games.

- 5) Lack of affordance within artificial environments to properly represent desired learning objectives.
- 6) Lack of alignment for objectives within commercial gaming environments to state and local standards.

Greater acceptance of the use of educational games as a tool to aid in the task of teaching is hampered by these points mentioned. Our work tries to minimize problems (2), (4) and (6), with greater focus in the latter, as will be discussed in Section IV.

Another motivation for the development of this work arose during the evaluation of a children's educational game [5] with school coordinators. This game, named Papa-Letras, is inspired by the classic game Pac-Man (Figure 1). The player must capture in a maze, the syllables that constitute the image shown in its center, while avoiding monsters that walk randomly on the scenario. The game provides teachers logs with information concerning the successes and mistakes made at the game.



Fig. 1. Papa-Letras [5]. The player must capture the correct syllables from the image displayed in the center of the maze, while avoiding the monsters attack.

During Papa-Letras's evaluation in a Brazilian private school with teachers and coordinators, was pointed a yearning for greater manipulation of educational content by teachers in the game, since in many cases, the fixed educational content was not the same as the one taught in the classroom (problem 6). They suggested that the teacher could be able to define the linguistic aspect approached, e.g. spelling of words.

We realized the necessity to build a tool that, in addition to providing the ability for teachers to customize game content, would be also robust enough to contain a repository of various games from different disciplines of knowledge (languages, mathematics and others) and would not require manual installation in each school, allowing the use in any school in the country with internet access.

Thus, we developed a portal that allows the insertion of several educational customizable games targeted to children, and proposes to solve some of the problems identified for the use of games in schools by teachers and students.

III. RELATED WORKS

We observed that in the literature there are not many works that propose a customizable content solution to educational games. This section presents those most related to our work.

The work [14] proposes an educational gaming system that is modular and extensible, and its users (teachers) are able to associate their lesson plans to games, which will then be available to students.

For its implementation, the authors used a layered architecture. The first layer is the website and database, which makes the interface between the teacher and the system. In this layer the teacher can enter a lesson plan, then customizes three different types of games (Flash Card, In Game and Mini Game). Teachers can also track statistics of their students through visual representations such as graphs and tables. The second layer is an abstraction between the database and website. It optimizes the queries to the database, trying to minimize requests, storing data for the layers that use it. The third layer is the one that manages the game, and was also designed to be an abstraction. Through it, new games can be added to the system by the system's programmers. The only requirement for the games is that they have to communicate with the second layer.

This system only allows insertion of math games, because the content to be customizable by the teacher is fixed. The interface has tools for the teacher to define mathematical problems and their solutions, but there are no mechanisms that enable other types of customization. Despite that possible changes in the system to cover other disciplines (e.g. languages, sciences and others) are not complicated due to the modular characteristic, it is always needed the intervention of a system programmer. In addition, the games become dependent of a predefined structure (problems and solutions), which may greatly restrict the creativity in the development of new ones. The focus in the design of this system did not consist of improving the aesthetics of the user interface, but in the core that allows the customization proposal.

In other study [15], a framework was also implemented for enabling teachers (of students that are approximately 7 years old) to customize games.

It was observed that many games in Malaysia, sometimes imported from other countries and deployed in schools, did not covered the local syllabus. This problem was due to cultural differences that occur from country to country. Variations in accent were also present in these games, causing greater difficulty of understanding by children. Many games produced in the country did not meet the heuristics used for the evaluation of educational games. Given these problems, it was observed the necessity to provide a mechanism for teachers to tailor the games and focus on the specific characteristics of their students (their biggest difficulties for example). The games's disciplines include Mathematics and English.

Another work about customization for games [16] consists in developing an engine for creating games for mobile devices (cellphones). These games can be created by users (teachers, who will use them as support for their courses) in a relatively easy way, using different teaching and content styles.

The aim of the project is that modifications can be performed by users who do not have programming skills (more specifically the Java ME, with which the engine was developed), through an interface that provides ease of use and learning.

The customizable contents of the games are editable through some defined structures, such as quizzes, decision trees and multimedia. In each one of these structures, the game creator can enter data and define flows of interaction for the player, based on the scenario on the map, similar to RPG games (Role Playing Game). However, the changes that can be made, as in the previous work mentioned in this section, are already pre-defined in the system, through those structures mentioned.

Various are the benefits for teachers with customization of educational games for children: the educational content is completely chosen by the one who has the most property in deciding what content students should or should not play and learn, and this content provides entertainment through games, causing greater stimulus to children.

The works described allow the provision of games that can be customized by teachers to their students, however, the customizations must follow a structure previously defined by system's designers. In order to create new structures, or games in other cases, the system itself must be changed.

