Lessons Learned: Outcomes from the Development and Delivery of an Online Learning Platform Using Silverlight 5/XNA and Windows Azure

Neil Gannon, Tony McCabe, Paul Powell
Institute of Technology Sligo (Ireland)
Gannon.Neil2@itsligo.ie, McCabe.Tony@itsligo.ie. Powell.Paul@itsligo.ie

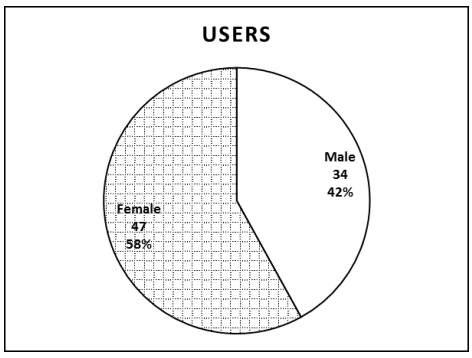
Abstract

Online learning has grown rapidly in recent years with the adoption of learning platforms [1, 2, 3, 4, 5] such as Moodle, Coursera, Udacity and edX all of which offer passive learning options such as closed captioned videos of lectures, multiple choice questions, essays and other online submissions [6] for assessments. The technologies behind these offerings are well known and many platforms exist for those who wish to follow this route. However groups looking to deliver interactive 2D or 3D game based learning experiences with automatic grading over the internet have limited choices in what technologies can be used [7, 8, 9, 10] and the impact of the selected technology varies for both learner and educator.

This study outlines the methodology and outcomes of delivering interactive browser based educational applications developed using Silverlight\XNA and Windows Azure [11] throughout a three month period in 2012 to a group of over 80 third level students and lecturers in the field of medical biotechnology. The students accumulated over 253 hours in total active playtime over the course of 9 individual lessons (~3 hours per user).

1. Introduction

Since 2010 the Institute of Technology Sligo has been delivering virtual medical biotechnology laboratory practical sessions to students. Each virtual practical session called a mission involves students completing assigned tasks. Multiple missions are grouped together into individual games for specific student groups. Metrics are gathered on the student's interaction within the missions and progress throughout the game. Missions include a combination of instant feedback and uploaded assessment. Students previously had the option to install the application on their own personal computer or attend weekly game playing classes where students could play the game in a supervised computer laboratory. In an effort to make the games more accessible for both students and lecturers the existing delivery method was migrated to a modern web based technology. This web based approach incorporates browser based 3D simulations together with a range of different application features including media streaming, game editors and a range of supporting web services while also allowing for seamless updates of missions.



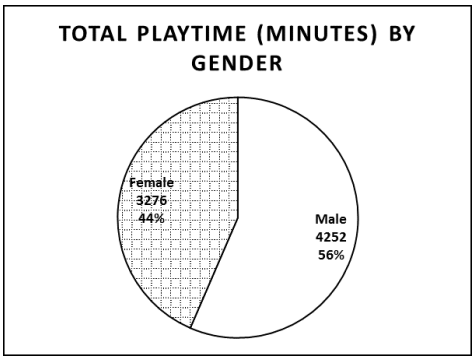


Figure 2 Total Play Time by Gender

A total of 81 students [Figure 1] participated in the online delivery. This group was divided into two further subgroups. Each group played individual games consisting of multiple missions in which players must complete specific objectives.

HETAC Level 7 Degree [12]
-Focused on knowledge acquisition and practical application.
-Required users to learn standard operating procedures for a medical biotechnology laboratory.
-Five individual game missions and two assessments were delivered [Figure 3]. Each mission had unique requirements in both learning outcomes for students and software functionality.

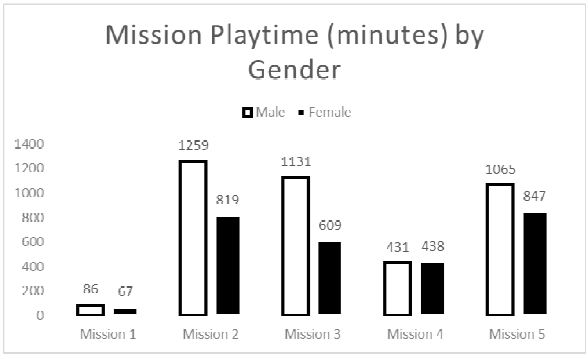


Figure 3 Mission Play Times for Level 7 Group

HETAC Level 8 Degree [12]
Focused on critical thinking and applying knowledge to new problem domains.
Required users to plan experiments by picking the required reagents and equipment. All items included in game are based on the specifications of their real world counterparts.
Four individual game missions were delivered. All Level 8 missions were identical in software requirements. Only total budget amounts and experiments varied between missions.
The survey data presented in the results section of this paper is based on the Level 7 group of users.

2. Methodology

Multiple applications needed to be delivered to users in a web browser. To reduce the burden on users in terms of granting access to the 3D graphics capabilities of the Silverlight plugin [13] and to simplify the incorporation of multiple missions in the same client a unique web browser delivery method was developed.

The system mimics how current mobile platforms application catalogues deliver applications to client's devices. The following section details the key parts of the architecture and important features.

