An Assessment Engine: Educators as Editors of their Serious Games' Assessment

Yaëlle Chaudy, Thomas M. Connolly, Thomas Hainey
University of the West of Scotland, Paisley, Renfrewshire, Scotland

Yaelle.chaudy@uws.ac.uk
Thomas.connolly@uws.ac.uk
Thomas.hainey@uws.ac.uk

Abstract

Serious Games (SG) are increasingly used by educators to assist the teaching and learning process and offer many advantages over traditional education. They are highly engaging, motivating and they have the potential to adapt to each student quickly becoming an ideal supplementary tool for education. However, if the majority of teachers agree that using SGs increases the motivation, learning and retention of their students, very few of them are ready to trust their assessment to verify that the learning goals have been met. They would rather adopt a more conventional method such as a paper-based test.

We believe two main reasons explain this attitude: a lack of ownership over the games used and the rigidness of the games, making them unmodifiable by the teacher. To overcome these issues, we have developed an assessment engine to be used by both SG developers and educators. The engine's design results in a separation of a game and its assessment, and the resulting modularity allows the teachers to modify the assessment of a game even after distribution.

This paper focuses on the teacher interface of the assessment engine. After reviewing the literature associated with in-game assessment and learning analytics, the paper will provide a summary of the engine and its functionalities, present the Learning Analytics (LA) dashboard. We will then describe the visual language that allows teachers to edit a game's assessment system based on the LA reports. Thereafter, the authorisation and versioning mechanics of the engine will be detailed, showing how the system regulates the access to the games and stressing the fact that every teacher will have, after modification, a unique game customised to their students' needs. Finally, we will provide conclusions and state the remaining work to be undertaken.

1. Introduction

Serious games (SG) are increasingly used by educators to enhance the teaching and learning process. Various studies about the educational use of SGs show empirical evidence of increased learning and retention (Squire et al., 2004; Girard, Ecalle and Magnan, 2012) as well as motivation (Rosas et al., 2003; Annetta et al., 2009). However, if many teachers now agreed to use SGs as a complementary tool to their usual teaching method, they would not agree to use it as a full-fledge assessment tool. Yet, the potential of SGs in terms of assessment is well recognised, they offer many options for assessment. Using real-time assessment data, a serious game can be adapted to a learner's needs (Conlan et al., 2009; Kickmeier-Rust et al., 2008; Göbel, 2009). A serious game also offers the possibility of formative assessment and feedback according to a player's actions (Jarvis and de Freitas, 2009). In addition, a serious game can have assessment logic embedded into its core mechanics and offer non-invasive assessment (Kickmeier-Rust et al., 2008; Shute, 2011).

So why is there a lack of trust towards a game's assessment? We believe there are three key reasons for this. Firstly, the teachers do not feel sufficiently in control of the assessment, they do not have a sense of ownership over the SG and might feel threatened by a game rather than supported by it. Secondly, the SG is not very flexible as it is often distributed as a "Black Box", meaning it is self-contained, cannot be modified and data is very hard to retrieve from it (Torrente et al., 2010). In traditional teaching, improvisation and adaptation are key aspects of the teachers' role (Hunt, 1976) and they tend to lose this capacity with the introduction of a tool they cannot modify to suit the needs of their students. The third and final reason is the lack of detailed reports on the gameplays that would provide educators with an insight about what the students learn through games. Learning analytics (LA) would give them access to the information they need, however, a Horizon report (Johnson et al., 2013) presents it as a technology expected to be adopted in a two-to-three year timeframe and they recognise that it is still beyond reach of most teachers.

Our research addresses these three issues by introducing an assessment engine (Chaudy, Connolly and Hainey, 2013) to be used when developing SGs. A visual editor then gives the teachers the ability to modify the assessment to suit their students' needs even after the end of the development process.

This editor addresses the first two issues noted above; the third is addressed by including an LA dashboard to the engine interface so the teachers are provided with detailed information about their students' performance. Based on the reports generated, educators will be able to make informed decisions about the modification of the assessment.

This paper will first review the literature on assessment, learning analytics and authoring tools available in SGs. It will summarise the engine's functionality and present the LA dashboard. Then the visual editor will be introduced in Section 5 illustrating, with an example, what can be changed and how. It will also explain how the system deals with all the created versions of a game.

2. Assessment, serious games and learning analytics

In this section, we summarise important educational concepts and show how they are used in SGs.

