School of Computing

UNIVERSITY OF LEEDS

COMP3860

Scoping and Planning Document

Student Name:Khen Cruzat
Programme of Study:Computer Science
Provisional Title of Project: Cloud Gaming and Simulation in Distributed Systems
Name of External Company (if any)**:**
Supervisor Name: Jie Xu, Peter Garraghan
Type of Project: Exploratory Software
***NOTE to student**: ensure you have discussed the content with the supervisor well in advance of the deadline for submission. Submit a **hard copy** to SSO. An **electronic version** of this report in pdf must also be submitted via the appropriate module folder in the VLE; with filename of the format <surname><year>-SP (e.g. SMITH15-SP.pdf).*

Signature of Student:	**Date: 5/2/2016**

Assessor *(leave blank)***:**
***NOTE to assessor:** feedback form is available to download from VLE resources 'Quick links'. On completion, please attached the feedback form to this document and return to SSO.*

Contents

1. Background Research for the project

1.1 Context

Cloud computing has always been seen as a way of improving compute performance by the use of multiple computers connected to each other. The games industry has rapidly evolved along the years and a great deal of demand is apparent. Developers of games have pushed computer hardware to meet the needs of consumers for more complicated games and realism. Even with computer hardware becoming cheaper and more of a commodity, the costs of driving graphics rendering for high-end games to run at the optimal settings of 1080p at 60fps are still relatively high.

Cloud gaming is new technology that can be seen as an alternative by having the games run remotely on a server and then streamed to the user. Performing computations remotely as with streaming games remotely is believed to gain traction in the future in the same way how streaming videos and audio have become ubiquitous through services such as Netflix and Spotify. NVIDIA's GRID Cloud Gaming advancements have shown that this is becoming the case. As stated by Mariano in *Is cloud gaming the future of the gaming industry* [1], cloud gaming is increasingly becoming an attractive option for consumers as higher end games can then work on simpler, cheaper clients as well as with devices that they may already own.

In the paper *Cloud Gaming: A Green Solution to Massive Multiplayer Online Games [3]* it mentions that NVIDIA has introduced SHIELD which is a mobile gaming device that can be connected a desktop PC with a compatible NVIDIA GPU and stream gameplay to the device via 802.11n WiFi. Another feature [6] is the ability to connect to one of NVIDIA's data centres to play games from their selection of stream-ready games. One of the benefits of this service is the convenience of not having to wait for the download and installation of the game as you simply pick a game and instantly start playing. The service also boasts gameplay performance of up to 1080p at 60fps.

1.2 Problem statement

Compute power of games can be offloaded to a server of much powerful computers and then streamed to a client with lower specification hardware such as a laptop or a mobile device. With this comes risks and shortcomings that need to be factored. One of these problems are the latency of the game. Latency can be huge factor in the gameplay such as high-paced games like first-person shooters or fighting games. The delay in pressing a button on the gamepad to seeing the action performed on the screen needs to be kept to a minimum. This idea of interaction delay tolerance being different from genre to genre of games is discussed by Shea et al. [2]. As stated above, a player of FPS games can only tolerate the least which is around 100ms whereas Role playing game (RPG) gamers can tolerate around 500 ms.

Another problem that is directly linked to delay in the system is the effect of packet loss. As stated in the *Eight Fallacies of Distributed Computing* [7], it should be assumed that latency is never zero as mentioned above as well as network is not always reliable. This means that packet loss can occur which in terms of cloud computing can mean the degradation of image quality. In the investigation conducted by Jarschel et al. [8] in which they surveyed average consumers about the importance of packet loss and delay. Generally the quality of the video streamed to the clients plays an important role as the participants were open to using such as a service if provided in good quality.

1.3 Possible solution

A possible solution to the problem should show the potential of cloud gaming as opposed to how conventional games are being played now. This can be done by producing a solution that demonstrates a game being played remotely on the cloud with minimal delay and issues.

Depending on the speed of progress, a minimal solution would have a working cloud system using the School of Computing Cloud Testbed to play a game remotely using another computer system connected to the same network. A full solution should optimise the architecture that greatly improves the interaction delay. If a GPU is accessible, then a solution which takes advantage of a GPU to offload some of the computation and produce a better performing system would be ideal.

In the paper *GamingAnywhere: an open cloud gaming system* [4] it shows an open source implementation of a cloud gaming system where games are provided via the cloud from a data center and streamed to thin client's system. A possible solution will take this as a foundation and improve upon it.

