Rapid Easy Authoring Platform for Serious Games  A Game-based Learning System for Work Safety Training

Peter Leong
Singapore Polytechnic, Singapore
peterleong@sp.edu.sg

Teo Kim Sheng
Singapore Polytechnic, Singapore
teo_kim_sheng@sp.edu.sg

Abstract:

In recent times, games-based learning has been attracting the attention of educators around the world. One of the challenges in deploying games-based learning to cover a larger proportion of the curriculum is the high costs associated with development of the games. Games development often requires specialized skills sets. Inputs from academic specialists for the learning content and pedagogy is also critical to the successful development of games-based learning. Our REAPSG platform empowers academics and trainers to create the games-based learning content on their own without the need for specialized development skills.

Keywords-component: game authoring tool; reusable game components; game templates; easy authoring; game knowledge mining; game evaluation

I. Introduction

The rapid proliferation of computer games on the internet and on smart mobile devices have enabled computer games to become a new media for mass communications. Many students in our institutions of learning spend a significant number of waking hours playing computer games. Many educators have experimented with games-based learning in class [1][2][3] to increase student engagement and motivation in learning [4][5].However, the high-costs (commonly reported in the millions of dollars) of creating customized game content for a particular lesson and curriculum have deterred wider adoption of game-based learning [6][7].

The high-costs of creating customized games stem from the fact that you require specialized 3D artists, modelers and animators in the game creations process. Specialist 3D graphics, physics, artificial intelligence and network programmers are also involved in the game creation process. Lastly, games created for a particular curriculum or geography often have a limited market size, and hence there is an insufficient number of users to spread the costs of game development.

For specific content domains such as work safety and health, the game developers need inputs for the domain experts in order to maintain the authenticity of the game setting and scenarios. Such domain experts may have little understanding about games technology. Hence, the frequent communications between the domain experts and game developers is necessary and this also drives up the costs of development; that is, the costs of engaging the domain experts as consultants in the project has to be included in the total project costs.

Our proposed approach to address this problem of high costs of game production is to build an easy-to-use set of tools such that academics and domain experts can create their own game-based learning content [12]
using our tools and deploy them to their students through the web browsers. The cost of creating the generic platform which we have named REAPSG (Rapid Easy Authoring Platform for Serious Games), can be factored over large number of potential users who will all use the platform to create content. The REAPSG platform will also provide some pre-configured game templates to assist the academics and lecturers in preparing the game-based learning content for their lessons.

II. Related Works

A number of projects have been initiated to use games-based learning in the classroom. The project [8] in Malaysia reported successful deployment of games-based learning among secondary one students. In Singapore, games-based learning has been used in schools [9][10][11]. In Victoria University, Australia, they have been using games-based learning for vocational training in the construction industry [12]. One source of the game based learning content for this projects have been commercial-off-the-shelf games repurposed for an educational outcome. Other projects have a customized game created specifically for their educational purpose, and end-users and trainers are not able to add to the content or modify it significantly. Some games-based learning projects make use of virtual world environments such as Second Life [13]. However, construction of new content logic within Second Life require specialized skills which are not easily picked-up by general academics and lecturers. Our approach in REAPSG is to make it more limited in functionality, but to make it very easy for any general academic or lecturer to be able to use the platform with minimal training.

III. Overview of the Rapid Easy Authoring Platform for Serious Games

The REAPSG architecture is based on the concept of empowering the teacher to create their own game-based content [14][15]. Teaching and engaging educators about serious games is challenging, but it is a model that can scale-up to many more courses and students with the engagement of educators in the process. Our REAPSG architecture must allow educator to create their own customized games [14][15][16], should mitigate the difficulties educators face in using game engine tools by themselves [17], provide some jump start content (prefabricated content templates to speed up the process), provide libraries of easily re-usable assets, and it should not require computers with high 3D graphics processing capability.