2.1 Assessment

Allen (2004) defines assessment as "*The use of empirical data on student learning to refine programs and improve student learning*". It is an essential part of the teaching and learning process; teachers need it to ensure they achieved their teaching goals and it helps students monitor their progress. There are three key principles any assessment should follow: validity (does the assessment measures what it is supposed to?), reliability (is the assessment accurate and consistent?) and practicality (can the assessment be realistically applied?) (McClure, Sonak and Suen, 1999). From an earlier literature review (Chaudy, Connolly and Hainey, 2013), we classified approaches to assessment integration in SGs into 6 categories:

- *Monitoring of states:* States of the gameplay are monitored and associated to a probability of achieving a learning goal.
- *Quests:* Assessment is part of the game as a quest that can be assessed when completed.
- *Use of an assessment model or profile:* Assessment is based on an existing model or profile.
- *Non-invasive assessment:* The student is unaware the assessment is taking place. The game can also adapt to the player.
- *Quizzes:* Quizzes (i.e. multiple choice) are integrated at various stages of the game.
- *Peer assessment:* The assessment is performed by peers (i.e. forums).

2.2 Feedback

Feedback and evaluation are closely related concepts. Hattie and Timperley (2007) define feedback as "*information provided by an agent […] regarding aspects of one's performance or understanding*". Feedback is part of the 10 things that most influences the achievement of the learner according to Hattie and Gan (2011). They explain that feedback occurs at four different levels:

- *Task level*: Give information about the correctness of the answer.
- *Process level*: Give information about the process that led to the answer.
- *Self-regulation level*: Give information relevant for self-assessment, for deeper understanding.
- *Self level*: Do not give information about the task, but about the learner (e.g. "Well done!").

In the Chaudy, Connolly and Hainey (2013) literature review feedback was analysed and defined according to the following terminology:

- *Invasive*
    - *Guidance feedback*: Hints are given to the player
    - *Performance feedback*: Confirmation or information
    - *Immediate feedback*: "just-in-time" feedback triggered on response of an action
    - *Delayed feedback*: provided at the end of the game or a level
    - *Learning outcome feedback*: triggered by a low or high score.
- *Pervasive*
    - *Game adaptation feedback*: The player's action triggers a modification in the game.

2.3 Learning analytics

Brown (2011) defines learning analytics (LA) and its purpose: "*At its core, learning analytics is the collection and analysis of usage data associated with student learning. The purpose of LA is to observe and understand learning behaviors in order to enable appropriate interventions*". Data from the gameplays of various SGs are collected and data mining allows conclusions to be drawn about the games and possible improvements to be identified. The conclusions can be of various types such as time spent on particular tasks, answers given to questions or final scores. Learning analytics differs from game analytics (i.e. data mining on every interaction within the game) as it focuses on the actions meaningful for the learning process. The 2013 Horizon report (Johnson et al., 2013) presents LA as a technology expected to be adopted in a two-to-three year timeframe. It also outlines various advantages in using learning analytics to overcome some of today's challenges in e-learning. Johnson et al. (2013) also argue that LA is "*an effective, efficient way to assess student responses, provide immediate feedback, and make adjustments in content delivery and format*", however, they recognise that LA is still beyond the reach of most teachers.

3. Literature review: Authoring tools in SGs

3.1 Search terms and criteria

A literature review was conducted to identify different approaches to authoring tools in SGs. The authoring of assessment in particular was stressed but we also selected papers discussing authoring of games to include more papers. The literature review was performed using the following search terms:

("authoring tool" OR "visual editor") AND ("serious game" OR "educational game") AND assessment

Relevant papers were identified using the following two criteria: papers from 2004 onwards and papers discussing an authoring tool for educators. Papers presenting editors not related to SGs were excluded. Where possible, the search was based on abstract, titles and keywords to focus on relevant papers.

3.2 Databases searched

The electronic databases searched in this review included those identified as relevant to education, information technology and social science: ACM (Association for Computing Machinery), ASSIA (Applied Social Sciences Index and Abstracts), Cambridge Journals Online, Index to Theses, Oxford University Press (journals), Science Direct, EBSCO, ERIC (Education Resources Information Center), IngentaConnect, Emerald, Springer and IEEE (Institute of Electrical and Electronics Engineers).

3.3 Results

The search returned 553 studies, however, after applying the two criteria only 8 relevant papers were identified, 7 of them showing empirical evidence of the usability of their system. The studies discovered in the literature review are presented in Table 1, categorised according to the type of visual editor they present:

- *Text-based user interface*: The user is asked to describe the game/assessment using text only. This can be done using a Domain-Specific Language (DSL) or existing configuration languages such as eXtensible Markup Language (XML).
- *Form-based user interface*: The user edits or creates the game/assessment using forms. Text fields, selects, upload buttons etc. are used.
- *Visual Editor***:** The user creates a visual representation of the game/assessment. This can be done using drag-and-drop options with flow or state diagrams.

