Michael Richardson

December 27, 2013

CS 4670

Project Proposal

# Abstract

The project is to build a multi-platform space-themed game with a modular architecture and elements of gameplay for both entertainment and education. My portion of the project is to define requirements, design, build and test a server to run on a Linux machine and clients to run on various platforms including web and mobile to play the space game. I will be implementing an early prototype of the game to test the clients and server.

# Detailed Description

This is a long-term project spanning multiple semesters with a modular architecture so students can work on modules independently or as part of a research group. The game will have a space theme and will incorporate aspects of role-playing, strategy, and 4x games (explore, expand, exploit, and exterminate).

My portion of the project is to define requirements, design, build and test libraries for clients and a server with a database for players, characters, and world (setting) information. The server will be written in python and will need to support HTTP traffic as well as sockets and web sockets. The database will store data as JSON objects in MongoDB. The API for the server will adhere to principles of HATEOAS and REST. I will be creating a central server program with libraries for database, communication and game logic.

The communication libraries will be built to support socket connections, JSON traffic and elements of the API and protocol(s). The protocol will include definitions of the types of data and their properties. TCP socket connections for both HTTP and real-time traffic. Players will be able to use a variety of clients including: desktop applications, native mobile applications, desktop and mobile browsers and SMS. These clients may use 3D graphics, 2D graphics, ASCII Graphics or just text, depending on the type of client and available network bandwidth. Some parts of the game will not be accessible every client (SMS being particularly limited).

I will develop a desktop client for Windows, one or more mobile clients and a web (browser) client. The desktop client will be using Visual Studio 2013 to build either be a console application or a WPF application. The mobile client will be will be built for Windows Modern, Windows Phone or Android. The browser client will be built to be compatible with all major desktop browsers (Chrome, Safari, Firefox and Internet Explorer).

The early prototype of the space-themed game will allow players to move around a map, collect items, achieve some victory condition and communicate real-time with other players. The objective of the server and client is to demonstrate the functionality of the libraries and protocol. Git will provide version control for all the project source code and documentation. The Source code will be released as open source under the Apache License.

# Applicable Tools, Platforms and Standards

* Server
  + Linux
  + Sublime Text
  + Python
  + MongoDB
* Clients
  + Android\*
    - Eclipse **or** IntelliJ
    - Java
    - Phone
    - Tablet
  + Windows Metro/Modern
    - Visual Studio 2013
    - C# **or** JavaScript
    - XAML **or** HTML and CSS
  + Web
    - Chrome
    - Firefox
    - Safari
    - Internet Explorer
    - JavaScript
    - HTML
    - CSS
    - Intel XDK
  + Windows Application
    - Console App
    - WPF App
    - Visual Studio 2013
    - C#
    - XAML
* Other
  + NotePad++
  + REST
  + JSON
  + HATEOAS

# Licensing

The source code will be licensed under the Apache License, Version 2.0 as follows:

Copyright 2014 Michael Richardson

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

# Proposed Deliverables

1. Source Code (GIT Repository)
2. Communication Libraries and Documentation (Finished and Polished)
3. Multi-platform game (Prototype)
4. Mobile Client (Prototype)
5. Web Client (Prototype)
6. Desktop Client (Prototype)
7. Server (Prototype)
8. Final Project Report
9. Oral Presentation
10. Daily Journal/Work Log
11. Any other relevant items

# Project Schedule

January 6 – Classes Begin

January 13 – Gather and define requirements

January 20 – Design game and API

January 27 – Build Server communication library

February 3 – Build Server communication library

February 10 – Build Client communication libraries

February 17 – Build Client communication libraries

February 24 – Build Client communication libraries

March 3 – Build Client communication libraries

March 10 – Implement database

March 17 – Implement game prototype

March 24 – Implement game prototype

March 31 – Testing

April 7 – Prepare final report and presentation

April 14 – Prepare final report and presentation

April 21 – Prepare final report and presentation

April 25 – Classes end

# Project Details

**Advisor:** Brian Durney

**Completion Semester:** Spring 2014