An improvement can be applying the *Outatime [5]* system which is said to be able to mask up to 120ms of network latency. Outatime uses a prediction algorithm namely the Markov prediction to speculate which frames are needed next in order to reduce the delay. The predicted frames are sent to the client one entire RTT ahead of time. To smooth out noise and misprediction, Kalman filtering is used which in turn reduces the video shakes. Due to this technique the client perceives little latency.

Shirmohammadi's paper on *Delay Reduction in Cloud Gaming* [9] discusses the areas in which delay can be reduced in a cloud gaming system. Under the assumption that there is no control over the transport network itself since its controlled by the ISP, research is mainly focused on the cloud and the client side. It is said that that the video encoding process contributes up to 52% of the processing time. The other method explored is to reduce the delay by optimizing the cloud infrastructure through the application of Software Defined Networks (SDNs). Compared to the conventional method of Open Shortest Path First, it reduced end-to-end delay by 9%.

A challenge that can arise is that the processing done on the server side can be too long and would render the game unplayable. Since I have not done work on encoding OpenGL frames to video or decoding them, this will be be a challenge to optimise the process.

With the Computer Graphics module, it will provide a strong foundation in creating the game engine and the simple 3D game which means I shouldn't need to spend too long in creating it. Also, since I have taken the Distributed Systems module it will help in tackling with networking issues I have to face in implementing the cloud gaming system. I will need to create a way for client computers to easily communicate with the Cloud to send and receive the game data.

1.4 How to demonstrate the quality of the solution

A way to measure the success of my solution in solving the problem is that the produced cloud gaming system to run better than the game running on the client's system. This means the game should be able to produce more frames per second on the cloud gaming system than on the client's system. Another measurable is to make the interaction delay small enough that the user's gameplay experience won't be affected. This delay will be calculated my measuring the round trip time of when the user presses a button to when the frame corresponding to the action is received from the server.

2. Scope for this project

2.1 Aim

The aim of the project is to generate an understanding of how cloud gaming can be improved on in order to become a service to the average consumer in the same way how video and music streaming became available to the masses. Through implementing a simple model of cloud gaming, it will act as a test platform to improve interaction delay by optimizing the architecture. A chosen technique will be implemented and compare the benefits of using this technique in reducing latency.

2.2 Objectives

1. A simple game program, that is computationally expensive enough to not perform optimally on a single machine (simple flight simulator with procedurally generated trees). This machine should be lower end and not the DEC-10 computers since the specifications of the hardware are very high end. On the other hand it would still be interesting to see if there would be any performance improvements that could be made in comparison to high end computers and the cloud system produced at the end.

2. A simplified cloud gaming system where the game created is launched on the cloud and input on the client side in the form of button presses on the keyboard is sent to the game on the server. The game frames produced are then sent to the client's screen.

3. An optimised cloud gaming system where it performs better than running the game on the thin client's system and has acceptable interaction delay. It will use the technique investigated and should show an improvement. Also make use of GPU connected to the server for increased performance.

2.3 Deliverables

A deliverable the will be produced at the end of the project of code that demonstrates a simple game/simulation rendering graphics on a server then streamed to a client computer. A manual on how to setup the client and server will be produced so the cloud gaming system can be easily setup and launched. Along with these deliverables will be the project report that explains the problem the project is trying to solve and the schematics of the solution produced as well as an evaluation of the solution. The evaluation will include comparisons using frames per second and latency times. It will show the difference in performance and interaction delay.

3. Project schedule

3.1 Methodology

The approach that will be undertaken for this project will be iterative. After doing background research, the next process is literature review of the information I have found. Comparisons of different techniques that will reduce latency and increase the overall performance of the system will be made. A single technique will then be chosen to focus on and improved on. The next step is to design the architecture for the cloud gaming system along with how the technique will be applied.