Table 1: Papers found on authoring tools in SGs

Study	Description and target users	Assessment	The editor	Empirical evidence on usability
Text based				
Rodríguez-Cerezo, Gómez-Albarrán and Sierra (2013)	Proposes a model where developer and instructor work together to design and implement exercises and simulations models. A tool is created for authoring exercises based on these models. The model was made for engineering teachers.	Assessment reports are designed in the model conception stage and are generated for each student playing the simulation.	Tool uses a Domain Specific Language.	Case study performed over a three years period with engineering instructors. Informal interviews were used. Data was collected with both educators and students.
Form-based				

Sorensen and Ramachandran (2007)	Explains the authoring tool of the *SimCore* framework for development and modification of training scenarios. The tool is aimed at healthcare trainers.	Editor has a section for the definition of solutions and error rules as well as hints and feedback.	Tool is a user interface with text fields, selects and tabs for every component of the game. The editor generates a number of XML files.	Experienced healthcare trainers have tested and helped improve the framework.
Tornero et al. (2010)	Describes *e-Training DS*, an authoring tool for integrating games in e-learning management system (LMS). Teachers can create games by customising existing generic ones. The tool is aimed at teachers in general	Assessment and tracking of performance is configured within the editor based on provided assessment logic and interaction types. During the game, information is communicated to the LMS that stores it.	The tool supports different text fields and upload options according to the game template chosen. A preview is also available.	Case study conducted, the time spent creating and modifying the game and its assessment was measured to prove that the process is feasible for an educator.
Porayska-Pomsta et al. (2013)	Presents *SHARE-IT*, an authoring environment for customising games. Activities can be added from provided templates. The tool is aimed at teachers and parents of autistic children.	There are existing rules that can be used for each game created.	The editor is simple and visual; the user adds components by selecting them and filling fields. The tool generates an XML file that is used by Unity.	n/a
Visual Editor				
Mehm et al. (2012)	Presents *StoryTec*, an authoring tool for SGs. Its main interface is used by game designers, artists and educators. A story Editor is made available to the latter, the structure of the game is created or modified from an existing template. The tool is aimed at teachers in general	StoryTec has an ActionSet Editor that handles every action made by the player; actions related to assessment are provided. The Knowledge Space Editor describes all the learning outcomes of the game.	The interface is rather complex, with various panels showing different views of the game (view, state diagram, object browser and property editor). Drag and drop is used to create new objects and display them.	Experiment focused on the interface usability. Three professional game developers used the tool and were interviewed. A study also conducted comparing StoryTec to e-adventure.
Torrente, et al. (2010)	Presents the game editor <*e-adventure*>. This tool enables teachers to create serious games for their students. The tool is aim at all teachers.	Assessment (mainly quizzes and quests) can be configured.	Numerous panels to show every step of the game creation. Data is modified with forms or handled using a visual language. The tool generates an XML file.	Various studies conducted using <e-adventure> for game creation.
Gaeta et al. (2014)	Presents the ALICE project and its storytelling design model along with an authoring tool used to create a story for a game and configure the activities made available.	Assessment events are handled and can be plugged into the story plot. Activities must be selected from the library.	Editor contains three panels, a design area, an activities library and a properties area. The solution produces XML.	Experiment evaluated usability and effectiveness of tool. Questionnaires based on the System Usability Scale.

Marchiori et al. (2012)	Describes an instructor-oriented game-authoring tool that focuses on narrative: WEEV. The tool is aimed at teachers in general	The user can add correct and wrong reactions to events in the story and have points associated.	The game story is represented by a state diagram. The user can add and modify the objects and actions available in the story.	Three evaluations performed; one formative to improve the software, one to evaluate the usability and a third to evaluate the creation of a game.

4. EngAGe : an Engine for Assessment in Games

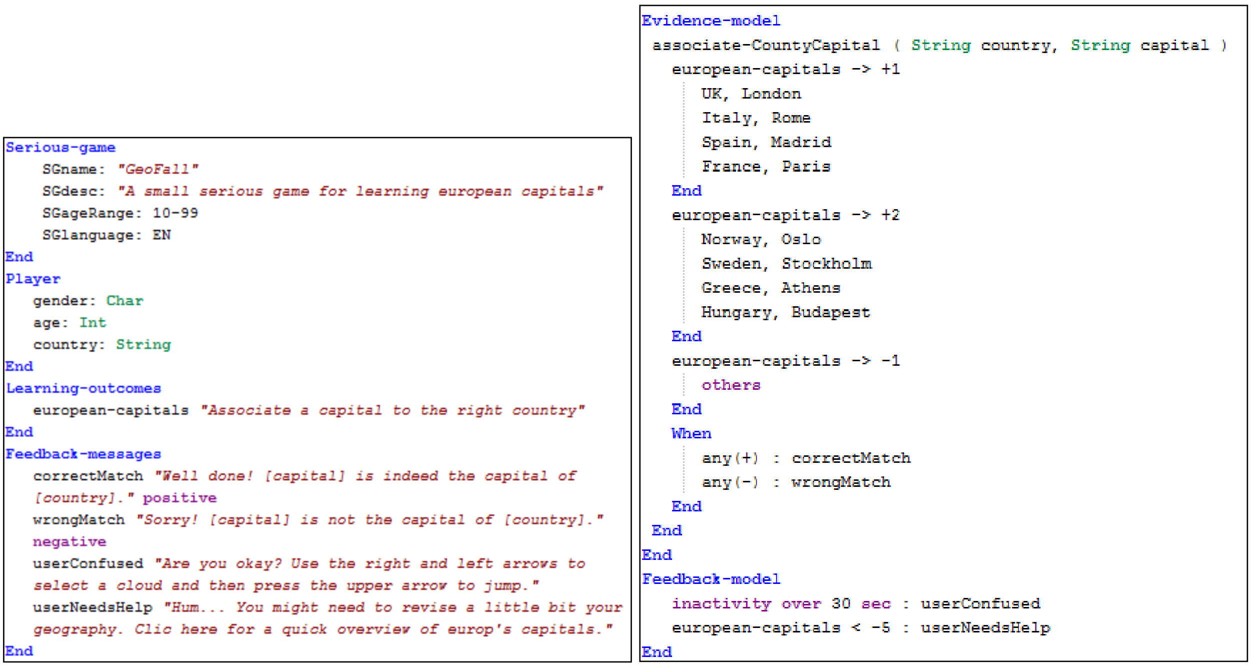
The assessment engine is presented in more details in a previous publication (Chaudy, Connolly and Hainey, 2013) and this paper will only provide a brief summary essential for understanding the following section. EngAGe allows developers to save a considerable amount of time and cost not only with implementing the assessment process into their game, but also with thinking it through. One of the goals of the engine is to provide game developers and teachers with a tool for assessment as well as with guidelines on the assessment features that should exist in their game. Importantly, EngAGe also allows the separation of the SG's game mechanics and its assessment logic, making the whole system more flexible. The engine specifications include the following:

- A Domain-Specific Language (DSL) to be used as an assessment configuration descriptor.
- A set of web services to parse the DSL and perform the assessment.

4.1 EngAGe during the development process

4.1.1 Defining the assessment

The first important step in developing a game with EngAGe is the definition of the assessment. This is specified in a single configuration file, independent from the game, and a Domain Specific Language (DSL) was defined for this purpose. The DSL has its own semantics and syntactic rules; it embeds the domain (assessment in SG) knowledge into the programming language. The expertise of assessment is, consequently, situated in the engine and no longer in the game. The DSL is based on the educational concepts defined in Section 2 and on the analysis of existing SGs (Chaudy, Connolly and Hainey, 2013). This language can be seen as guidelines to the assessment process. Its intention is to encourage SG developers to include details of the assessment that otherwise they might have omitted if they were implementing it themselves. An example DSL for a game associated with European capitals is provided in Figure 1.



```
Serious-game
    SGname: "GeoFall"
    SGdesc: "A small serious game for learning european capitals"
    SGageRange: 10-99
    SGlanguage: EN
End
Player
   gender: Char
   age: Int
   country: String
End
Learning-outcomes
   european-capitals "Associate a capital to the right country"
End
Feedback-messages
   correctMatch "Well done! [capital] is indeed the capital of
   [country]." positive
   wrongMatch "Sorry! [capital] is not the capital of [country]."
   negative
   userConfused "Are you okay? Use the right and left arrows to
   select a cloud and then press the upper arrow to jump."
   userNeedsHelp "Hum... You might need to revise a little bit your
   geography. Clic here for a quick overview of europ's capitals."
End
```

```
Evidence-model
 associate-CountyCapital ( String country, String capital )
   european-capitals -> +1
      UK, London
      Italy, Rome
      Spain, Madrid
      France, Paris
   End
   european-capitals -> +2
      Norway, Oslo
      Sweden, Stockholm
      Greece, Athens
      Hungary, Budapest
   End
   european-capitals -> -1
      others
   End
   When
      any(+) : correctMatch
      any(-) : wrongMatch
   End
 End
End
Feedback-model
   inactivity over 30 sec : userConfused
   european-capitals < -5 : userNeedsHelp
End
```

Figure 1: Example of a configuration file using the DSL.

4.1.2 Performing the assessment

At the creation of the game, the developer will send the configuration file to the engine. The engine will populate its internal database and will make a set of web services available for the developer to use in the SG. These web services are capable of processing a specific communication protocol to perform the game's assessment. The engine will return an id the game should use when invoking the web services. During gameplay, the SG will communicate with the engine to perform the assessment and retrieve feedback. The set of web services includes:

- *startGamePlay:* creates a new gameplay session and returns its id.
- *assess*: assess the actions made by the player based on the rules specified in the configuration file, it updates the scores and triggers feedback (if any).
- *getFeedback:* returns the feedback (if any) to be triggered.
- *getLog:* returns all actions performed by the player and all feedback triggered. This is useful for the debriefing phase of the game, the final summative assessment, but also throughout the gameplay to adapt the game to the player.

4.2 The learning analytics dashboard

Unlike developers, who can deal with the engine directly and retrieve meaningful information from it, the teachers need more help and guidance. A web interface, associated with the engine, has been developed for this purpose. It provides them with a visual report synthesizing all the data collected from the various plays of their games: the learning analytics report. The learning analytics section of the interface largely depends on the configuration file as this describes a player's meaningful characteristics, learning outcomes, actions and feedback.