IV. PINGO

Pingo was developed using the open source, web platform, Wordpress [17], a content management system (CMS) based on PHP and MySQL. Although WordPress is more often used for creating blogs, it enables the development of regular sites due to its high extensibility, made possible by plugins, themes and PHP programming. The motivation for choosing the Wordpress platform, instead of starting a new development through other web programming frameworks, was the features already present in the platform that would not needed to be re-implemented (e.g., users and session control, database integration), reducing the cost with development time and ensuring the concentration of effort with the concept and logic of the portal itself. In addition, Wordpress provides a great documentations, a large amount of plugins, and content management interface (GUI) of the website, with possibility to perform more specific modifications by editing PHP code.

Developing the portal with a web platform ensures that Pingo can be accessed either at school, in classes at a computer lab, or at home. This solution is designed to minimize the problem (4) of lack of time in schools for activities involving games, since students are not restricted to use the portal only in available time at schools, because those who have internet access at relative's home, or in lan houses, will be able to play games at Pingo.

Pingo integrates the game engine Unity 3D [18] which enables the creation of 2D or 3D games with high quality graphics. Several companies use it to develop commercial games, such as Ilusis (<http://www.ilusis.com.br>), Cartoon Network, Coca-Cola, Disney, Electronic Arts, LEGO, Microsoft, NASA, Nexon, Nickelodeon, Square, Ubisoft, U.S. Army, Warner Bros., large and small game studios, independent professionals, students and hobbyists [19].

Unity 3D has a large community around the world, with nearly two million registered developers, and four hundred thousand monthly active. It has an integrated development environment (IDE) that eases the creation of games for consoles, mobile devices, desktop platforms and web browsers. Unity is mainly used for creating games for mobile devices and web browsers (using Unity Web Player plugin).

Wordpress has an integration with the Unity Web Player plugin. Thus, games developed in Unity can be integrated within the pages of websites developed on the platform. This integration is made possible by publishing an Unity project in a format compatible for the Unity Web Player plugin, and then deploying the file in Wordpress.

Due to the quality of games that can be created, the popularity, the wider community and documentation, and integration with Wordpress, the Unity was chosen as the game engine to be used with Pingo. We hope that, with the integration between the portal and Unity, it become easier to get volunteer developers to insert their games on the portal, and that the problem of low quality of graphics in educational games (problem 2) is minimized.

The Unity Web Player plugin (<http://unity3d.com/webplayer/>) must be installed on every user's computer of to enable the games to be executed. On the homepage is a link to the page where the plugin can be downloaded. This is the only requirement that should be performed by users, and only a single installation enables all browsers to run the games (similar to Flash technology from Adobe Systems, in which a single installation activates content to all browsers). The minimum requirements for the plugin installation are:

- Operating Systems Windows XP/Vista/7/8, browsers Internet Explorer, Firefox, Chrome, Safari, Opera;
- Operating System Mac OS X or newer, browsers Safari, Firefox, Chrome.

In Figure 2 is shown an example of how a game is integrated to the portal, within Pingo's pages.

A. User Groups

There are three groups, or roles, of users in the portal (teacher, student and game developer). Each role limit what an user can perform on the system. Users must authenticate using an unique login and password combination. When authenticated, the pages are displayed according to the user role. For safety, users can not access the pages that are not concerned for their own role and each user has only one role. Figure 3 presents an outline of the different pages and which role can access them.

Each user role has a FAQ page specific to its needs. Users not logged into the system, also can access help information through posts on a blog.

Only teachers are able to register on the portal by themselves. Students are registered by their teachers and developers are registered by those responsible for the portal (to ensure that only authorized developers - and previously instructed - submit games to the system). When developers are registered in the

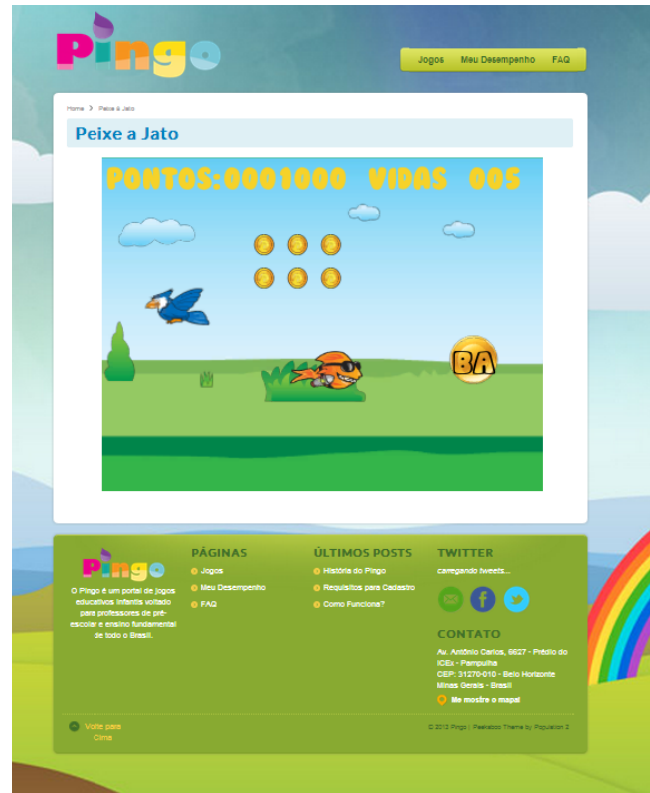


Fig. 2. Games made at Unity 3D run on Pingo's pages.