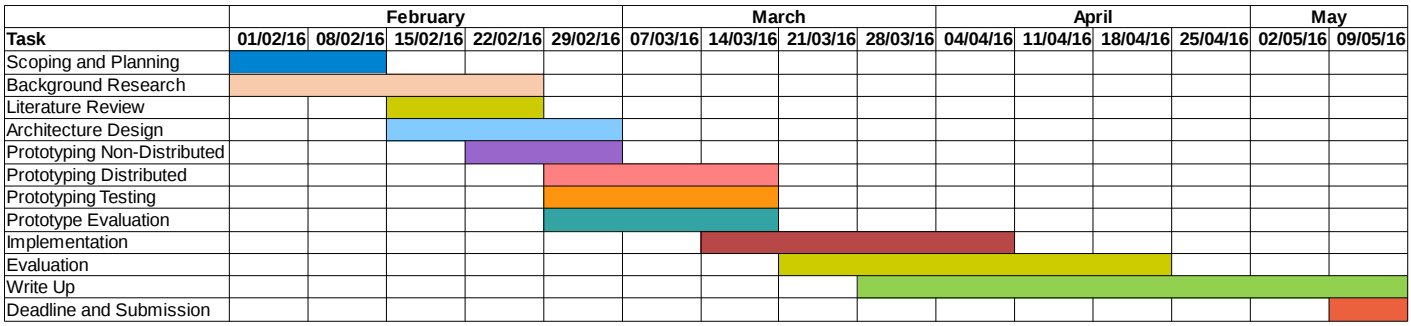
A prototype of the game will be produced and need to make sure it is running properly on a single system in the DEC-10 Lab. The game will render procedurally generated trees using Lindenmayer system which is computationally expensive to produce randomly generated trees. The game will also use lighting and shadows which adds more to the compute power needed. The area of trees can be navigated around by the player. I will evaluate the suitability of the game to make sure that it will provide a good measurement of performance.

After evaluation of the non-distributed approach, I will have the game run on the School of Computing Cloud test bed. This will be done by creating a virtual machine using the OpenNebula interface and running the game on it. I will record the difference in performance in terms of frames per second.

When this is completed, I will create a server-client system where the server can receive user input from a client and relay the commands to the OpenGL program. Then I will encode the OpenGL frames produced using x264 encoding so then the video frames can be streamed to the client. These video frames will be decoded on the client side using a video player. The performance of this measured in terms of frames per second produced as well as the round trip time for user to input a command and then to receive the video frames.

Using an iterative approach will make sure the project can be completed in manageable chunks. This also produces a deliverable at the end of each iteration.

3.2 Tasks, milestones and timeline



As stated in the objectives and methodology, this Gantt chart shows the schedule of the project.

3.3 Risk assessment (if appropriate)

A risk involved with this project is the availability of GPU with the School of Computing Cloud Testbed. If I wish to explore offloading rendering to a GPU to increase performance, I will not be able to do this. Currently there are two GPUs installed on the servers, but these are not yet made to be accessible as the hardware has not yet been integrated with OpenNebula. This means the rendering and processing on the cloud side will have to be all done on the CPU and the game that will be developed will have to be compute intensive and shouldn't use graphically intensive tasks such as textures.

Another risk that can occur during the project is the actual availability of the cloud that I will use. The servers have the possibility of going down for many reasons such as hardware failure and server maintenance. This means I will not be able to work on the cloud side of the project and can slow down the overall progress.

References

[1] Mariano, B. and Koo, S.G.M 2015. *Is Cloud Gaming the Future of the Gaming Industry.* Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on

[2] Shea, R., Jianchuan, L., Ngai, E.C.-H and Yong, C. 2013. *Cloud Gaming: Architecture and Performance.* Network, IEEE **27**(4)

[3] Chuah, S.P., Yuen, C. and Cheung, N-M. 2014. *Cloud Gaming: A Green Solution to Massive Multiplayer Online Games.* Wireless Communications, IEEE **21**(4)

[4] Huang, C-Y., Hsu, C-H., Chang, Y-C. And Chen K-T. 2013. *GamingAnywhere: an open cloud gaming system.* Proceedings of the 4th ACM Multimedia Systems Conference. ACM New York

[5] Lee, K., Chu, D., Cuervo, E., Kopf, J., Wolman, A., Degtyarev, Y., Grizan, S. and Flinn, H. 2013. *Outatime: Using Speculation to Enable Low-Latency Continuous Interaction for Mobile Cloud Gaming.* MobiSys 2015. ACM

[6] NVIDIA. 2016. *NVIDIA GeForce Now.* [Online]. [Accessed 4 February 2016]. Available from: https://shield.nvidia.com/game-streaming-with-geforce-now

[7] Wilson, G. 2015. *Eight Fallacies of Distributed Computing.* [Online]. Accessed 7 February 2016]. Available from: http://blog.fogcreek.com/eight-fallacies-of-distributed-computing-tech-talk/

[8] Jarschel, M., Schlosser, D., Scheuring, S., and Hobfeld, T. 2011. *An Evaluation of QoE in Cloud Gaming Based on Subjective Tests.* Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on

[9] Shirmohammadi, S. 2015. *Delay Reduction in Cloud Gaming.* IEEE COMSOC MMTC E-Letter

Appendix A. How ethical issues are addressed

This project will not make use of human subjects to test the cloud system as it will just be a proof of concept. This means that there will be no ethical issues that needs to be addressed.