In our European capitals example the report would display data about the pairs of country-capitals selected by players, the feedback messages that were triggered and the overall score for the learning outcome *european-capital*. At present, the learning analytics dashboard will only allow educators to refine findings by gender, age or country as they are the only known values for the players.

The learning analytics report allows for flaws in a game's assessment to be identified. A teacher could discover, for instance, that a task is too difficult or ambiguous if the majority of players got it wrong. Another quest or question could be considered unhelpful if all the players answered it correctly and in a short amount of time. Thanks to the learning analytics reports, the teacher can make an informed choice regarding the modification of the SG's assessment.

4.3 Summary

The architecture is service-oriented comprising an application server, various web services and a database. Figure 2 illustrates the three main users of EngAGe: developers, teachers and players.

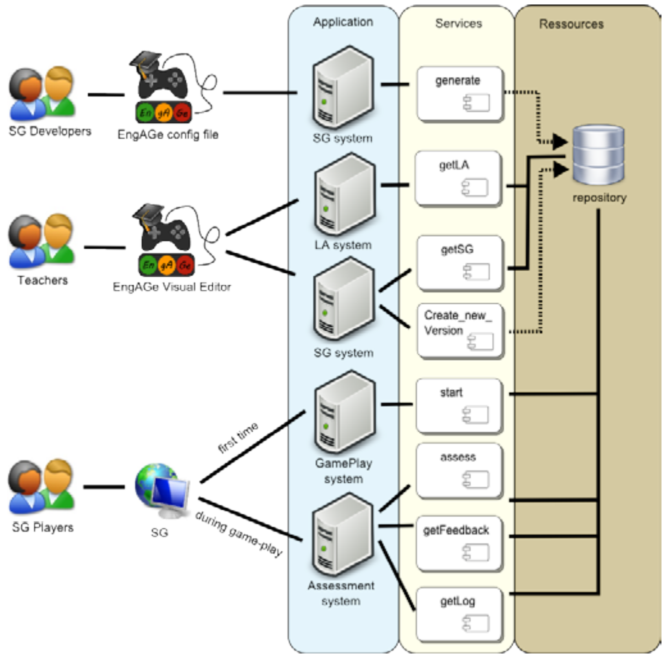


Figure 2: Overview Diagram of EngAGe

5. The visual editor: An authoring tool for teachers

If a game was developed using EngAGe, its assessment will be modifiable by the teachers. Once students start playing the game teachers will be able to review the LA reports, identify changes to be made to the assessment and use the editor to modify the assessment without the need to ask the developer. This section presents the editor and the versioning system.

5.1 What can be modified and how?

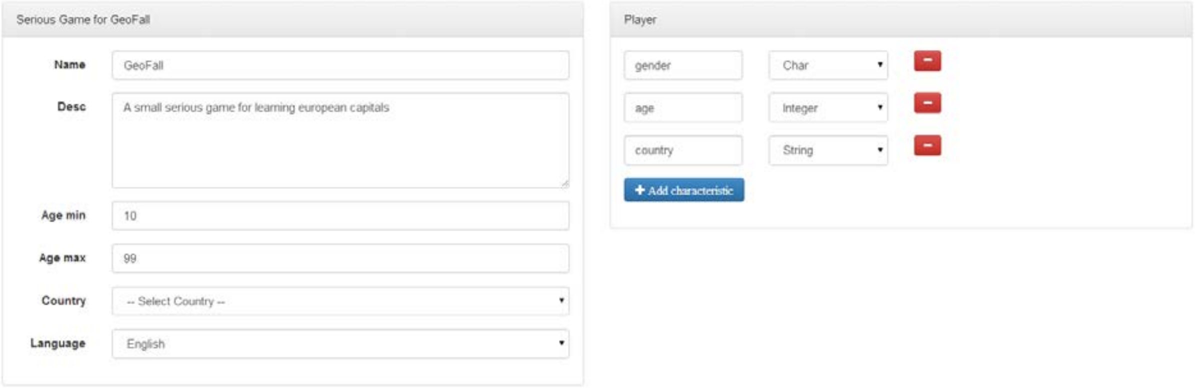
When starting the editor, a teacher is asked to select a game from the list of games he/she has access to. Doing so will send a request to the database and will retrieve the configuration file associated with the game, converted to a JavaScript Object Notation (JSON) format for ease of use. The JSON is then converted to a visual language by a Javascript program using the angularJS framework (Green and Seshadri, 2013).

There are five key blocks in the configuration file and they have corresponding representations in the editor. We designed the visual editor based on the literature review detailed in Section 3. It uses traditional HTML forms whenever the data to be collected is simple enough as teachers are used to these forms and there is no usability issue. However, as the core of the assessment logic is more complex, it will be represented with a visual diagram. We discarded the text-based option for the authoring tool as our target users, teachers, are not expected to be familiar with programming languages or even the English language.