system, they get via e-mail, along with data access, guidelines for creating suitable games that properly communicate with the portal.

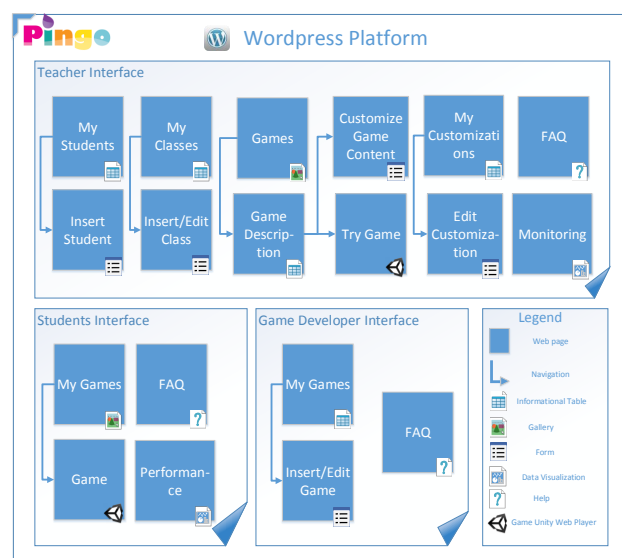


Fig. 3. Pingo was developed with Wordpress and provides different interfaces according to the role of the logged user: teacher, student or game developer.

1) Game Developer Interface: Game developers can register their games, detailing information such as title, description,

identifier, group age and discipline recommended, and inserting images which represent the game.

These data are displayed to teachers when they are choosing games that will be customized. During submission, they must also include the game file (extension .unity3d), compiled for the Unity Web Player format, and must also specify the schema that defines the parameters of the game that must be customized by the teacher. This schema is going to be defined and detailed on Subsection IV-C.

Game developers can consult their list of registered games, as well as their moderation stage. This list is paged, and a search engine allows the user to find the desired information. Game information can be changed (while the game is not published for teachers and students), including the schema that defines the customization, and each time a change is made, the moderation process is started over.

2) *Teacher Interface*: Teachers must first register all their students in the system, providing a single login, name, surname, e-mail (of the child, or parent) and password. The provided e-mail will receive all the necessary access information. A list of students is displayed along with a search engine, and when many records are returned, the results are paginated.

Then, they should create their students classes by entering the name, description, and which students belong to them. There is a restriction that a student can only be associated with a single class, it means that Pingo does not allow previously associated students to be joined in a new class. Classes may be altered at any time, and a paged list with search engine is also offered to the teacher, who also has the option of removing a class. Removing them does not remove the associated students.

All games available on the system can be viewed by teachers, and filtered by discipline or recommended group age. The game list contains the name and a indicative picture, and when selected, a page with all game description is displayed (Figure 4). The teacher can see the game description, the discipline for which it was designed, group age, and images that indicates the features of the game. The teacher can try the game, playing it with a standard content to better understand the gameplay and how the educational content will be displayed to students. If the teacher is interested to make the game available to his students, he must then customize it.

Teachers make the games available for their classes through customizations: each game customization is related with a class, and the students of this class will automatically have access to these games with content customized by their teacher. Until a customization is made, the game is not accessible by students.

Customizations are also displayed on paginated lists with search engine, and can be changed or removed at any time. To the same class and the same game, the teacher can only perform one customization. Classes that already have an associated game customization can not have another one for that same game.

Teachers can also monitor students and classes statistics through graphs and charts that are periodically updated.

3) *Student Interface*: Students can access the system as soon as they receive the e-mail with access information.