2.1 System Architecture

The system can be broken into three core pieces. [Figure 4]

-Windows Azure Hosted Web Services (Windows Communication Foundation)
-Windows Azure Storage (Blob and SQL Azure)
-Thin Web Client (Silverlight)

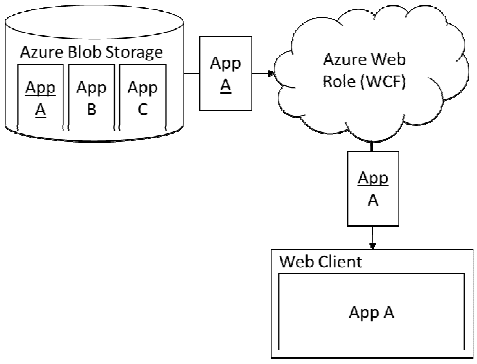


Figure 4 Architecture Overview

2.2 Web Services

Multiple web services were developed to serve the range of applications accessed by users. The web services are responsible for the providing a method to download a range of applications from Azure blob storage.

2.3 Thin Web Client

The previous games had been developed in C# and the Microsoft XNA Framework. The development window was restricted to four months, meaning the chosen web technology needed to facilitate a smooth transition from the existing desktop applications to web based versions.

At the time of development (Jun 12 – Sep 12) the Microsoft Silverlight 5 plugin met all the requirements for the project. Silverlight 5 is a.NET based browser plugin that supports a wide graphics feature set based on the XNA Framework and DirectX 9. Along with new gaming features Silverlight faciltates rich internet applications and has a seamless integration with Windows Communication Foundation web services. This made Silverlight a natural choice.

2.4 Applications

In total, six applications were developed and delivered to the Level 7 user group online. All applications followed standard Silverlight 5 development practices. Development and testing is identical to that of a standard Silverlight application. The only requirement for an application is the implementation of a supplied interface that would identify the entry point for the application. Developed applications are stored in Azure blob storage. The applications are delivered through a web service and used by the web client to load and create an instance of an application, thus hosting the downloaded application inside the client.

3 Results

Initially the delivery of the applications to users was intended to be monitored with adequate computing laboratories reserved and used solely for the purpose of playing the games. Due to unforeseen infrastructure restrictions imposed on users via corporate network policies on networked machines, the majority of users were unable to use the 3D simulations and reverted to using 2D versions on their own personal computing devices.

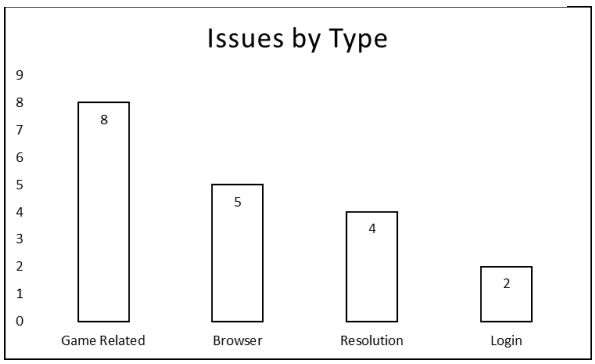
Over the course of the delivery numerous issues became apparent from the users perspective. The majority of users were unable to play the 3D versions of the simulations and needed to revert to the 2D simulations. Most issues were not reproducible and a lack of technical knowledge on the part of the users resulted in misinformation being relayed about applications issues.

A survey of Level 7 users was performed to identify the cause of their issues. The survey was designed for non-computing individuals and focused on general hardware and software features. Users were asked to list the issues they encountered and provide details on issues. On analysis it was found that the same issues occurred for most users. These issues have been divided into four main categories.[Figure 5]

-42% of users reported game related issues. These issues ranged from simple game fixes that were resolved within a 24 hour period to performance issues that were unique to each individual user due to varying hardware. [Figure 5]

-26% of users reported issues when using specific web browsers, these issues related to unresponsive browsers or failure to load the Silverlight client. Interestingly Google Chrome was the

Figure 5 General Issue Categories



source for most of these issues on both Windows and Mac OSX operating systems, neither Internet Explorer nor Safari were mentioned as the source of issues. [Figure 5]

-21% of users noted screen resolution issues. Some missions had visual elements of a fixed size (none of these elements were more than 600 pixels in width or height) and users were unable to see all the items they needed to see. The applications all targeted 1024x768 resolution screens. As of January 2013 90% of screen resolutions are over this. [Figure 5]
-11% of users were unable to login to the service. All users were assigned a default starting password which they changed on the first login. It was further found that 100% of these students used social media services which require the same login process. [Figure 5]

As 89% of the issues reported were related to performance or the inability to successfully launch or run the applications, it was important to profile user's hardware and operating systems.
Surprisingly 35% of users were still using Windows XP, an 11 year old operating system. Such a large percentage of XP users means that their hardware was purchased over 5 - 6 years ago and technically not suitable for running the more advanced features of the application. This combination of outdated software and hardware resulted in these users not having the optimal experience.