5.1.1 Form-based user interface

Both the *Serious-game* and the *Player* sections of the configuration file can be represented easily without a visual diagram as demonstrated in Figure 3.

- *Serious game:* Here, the teacher can modify all the data describing the SG without restriction. An undefined field in the configuration file can be added here.
- *Player:* For learning analytics purposes, a teacher might want access to individual data about the players: their age, level, mother tongue etc. These can all be defined here. Each characteristic is defined with a name and a type (types can be integer, string, float, boolean, character or enum). The new version of the SG will ask the student to complete the information the system does not already have.



Serious Game for GeoFall

Name: GeoFall

Desc: A small serious game for learning european capitals

Age min: 10

Age max: 99

Country: -- Select Country --

Language: English

Player

gender: Char

age: Integer

country: String

Figure 3: Serious-game and Player edition

5.1.2 Visual representation of the assessment

The assessment logic of the configuration file is based on three main components: learning outcomes, feedback and actions. These components are closely related as shown in Figure 4. It was relatively obvious that a visual representation would be the best option for this section of the editor. These components are represented as geometrical shapes and arrows show feedback triggers and score modification. When selected an element opens a properties area on the right side of the editor for modification. The visual editor is illustrated in Figure 5 along with the properties area of an action.

Feedback: A feedback is any reaction to a player's action or inactivity. Feedback can be created, modified and deleted from a game assessment. Having feedback is not compulsory. A feedback is created with a name (unique), a message and a type (positive, negative, hint or badge). When a feedback is selected the editor displays the list of components (learning outcomes or actions) that triggers it.

Learning outcomes: The definition of learning outcomes is crucial in a teaching and learning process. A teacher might decide to split a general learning outcome into more specific ones in order to identify more precisely where his/her students are struggling. The visual editor permits the creation, modification, and suppression of learning outcomes. The only requirement is for the SG to have at least one learning outcome. A learning outcome is created with a name (unique), a description, and a starting value. The teacher can also specify if a feedback is triggered when the score associated with the learning outcome goes over or below a defined limit.

Actions: Actions cannot be created as they are too closely linked to the game mechanics; existing ones are associated to specific pieces of code in the SG and there would not be a method implemented to handle a new one. However, teachers can update existing functions. Actions have a name (unique and that cannot be modified), a set of parameters (that cannot be modified), a set of values for these parameters. Teachers can also specify a learning outcome and its mark as well as feedback to be triggered if the action is performed (i.e. if the values selected by the player were part of the values set).

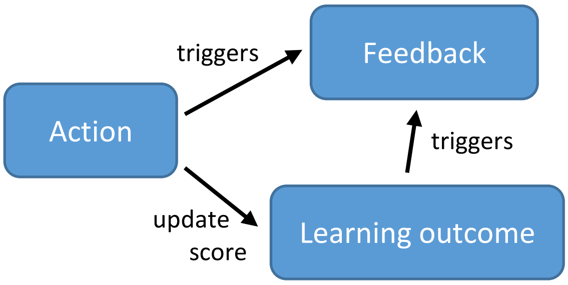
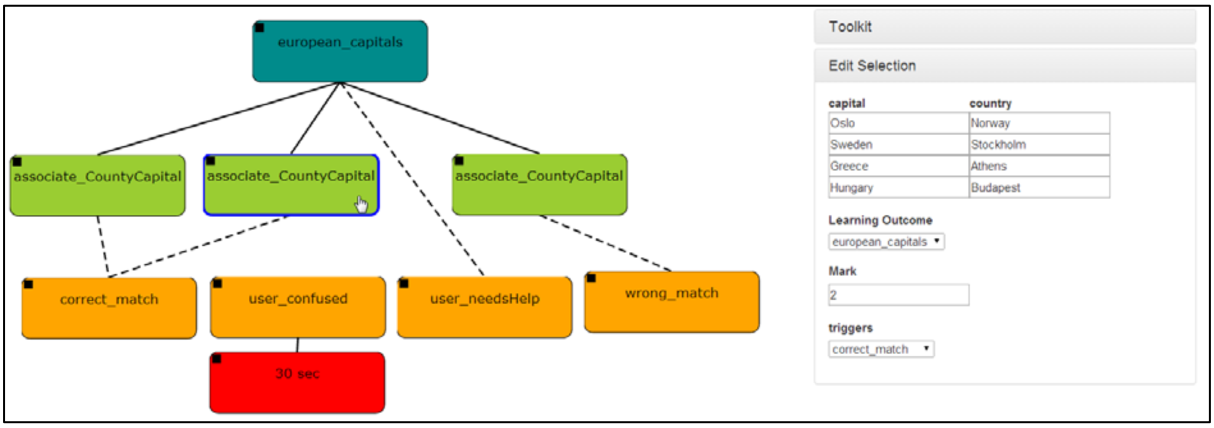


Figure 4: Relation between assessment components of the configuration file



capital	country
Oslo	Norway
Sweden	Stockholm
Greece	Athens
Hungary	Budapest

Learning Outcome: european_capitals ▾

Mark: 2

triggers: correct_match ▾

Figure 5: Visual editor of and properties area of an action.