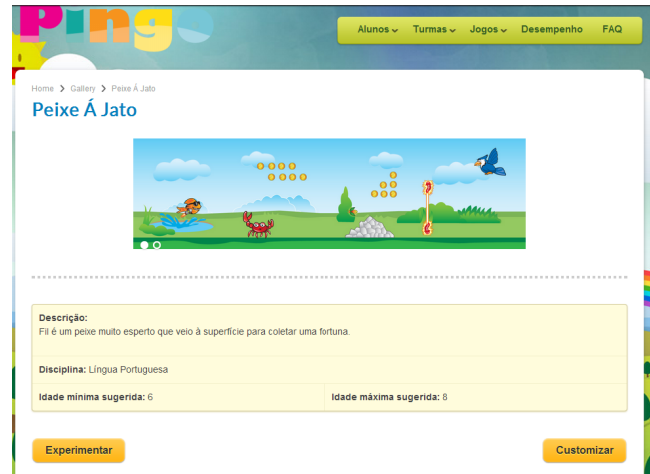


Fig. 4. Game's description page. The teacher can read the information, navigate to the game page or to the customization page.

They have access to a list of available games, but unlike the teachers, this list does not contain all the games in the system, but the games that the teacher who registered them customized to the class in which they belong. If the student is not yet associated with any class or their class does not have any customized game, no game will be listed.

When interacting with games, data such as time spent, score, level reached, hits and mistakes are stored to be monitored by the teacher, or in their own performance page.

4) *Guest Interface*: Guests are all users that are not logged into the system. For these users the interface appears similar to a blog, where news and informations are displayed for the teachers, schools, educators, parents and developers that are interested in Pingo. Guests can also contact the responsables for the portal through a contact form. Even when logged, all users can access the blog and contact page.

B. Database

Wordpress integrates with MySQL database. We added new tables to the database application to allow game content customization and the user interface development (Figure 5).

We created the tables *class*, *game*, *customization*, *score* and *hits and mistakes* to enable the persistence of the game customization logic.

Student and *game developer* tables are extensions of the default *users* table in Wordpress. Students table stores which teacher did the registration and with which class is associated, and game developers table has a contact telephone number.

C. Content Customization

It is not possible to anticipate during the design, all kinds of customization that a game may require, especially considering that the portal will host games from various disciplines for different educational levels. Therefore it is necessary to allow the definition of what can be customized in the game by its developers. This definition affects directly on how the teachers will fill the customization form for each game.

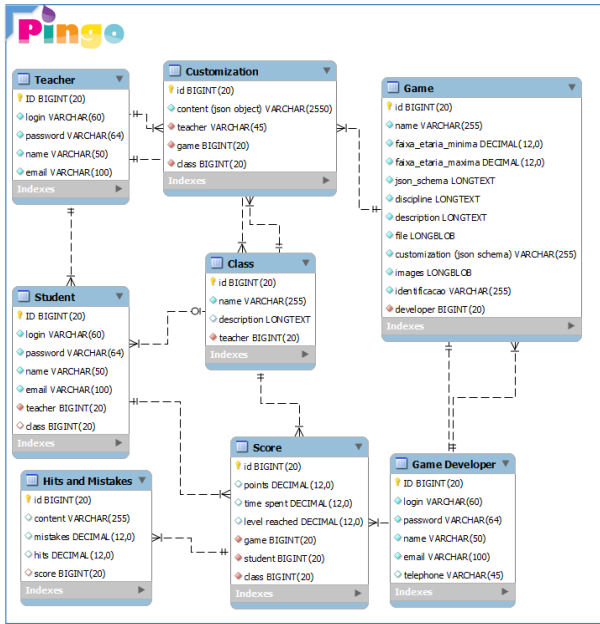


Fig. 5. Pingo's Entity-Relationship Diagram.

To exemplify how a game is customized at Pingo, we will use the first game created: the Jetpack Fish (Figure 6), a 2D game developed at Unity 3D. It was designed as a proof of concept of the proposed concept that enables games content customization by teachers. Its main objective is to test the customization feature and the dataserver layer provided by Pingo.



Fig. 6. Game Jetpack Fish with content defined at the JSON object from Code 2.

Jetpack Fish is a game to help learning portuguese language, and focuses on the study of syllabic division of words. The child interacts with the game through the character Fill, a very greedy fish who decided to leave the water and go to the land to collect treasures. At each level the fish must collect coins corresponding to the correct syllables that represent a picture and avoid the wrong syllables that make lose points.

The player must be careful to not collide with enemies, and can only advance to the next level when all the syllables are captured in Fill's treasure. The main character has animations for the actions of walking, running, flying, falling and dying. Enemies (birds, crabs) are also animated. When correct syllables are captured by the player, the score is incremented, and when the wrong syllables are captured, it is decremented.

