CS351: Red Run Proposal

Jeffrey Nichol, Jayson Grace,
Troy Squillaci, Adam Mitchell, Jordan Medlock

November 7, 2014

Abstract and Implementation

The objective of this project is to create a simplified version of the game Death Run (a 3D game), which is a derivative of Gary's Mod. In this game, there will be two teams, the Doom Bringers and the Victims. The Victims role will be to navigate an obstacle course in which there are many traps that can be triggered by the Doom Bringers. The Doom Bringers role will be to kill off The Victims using the traps at their disposal, before they can reach the end of the course. It is important to note that once a trap has been triggered by the Doom Bringers, there will be a cooldown before the trap can be triggered again.

The game will required both a server and a client. The server will have a database on it which will store data pertaining to the map, such as trap locations, trap type, victim locations, and Doom Bringer locations. Additionally, the database will store information about the Victim's attributes such as health and the player model, as well as Doom Bringer attributes, which will be similar to the Victim's attributes with the exception of health. The client will be responsible for rendering the OpenGL component of the game. It will also provide users with the capability of input such as controlling the character, and in the case of the Doom Bringers, springing traps.

Lastly, there will be a proprietary network protocol created to facilitate the transmission of data between the client and server. It will be as close to real time as possible, so that there will be minimal latency between the client and server.

Roles and Responsibilities

There are five members on our development team. Each member and the respective responsibilities are as follows:

Jeffrey

- Graphics -
- Project Manager - Help keep track of roles for each team member on the project, serves as a communications medium for technical implementation concerns

Jayson

- Systems Engineer/Administrator - Setting up test machines, building and maintaining the server, troubleshooting
- Network Engineer - Configuring a Virtual Private Network (VPN) for testing and communicating with the server, troubleshooting
- Network Design - Creating a network protocol to facilitate communication between the client and server
- Database Engineer - Determining the design for the database and implementing an ORM for usage by the rest of the team
- Model Design - Utilizing the Object Relational Model (ORM) and the network protocol to facilitate a communications framework for the other application layers in our application model

Troy

- Graphics Lead
- Map Design

Adam

- Systems Administration - Assisting the team with the windows test systems for various functions
- Map Design

Jordan

- Graphics
- Map Design

UML and ERD

To keep track of the various aspects of our project, there will be a formal UML document hosted at genmymodel.com that serves as a contract between all developers on this team. This contract will help to determine how each individual member will tie their code into other member's existing infrastructure.

The ERD will act as the formal database design document. Through this document, the team will be able to understand the implementation details of the database.

Development Cycle

For each major feature of the program, there will be a feature branch in git. Each time a feature is completed, or a portion of any particular feature needs to be tested by another member of the team, the branch will be merged into the develop branch. From the develop branch, another member of the team can run the tests against that particular component and report to the team on the results of that test. If there is a problem with that feature, the resolution(s) to the various problem(s) will be conducted on the feature branch, and then the Quality Assurance process will be repeated.