A. Architecture

The system is divided into two components: 1) the game authoring tool or game editor, and 2) the game player component. The game states during play are recorded in a database on the game server. In order for student or learner to play the game, they must use a web browser to download the browser plug-in or web player. This is done automatically when the user opens a particular URL that points to the lesson content.

The game authoring tool or game editor (see figure 1) is the main tool for academics and subject matter experts to create the game content. It runs as the desktop client on the lecturer’s workstation or laptop. After the lecturer has finished editing the game-based lesson, the lecturer would save the game state and it is recorded in a database on the game server. The REAPSG provides a number of basic game templates such that the lecturer does not have to create the entire game world from the ground up if he or she wants to just modify the content to suit his own purpose. However, if the lecturer wants to create everything in the scene from a blank slate, he would also be able to do so.
The game player component for REAPSG runs in a web browser (see figure 2). In order for student or learner to play the game, they must use a web browser to download the game at a particular URL that points to the lesson content. The URL link is specially encoded with information that points only to a particular lesson. Authentication of the user can occur via normal web-based authentication schemes.

If a web-based lesson management system (LMS) such as Blackboard is available, the URL based game loading mechanism can easily be integrated with the LMS content. It is the lecturer/trainer who must use the game editor tool to create the game-based learning content, and when he saves the edited work on the server, it will also generate an encoded URL which is linked to the game server content. The lecturer must copy the URL string (as generated) into the LMS (e.g. Blackboard) page. The URL mechanism is portable across all web-based LMS as it is an inherent part of the WWW cyber-infrastructure.

One of the concerns for 3D content is the size of the file to be down-loaded via the communications link. We have optimized the system to reduce the amount of data to be retrieved to a minimum. We actually download the 3D model, geometry and textures instead of video format of the game scene, and the rendering is done using the graphics processing capability of the user’s own laptop or workstation; this reduces the size of the data significantly. The corollary concern is the graphics processing power of the laptop or workstation and whether it will be able to render the 3D scenes with sufficient quality at an acceptable speed. Most modern laptops and workstations have basic 3D rendering capability, so if the game 3D models are kept to low-polygon count, the rendering time would be reasonable. We may not be able to support some of the 3D effects such as anti-aliasing, fog and atmosphere on machines without better 3D graphics cards. The rendering engine being used for REAPSG must be able to right-size the quality of the 3D content according to capabilities of the hardware; for example, HD quality graphics on high-end hardware but lower resolutions on lower-end hardware.

We have also partitioned the content into separate asset bundles such that you only down-load the 3D graphics models you are using in the lesson instead of all available models in the entire REAPSG library. Aside from minimizing download times for the web player, such an approach will also make the system easy to expand in the future, as new assets can simply be added to the server without requiring large amounts of code changes.
B. Implementation

To enable rendering of the graphics on multiple platforms ranging from desktop computers, to web browsers and mobile computing devices, we decided to use a game middleware. We selected the Unity3D game middleware because of its wide support of different devices playback platforms. Unity3D is able to support the Flash platform, iOS, Android, Windows desktop, Mac OS desktop, Wii U, PS3 and Xbox360. The Unity3D provides the graphics rendering capability, visual effects such as fog and particle effects, and also the 3D physics for game objects such as friction, elasticity, acceleration, velocity and angular momentum. The game physics engine does not support a number of physics effects such as fluid flow, thermodynamics, convection, conduction and electricity. Such physics effects, if needed, must be implemented in code by the programmer in the REAPSG platform.

The REAPSG supports two basic types of game formats. The first is a “Spot-the-X” format where the player has to use visual cues to distinguish items which have something amiss from the normal items in the scene. The challenge in this game format is largely based on time and ability to pick up visual cues using content knowledge about the setting of the game scene. For our REAPSG, our initial target application is construction work safety. For the “Spot the-X” format games, the safety trainees must use their knowledge about safety to indicate which conditions are hazardous in the scene. For example, a scaffolding without proper means of securing it would be hazardous. The safety trainees must be able to complete a level and spot a variable number of hazardous conditions within a time frame. The game will be played from a first-person perspective, to provide greater immersion and realism for the player as he/she navigates in the virtual worksite looking for potential hazards.