In the game's design, it was decided that the following parameters must be customized by teachers: images, correct and incorrect syllables. The teacher can adjust this information in accordance with the curriculum given in the classroom, for example, to teach the proper writing of fruits, the teacher can customize the game by inserting pictures of pineapple, apple, melon, pear, etc., and their syllables. Thus, their students when playing, will have fun and exercise the spelling of those words that were taught in class.

```

1  {
2  {"Type": "object",
3  "$ Schema": "http://json-schema.org/draft-03/
  schema"
4  "Required": true,
5  "Properties": {
6  "Words": {
7  "Description": "Every level has a word."
8  "Title": "Words"
9  "Type": "array",
10 "Required": true,
11 "Items": {
12 "Type": "object",
13 "Required": true,
14 "Properties": {
15 "Image": {
16 "Type": "string",
17 "Required": true,
18 "Description": "image url"
19 "Title": "Image"
20 }
21 "Right_syllables": {
22 "Type": "array",
23 "Title": "Right Syllables"
24 "Required": true,
25 "Items": {
26 "Type": "string",
27 "Title": "Right Syllable"
28 "Required": true
29 }}
30 "Wrong_syllables": {
31 "Type": "array",
32 "Title": "Wrong Syllables"
33 "Required": false,
34 "Items": {
35 "Type": "string",
36 "Title": "Wrong syllable"
37 "Required": false
38 }}}}

```

Code 1. JSON Schema that defines the necessary customization in the game Jetpack Fish.

The games, as mentioned, are created by their developers externally to the portal through the game engine Unity 3D. This engine allows the creation of games in 2D or 3D, and

once they are finished, they can be published with an extension compatible with web browsers, making it possible to interact with the game directly by Pingo's pages. The game developer, to include the game at Pingo, must provide its executable file, and then perform the steps to define the customization that is offered by the game. For this, we opted to use JSON Schema [20], a data format descriptor language, readable by humans and machines. This schema defines which data can be entered by the teacher, specifying type, name, description, whether the field is mandatory, minimum and maximum quantities, among others.

Because each game has different customization requirements, it has also a specific schema. The game developer is the only one capable of defining the exact type of customization that his game needs, so, he is the one that, when registering the game, includes the JSON Schema.

From the customization schema defined by the game developer, the portal automatically generates a form (using the open source framework inputEx [21]) (see Figure 7) that converts the data filled by the teacher in a JSON object (see code 2), which is sent to the game through a datasever layer.

This interface is the result of the definition inserted by the game developer when submitting a game, and if he changes the schema definition, the form reflects the modifications, without the need for an intervention from system's programmers. What can be specified through a JSON schema is converted into a form, and consequently into JSON object.

Fig. 7. Jetpack Fish customization form.

Figure 8 presents the necessary steps that must be accomplished in order to make a game content customizable.

Returning to the example of the game Jetpack Fish, the developer sets its schema defining that the teacher must include data of the following types: an image, and correct and incorrect syllables (code 1). For the image, it is necessary for the teacher to specify the path (a URL) of the image, and this path must be of type *string*. The correct and incorrect syllables are two arrays of strings. In Figure 9 is shown the form generated through the game JSON schema filled with a customization. This customization is represented by the JSON object from code 2 and this object is sent to the game that then displays the content inserted by the teacher to his students.

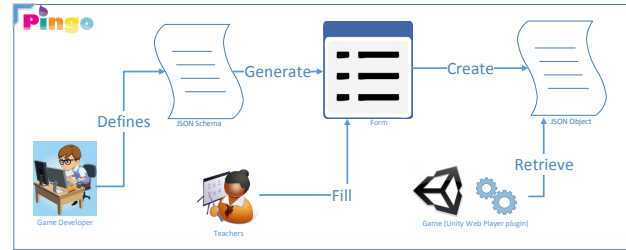


Fig. 8. Necessary steps required for a game content customization.

Fig. 9. Jetpack Fish content customization form with content inserted by the teacher.

```

1 {"Word": {{
2 "Image": "http://imagem.com/pera.png"
3 "Right_syllables": ["play", "ra"]
4 "Wrong_syllables": ["ba"]
5 }}}

```

Code 2. JSON object related to content inserted at the form from Figure 9.

D. Communication between Pingo and Unity 3D

The portal has a dataserver layer (Figure 10) which is an interface between games and the portal database, and with it is possible to retrieve the customized contents created by the player's teacher. This layer is also used to save the statistics of interaction with the game, for each player: score, level reached, mistakes, hits, time spent and other. The games must have scripts that communicate with the layer, recovering customized content and inserting statistics.

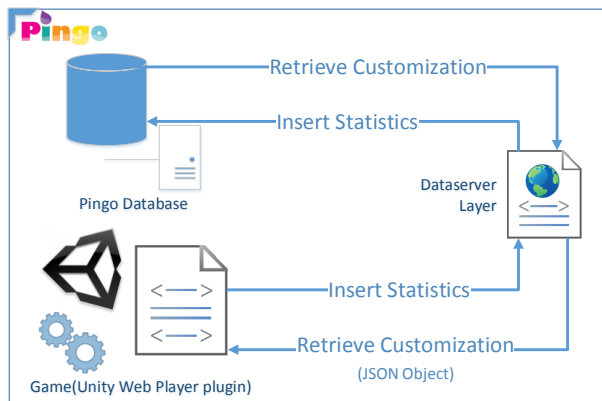


Fig. 10. Dataserver layer.