5.2 The versioning system

Once a teacher saves the modifications to a game, the database will be populated with a new version of the initial SG, new rows will be created in the tables *seriousgame*, *learningoutcome*, *feedback* and *feeback_trigger*. This new version is a fully functioning game and has the id <idSG>.<num-version> where idSG is the id of the parent game and num-version is the number of the new version created (that number is automatically incremented for a single game and does not depend on the teacher). When saving a new version, the teacher is prompted to specify whether this version will be the version to be played by his/her students and if the answer yes is selected, the table SG-student will be updated. When a student logs on to a game the system will automatically know, from that table, which version to play. The schema of the database is described in Figure 6.

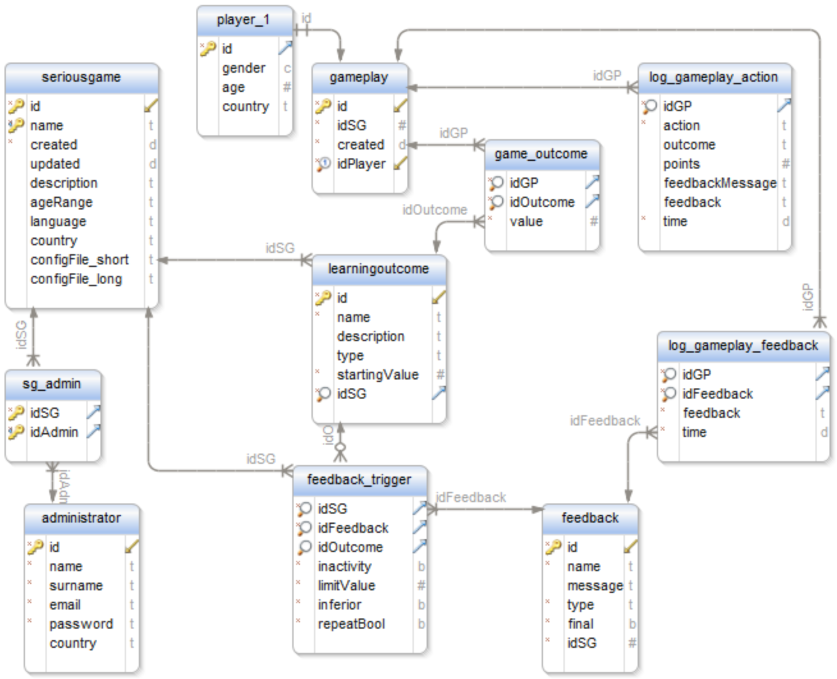


Figure 6: Database diagram of the project.

6. Conclusion

This paper has stated issues regarding educators' use of educational games as an assessment tool. It has presented a solution overcoming these issues: EngAGe, an Engine for Assessment in Games. EngAGe has three main components an assessment engine used by SG developers to evaluate the player's performance, a learning analytics dashboard generating reports on the use and outcome of a teacher's SG and a visual editor enabling teachers to modify the SG's assessment. This paper focuses on the latter and discusses how the teachers can use it to update their educational games at any time after the end of their development. A versioning system has been implemented to allow SGs to be modified by different teachers without generating conflicts. Every educator using the editor will create his/her own game that will be playable by his/her students.

An interesting future development planned for this project is adding a sharing system for teachers to have access to other teachers' versions of the games and to share their own with the community. Future work will also include gathering empirical evidence on the usability and usefulness of the visual editor.

References

Abt, C. C. (1987). *Serious games*. University Press of America.

Allen, M. J. (2004). *Assessing academic programs*. Boston: Anker Publishing.

Annetta, L. A., Minogue, J., Holmes, S. Y., and Cheng, M. T. (2009). Investigating the impact of video games on high school students' engagement and learning about genetics. *Computers & Education*, *53*(1), 74-85.

Breuer, J. S., and Bente, G. (2010). Why so serious? On the relation of serious games and learning. *Eludamos. Journal for Computer Game Culture*, *4*(1), 7-24.

Brown, M. 2011. *Learning Analytics: The Coming Third Wave*. [report] ELI Briefs, EDUCAUSE Learning Initiative (ELI).

Chaudy, Y., Connolly, T., and Hainey, T. Specification and Design of a Generalized Assessment Engine for GBL Applications. 7th European Conference on Games Based Learning (ECGBL), 3-4 October 2013, Porto, Portugal.

Conlan, O., Hampson, C., Peirce, N., and Kickmeier-Rust, M. (2009). Realtime knowledge space skill assessment for personalized digital educational games. In *Advanced Learning Technologies, 2009. ICALT 2009. Ninth IEEE International Conference on* (pp. 538-542). IEEE.