4. Conclusions

Despite the wide range of issues that occurred during the delivery, every student has played every mission that was made available. In light of what we learned from this exercise, we offer the following conclusions:
-Closer collaborations between developers and IT support is desirable. A lot of issues can be avoided if there are dedicated IT support structures in place to develop and deliver such experimental applications. The lack of infra-structure support added to a large number of unknowns being introduced, particularly with regard to user's hardware, compounds the technical challenges in

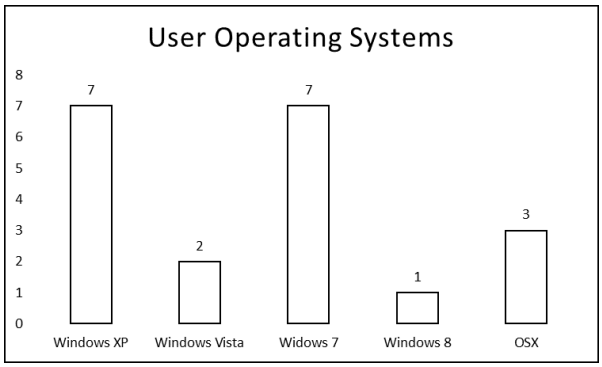


Figure 6 User Operating Systems

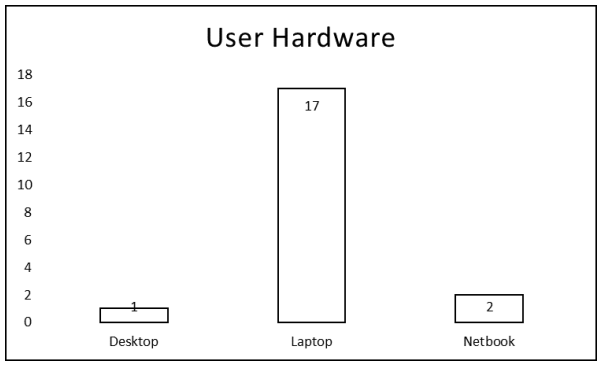


Figure 7 User Hardware Types

delivering rich interactive missions.
-The application delivery mechanism is hugely beneficial. It reduces the work load on users by centralising all applications under the one domain. Users are not aware that applications are being downloaded and deployed. Developers have freedom to update the applications with seamless transitions for users.
-Although Silverlight is not documented as supporting the loading of applications or even assemblies into an application. It is possible to do so using a technique that involves loading XAP resources which contains the application to be hosted.
-In using the software, the games industries minimum requirements for hardware is advised. Some user will be excluded and these users need to be pushed to update their hardware if they intend to use the advanced features of the missions. Alternatively these individuals can be catered for by less hardware intensive delivery with the same functionality. Both approaches use the same game based delivery engine.

6. Summary

Online game based learning has an important role to play in educational delivery of practical based learning. The ability to deliver rich 3D content requires a certain standard of hardware on the client side. The total playtime data gathered from game play [Figure 3] suggests that students embraced the technology as part of their practical learning. The delivery method outlined in this paper provides flexibility in updating the game mission content and functionality in a seamless way while allowing issues encountered by users to be dealt with in a timely manner with no disruption to the service.

References

[1] Moodle, Available at: https://moodle.org/ (Accessed, 19th December 2012)
[2] Coursera, Available at: https://www.coursera.org/ (Accessed, 19th December 2012)
[3] Udacity, Available at: http://www.udacity.com/ (Accessed, 19th December 2012)
[4] edX, Available at: https://www.edx.org/ (Accessed, 19th December 2012)
[5] Jenner, Matt (2012), *Being a Coursera student and just what is a MOOC anyway?* Available at: http://blogs.ucl.ac.uk/ele/2012/08/27/being-a-coursera-student-and-just-what-is-a-mooc-anyway/ (Accessed, 20th December 2012)
[6] University College London, *E-assessment,* Available at: https://www.ucl.ac.uk/isd/staff/e-learning/tools/assessment (Accessed, 19th December 2012)
[7] Unity Web Player, Available at: http://unity3d.com/webplayer/
[8] Adobe Flash, *Stage 3D,* Available at: http://www.adobe.com/devnet/flashplayer/stage3d.html (Accessed, 15th December)
[9] Google Native Client, Available at: http://code.google.com/p/nativeclient/ (Accessed, 15th December 2012)
[10] Microsoft Silverlight 5, *What's new in Silverlight 5,* Available at: http://msdn.microsoft.com/en-us/library/gg986857(v=vs.95).aspx (Accessed, 15th December 2012)
[11] Microsoft Windows Azure, Available at: http://www.windowsazure.com/en-us/ (Accessed, 16th December 2012)
[12] HETAC Awards Standard for Science, Available at: http://www.hetac.ie/docs/Science%20-%20Award%20Standards.pdf
[13] Some reasons why 3D is not working with Silverlight 5, Available at: http://blogs.msdn.com/b/eternalcoding/archive/2011/10/18/some-reasons-why-my-3d-is-not-working-with-silverlight-5.aspx
[14] W3 Schools Browser Display Statistics, Available at: http://www.w3schools.com/browsers/browsers_display.asp