This dataserver layer, written in PHP, is a security module that prevents games to perform direct queries on the portal database that could alter or retrieve information that is not of particular interest to the game logic. Using the layer, the games do not perform database access. They perform requests to the URL of the layer which then perform the necessary operations.

To also avoid games from unauthorized developers to perform requests to the layer, each developer gets a secret key. SQL Injection attacks are also prevented.

Games must provide the following required parameters to request the customization: identifier of the game (the same that must be provided during the registration of the game at the game developer's interface), player's login, and hash code of the previous parameters concatenated with the secret key. To insert player's statistics, it must provide the following: identifier of the game, player's login, score, and hash code of the previous parameters concatenated with the secret key.

The game retrieves the login of the player through a native method from Wordpress that provides the user login stored on session of the game's page.

The layer compares the hash code sent as parameter with the internally produced using the secret key concatenated with the identifier, login and score, also sent as parameters. If the codes are different, it means that the secret key used for encryption is not correct, then the layer does not perform the required operations. If both codes are identical, the validation was successful, then the operations are performed.

The two operations performed by the server layer are:

- 1) **Insertion of player's statistics.** Hash code, identifier, player's login and score are received as parameter. If

the hash validation succeeds, the layer adds a new entry to the *score* and *hits and mistakes* datatables, associating with the user sent as parameter.

- 2) **Retrieve customized content.** The hash code along with the identifier and login are received as parameter. If the hash validation succeeds, the layer recovers from Pingo's database which student class the user is associated with, and then which is the customized content from the game supplied as parameter and the student class identified. The layer returns the JSON object inserted by the teacher when he fills the game customization form. The game then uses this object to perform content manipulation.

In short, game developers receive, from those who are responsible for Pingo, an URL of dataserver layer location, and a secret key. With these two unique information, games are capable of communicating with the layer.

Scripts for communication with Pingo made to Jetpack Fish were implemented in JavaScript language. They can and should be reused for all subsequent games. They are:

- **Retrieve Customized Content-** To request to dataserver layer the customized content;
- **Insert Player's Statistics-** To enter score, time spent, level reached, mistakes and hits to the server layer;
- **Function MD5 (utility class) -** To generate the hash code with an MD5 function;
- **JSON Object Parser (utility class) -** To parse a JSON object (retrieved by the dataserver layer) into JavaScript arrays.

During future games implementation through Unity IDE, it will be necessary to add these classes in the game project. They are shared with the game developer when the access is granted by e-mail.

In Code 3, written in JavaScript, is displayed a snippet that illustrates how a request is made to the dataserver layer. Through an url concatenated with the parameters, a www request is made, and the response sent by the dataserver layer is received by the variable `hs_get`.

```

1 retrieveJsonUrl var = "http://layerurl/
  retrieve_json.php?";
2 var gameId = "peixeajato";
3 private var secretKey = "secret";
4
5 var hash = md5functions.Md5Sum (name + gameId
  + secretKey);
6 var retrieveJson = retrieveJsonUrl + "&
  idusuario=" + name + "&idjogo=" + idJogo +
  "&hash=" + hash;
7
8
9 hs_get = WWW (retrieveJson);
10 hs_get yield;

```

Code 3. Sample script to the Unity project in JavaScript that performs a request to Pingo's dataserver layer.

This solution increases Pingo’s safety because, as displayed in Code 3, the game does not perform direct database access, only requests information on a layer that is an interface between the portal and the game.

E. Game Moderation Process

Developers are encouraged whenever possible to create their games with the supervision of a specialist in the discipline of the game (mathematics, languages, geography, etc.). Games must not contain violence, and the vocabulary used must be appropriate for children. Another important point is that the game should only be submitted to Pingo after tests and inspections, to prevent users to find implementation bugs that could easily have been discovered with a validation process. To ensure that these conditions were followed in the game development, we apply a moderation process before publishing them to teachers.

This moderation process comprehends four stages: *pending*, *change requested*, *approved* and *rejected*. These stages are shown in the interface that lists games submitted by the developer.

When a game developer add a game to Pingo, or make any changes in a game already added, an e-mail is sent to the responsables for moderation process. At this point the game is at *pending* stage.

The game is then analyzed, and features like graphical quality, vocabulary, effectiveness of educational content transmitted, etc. are observed. If any change is required, the developer shall observe the stage *change requested* in the list of submitted games. If the game meets all the requirements the stage is changed to *approved*. When the game does not meet the desired profile, it is then *rejected*.

Once approved, the data inserted at the game registration can no longer be altered, since the game may be already in use by students and teachers, and a change could cause data integrity problems.