Gaeta, M., Loia, V., Mangione, G. R., Orciuoli, F., Ritrovato, P., & Salerno, S. (2014). A methodology and an authoring tool for creating Complex Learning Objects to support interactive storytelling. *Computers in Human Behavior*, *31*, 620-637.

Girard, C., Ecalle, J., and Magnan, A. (2012). Serious games as new educational tools: how effective are they? A meta-analysis of recent studies. *Journal of Computer Assisted Learning*.

Göbel, S., Mehm, F., Radke, S., and Steinmetz, R. (2009). 80days: Adaptive digital storytelling for digital educational games. In *Proceedings of the 2nd International Workshop on Story-Telling and Educational Games (STEG'09)* (Vol. 498, No. 498).

Hattie, J., & Gan, M. (2011). Instruction based on feedback. *Handbook of research on learning and instruction*, 249-271.

Hattie, J. and Timperley, H. (2007). The power of feedback. *Review of Educational Research*, (77), p.81-112.

Hunt, D. E. (1976). Teachers' adaptation: 'Reading' and 'flexing' to students.*Journal of Teacher Education*, *27*(3), 268-275.

Jarvis, S., and de Freitas, S. (2009). Evaluation of an immersive learning programme to support triage training. In *Games and Virtual Worlds for Serious Applications, 2009. VS-GAMES'09. Conference in* (pp. 117-122). IEEE.

Johnson, L., Adams, S., Cummins, M., Estrada, V., Freeman, A., and Ludgate, H. (2013). The NMC Horizon Report: 2013 Higher Education Edition.

Kickmeier-Rust, M.D, Mattheiss, E., Steiner, C.M. and Albert D. (2011). A Psycho-Pedagogical Framework for Multi-Adaptive educational games. In *International Journal on Game-based Learning, 1, 45-58*

Marchiori, E. J., Torrente, J., del Blanco, Á., Moreno-Ger, P., Sancho, P., & Fernández-Manjón, B. (2012). A narrative metaphor to facilitate educational game authoring. *Computers & Education*, *58*(1), 590-599.

McClure, J. R., Sonak, B., & Suen, H. K. (1999). Concept map assessment of classroom learning: Reliability, validity, and logistical practicality. *Journal of research in science teaching*, *36*(4), 475-492.

Mehm, F., Konert, J., Göbel, S., & Steinmetz, R. (2012). An authoring tool for adaptive digital educational games. In *21st Century Learning for 21st Century Skills* (pp. 236-249). Springer Berlin Heidelberg.

Porayska-Pomsta, K., Anderson, K., Bernardini, S., Guldberg, K., Smith, T., Kossivaki, L., ... & Lowe, I. (2013). Building an Intelligent, Authorable Serious Game for Autistic Children and Their Carers. In *Advances in Computer Entertainment* (pp. 456-475). Springer International Publishing.

Rodríguez-Cerezo, D., Gómez-Albarrán, M., & Sierra, J. L. (2013, November). A process model for the generative production of interactive simulations in engineering education. In *Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality* (pp. 95-103). ACM.

Rosas, R., Nussbaum, M., Cumsille, P., Marianov, V., Correa, M., Flores, P., ... and Salinas, M. (2003). Beyond Nintendo: design and assessment of educational video games for first and second grade students. *Computers & Education*, *40*(1), 71-94.

Shute, V. J. (2011). Stealth assessment in computer-based games to support learning. *Computer games and instruction*, *55*(2), 503-524.

Siemens, G., and Gasevic, D. (2012). Guest Editorial-Learning and Knowledge Analytics. *Educational Technology & Society*, 15(3), 1-2.

Sorensen, B., & Ramachandran, S. (2007). Simulation-based automated intelligent tutoring. In *Human Interface and the Management of Information. Interacting in Information Environments* (pp. 466-474). Springer Berlin Heidelberg.

Squire, K., Barnett, M., Grant, J. M., and Higginbotham, T. (2004). Electromagnetism supercharged!: Learning physics with digital simulation games. In *Proceedings of the 6th international conference on Learning sciences* (pp. 513-520). International Society of the Learning Sciences.

Susi, T., Johannesson, M., and Backlund, P. (2007). Serious games: An overview.

Tornero, R., Torrente, J., Moreno-Ger, P., & Manjón, B. F. (2010). e-Training DS: An Authoring Tool for Integrating Portable Computer Science Games in e-Learning. In *Advances in Web-Based Learning–ICWL 2010* (pp. 259-268). Springer Berlin Heidelberg.

Torrente, J., Del Blanco, Á., Marchiori, E. J., Moreno-Ger, P., & Fernández-Manjón, B. (2010, April). < e-Adventure>: Introducing educational games in the learning process. In *Education Engineering (EDUCON), 2010 IEEE* (pp. 1121-1126). IEEE.

Zyda, M. (2005). From visual simulation to virtual reality to games. *Computer*,*38*(9), 25-32.