The second format of games that REAPSG supports is the “Branching-Story” format. This game format represents a scenario as a tree of possible outcomes given different actions. Based on the trainee sequence of input decision, he will traverse along a different branch or path through the tree. Each different path or branch will lead to a different outcome; some successful, some partially successful, while others end in failure. The “Branching-Story” allows the trainer to capture repercussions of a series of less than optimal decisions. It is more flexible but less visually oriented compared with the “Spot-the-X” format. The REAPSG uses still images as the backdrop for the scenarios in order to create some amount of interest and to create a
setting for the story elements in the “Branching-Story” game. For example, instead of purely using text to
describe a construction machine, we will be showing the machine in an image form with text descriptions of the
scene.

Figure 3. Branching stories view in the REAPSG player

The lecturer or trainer uses the REAPSG game authoring tools to create both “Spot-the-X” and
“Branching-Story” games. The completed game scenario is stored on the server. When a trainee launches a
game on his browser, the information about the game scenario is downloaded to the web browser and displayed.

Figure 4. Question & answer screen in the REAPSG player screen

Our current implementation is based on broadband or LAN speeds between the game server and
browser (it will not work over modems or GSM/GPRS connections). The player component on the web-browser
can execute both “Spot the-X” and “Branching-Story” games.

IV. User Evaluation of the Rapid Easy Authoring Platform for Serious Games

Getting feedback from the lecturers and trainers who have to use the REAPSG to create games-based
learning content and getting them to actually use them in their lessons is a critical success [18] factor in the
whole project. Our subject matter experts in Work Safety and Health (WSH) are not very familiar with games-
based learning. However, we have created a platform for them to share their views with the development team
during the course of the project.
To improve the brainstorming process, all the game designers and developers attended a design thinking workshop prior to the actual commencement of the project. Using design thinking, we hope to achieve deep user empathy with the needs of the WSH trainers. We also used user observations as the WSH trainers used the initial prototypes of our system to gauge their reaction and to understand their viewpoint as they use such as games-based learning system for the first time.

To aid our decision and design process, we have also created a score card [19][20] or rubric on the features of the system when it is matched against the WSH trainer voiced and unvoiced requirements (see table 1). Each trainer is allowed to have only a fixed number of vote which he can put into any cell, or put all into the same cell. The row score indicates the usefulness of including that feature in REAPSG. The column total indicates the overall number of matches in each of the mandatory, good and optional categories. The total row score is an overall score for the system, a higher value indicates we have are closer to deriving maximum value for the REAPSG system for the trainers who will be using it.

<table>
<thead>
<tr>
<th>S/N</th>
<th>REAPSG Game Lesson Objects (SpotX/BS) and Features</th>
<th>Mandatory/Regulatory (x 10)</th>
<th>Good (x 5)</th>
<th>Optional (x 1)</th>
<th>Row Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fall from high place (SpotX)</td>
<td>2+3+3</td>
<td>1</td>
<td></td>
<td>85</td>
</tr>
<tr>
<td>2</td>
<td>Fall from high place (BS)</td>
<td>2</td>
<td></td>
<td></td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Column Totals</td>
<td>10</td>
<td>1</td>
<td>0</td>
<td>105</td>
</tr>
</tbody>
</table>

V. Conclusions

The REAPSG is a system that allows subject matter experts and lecturers to create their own customized games-based learning content easily, quickly and affordably. In the future, we are planning to expand the number of game templates created and also pilot the system with a number of students in the Diploma in Work Safety and Health (WSH). We believe REAPSG will be a step forward towards more authentic learning in WSH and will lead to greater knowledge and retention by the trainees. We also believe the interactive content will increase the level of engagement and motivation of the students.

ACKNOWLEDGMENT

This project is made possible with funding support from the Institute of Adult Learning CET Innovation Fund. Together with subject matter expertise from The Professional & Adult Continuing Education (PACE) Academy, Singapore Polytechnic.
REFERENCES