V. RESULTS

A math game developed for the portal named Glutton Caterpillar was the first experience of a new game created and submitted by a game developer at Pingo [22]. This game was developed especially for Pingo, and accomplished all the processes required for a game publication at the portal: game creation, data communication between game and portal, customization definition and game moderation.

The game consists of guiding a caterpillar to collect little pieces that will form the entire length of her body (see Figure 11). Each one of these little pieces has a unit of measurement associated, implying that the capture of the little pieces must be in ascending order. The amount of little pieces to be collected, as well as which units of measurement are exhibited, are defined by the teacher.

This game is already published on Pingo and can be chosen by registered teachers for customization.

Some difficulties have been identified for the submission of this new game. The first one was the lack of a test environment to validate the game developed in Unity. Currently, it is

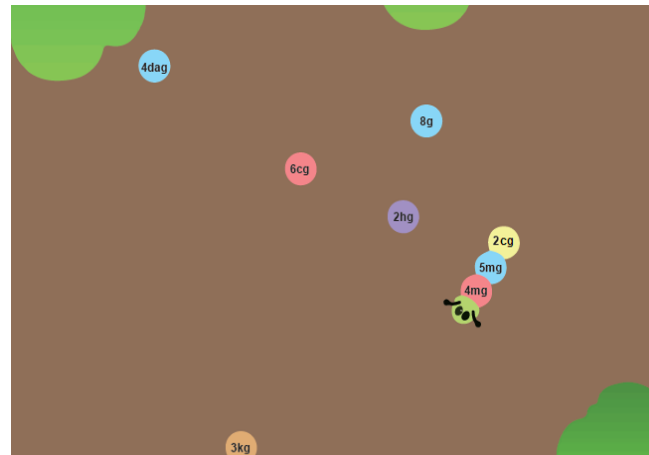


Fig. 11. Math game: Glutton Caterpillar.

not possible to retrieve a default JSON object during the game development, to validate this aspect before submitting to the portal. Communication errors with the dataserver layer can only be traced by those responsible for Pingo, and this responsibility should be transferred to the creator of the game.

Another perceived problem was the lack of appropriate feedback to the developer, since he can not know how the customization form will be generated at teachers pages from the Json Schema he provided for the game. Also, the game developers are not able to perceive potential errors in their schemas, since there is not a validation of schema syntax.

VI. CONCLUSION

In this paper we presented Pingo, a portal that offers, for teachers who adopt the system as a teaching aid in the classroom, content customization for various educational games. The portal was developed as a web platform that can be accessed from any computer with internet access, and the only requirement for its use is an installation of Unity Web Player plugin that enables the correct execution of the games in teachers and students computers. Game developers who have an interest to create a game for Pingo, receive access to the system, so they can submit their games and define how the customization have to be performed by the teachers.

Despite the existence of many works in the area of educational games, few are those that allow the customization of educational content. Approaching this aspect, our work offers contributions through the architecture and combination of technologies, and the portal itself.

The architecture implements an integration of game engine with an external environment that manages various aspects, as well as customization, such as class, students, monitoring, among others. Our proposal removes the responsibility from the game developer to create internal interfaces at the game that allow the edition of educational content by teachers, which could be a very complex task because of users access roles: it should restrict students of having access to the mechanisms of customization beyond the scope of the development of a game. The portal reduces the complexity removing this responsibility from the developer and transferring it to the external system

that manages the game using a simple concept of customization definition through a descriptive language.

The ability to customize games content is the most significant contribution made through the portal as it tries to reduce the barrier that exists in the adoption of educational games for teachers in schools, because most do not fit the curriculum followed in the classroom. With Pingo, teachers of children can choose, from the various games, those that suit best and can adapt their content to the needs of the students.

Another important contribution is the possibility of monitoring student performance, which is necessary since students' interaction with the system occurs outside of the school environment. This kind of data enables decision making towards the use of games or even the teaching in the classroom, since it can indicate which contents students have more difficulty and may be reinforced in the classroom.

The fact that any game developer could create an educational game through Unity and then add it to the portal is an easement for Pingo to have a variety of games, and in a reasonable quantity. Unity is quite widespread in gaming community, especially by independent developers.

To increase understanding of how the system should be used, we will record videos with step-by-step instructions on how to perform the different tasks for each of three groups of users.

To further facilitate the work of teachers, we will create a mechanism for exporting and importing customizations. With it, teachers can reuse customizations already made, for example those which have proved themselves effective, and the teacher could use them again at other classes. Avoiding a repetition of data input, the mechanism would allow importation and exportation of customizations from one class to another.

To improve feedback from the system to game developer, will be implemented a preview of the form generated from the JSON schema. Currently, developers do not know how the schema will be transformed into a form for teachers and a visual representation can help developers to prevent mistakes before they happen.

To avoid possible typos or syntax errors of JSON schema defined by the game developers, will be created a validation, preventing games with incorrect schemas to be submitted at Pingo.

A necessary improvement to be implemented related to the games available for children is the recovery of the previous score before loading the games. Currently, the scores are reset to zero with each new interaction. The dataserver layer must be modified to support a new action: recover existing score for the current player.

An important goal is to populate Pingo with several games, covering many different disciplines and age groups. We are going to redevelop Papa-Letras [5], following all the necessary steps for its integration with Pingo.

After performing all the desired changes, we will publish Pingo for the general public, aiming to have all contributions materialized with its effective use by teachers of Brazilian schools.

ACKNOWLEDGMENT

The authors would like to thank CAPES, CNPq and Fapemig for supporting their research.

REFERENCES

- [1] M. E. Gredler, "Educational Games and Simulations: A Technology in Search of a (Research) Paradigm," in *Handbook of Research for Educational Communications and Technology*, D. H. Jonassen, Ed. MacMillan, 1996, no. 39, ch. 17, pp. 521–540.
- [2] A. S. Egenfeldt-nielsen, "Overview of research on the educational use of video games," vol. 1, pp. 184–213, 2006.
- [3] E. Klopfer, S. Osterweil, K. Salen, J. Groff, and D. Roy, "Moving Learning Games Forward," Tech. Rep., 2009.
- [4] J. W. Rice, "New Media Resistance : Barriers to Implementation of Computer Video Games in the Classroom," *Journal of Educational Multimedia and Hypermedia*, vol. 16, pp. 249–261, 2007.
- [5] D. Nogueira, C. Coscarelli, L. Chaimowicz, and R. Prates, "Papa Letras : Um Jogo de Auxílio à Alfabetização Infantil," in *Simpósio Brasileiro de Jogos e Entretenimento Digital (SBGAMES)*, 2010, pp. 170–174.
- [6] S. S. Marathe, "Control & agency in customizable video games," in *International Communication Association*, Montreila, Canada, 2008.
- [7] J. Blom, "Personalization: a taxonomy," in *CHI '00 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '00, vol. 49, no. April, ACM Press. ACM, 2000, pp. 313–314.
- [8] BECTA, "Computer Games in Education Project," Tech. Rep., 2001.
- [9] T. Connolly and M. Stansfield, "Using Games-Based eLearning Technologies in Overcoming Difficulties in Teaching Information Systems," *Journal of Information Technology Education*, vol. 5, pp. 459–476, 2006.
- [10] J. P. Gee, *What Video Games Have to Teach Us About Learning and Literacy*. Palgrave Macmillan, 2003, vol. 1, no. 1.
- [11] A. McFarlane, A. Sparrowhawk, and Y. Heald, "Report on the educational use of games: An exploration by TEEM of the contribution which games can make to the education process," Tech. Rep., 2002.
- [12] M. Kebritchi, "Factors affecting teachers' adoption of educational computer games: A case study," *British Journal of Educational Technology*, vol. 41, no. 2, pp. 256–270, Mar. 2010.
- [13] R. F. Kenny and R. McDaniel, "The role teachers' expectations and value assessments of video games play in their adopting and integrating them into their classrooms," *British Journal of Educational Technology*, vol. 42, no. 2, pp. 197–213, Mar. 2011.
- [14] J. Clark, A. Fletcher, T. Fritz, and J. Schreiber, "Modular educational game system: A customizable framework for learning," in *2011 16th International Conference on Computer Games CGAMES*, Department of Computer Engineering and Computer Science, University of Louisville, Louisville, Kentucky 40292 USA. IEEE, 2011, pp. 248–253.
- [15] Z. C. Embi, "The implementation of framework for edutainment: Educational games customization tool," in *Information Technology 2008 ITSIm 2008 International Symposium on*, vol. 1. IEEE, 2008, pp. 1–5.
- [16] S. Mininel, F. Vatta, S. Gaion, W. Ukovich, and M. P. Fanti, "A customizable game engine for mobile game-based learning," in *Systems Man and Cybernetics 2009 SMC 2009 IEEE International Conference on*, no. October. IEEE, 2009, pp. 2445–2450.
- [17] Last Access June/2013, <http://pt-br.wordpress.com/>.
- [18] Last Access June/2013, <http://www.unity3d.com/>.
- [19] Last Access June/2013, <http://unity3d.com/company/public-relations>.
- [20] Last Access June/2013, <http://json-schema.org/>.
- [21] Last Access June/2013, <http://neyrice.github.io/inputex/>.
- [22] N. S. de Lima Leão, "Desenvolvimento de um jogo para auxílio ao ensino da matemática," UFMG, 2013.