1 Overview of Proposed System

1.1 Choice of platform

We have decided to use the Glassfish Open Source version as our platform. We did not consider the Oricle Glassfish Server because it would cost to use it for the project.

We considered the following:

- Glassfish Open Source Server
- Google App Engine
- Apache Tomcat

1.1.1 Glassfish Open Source Server

There are a number of benifits to this software above the other options. The main two reasons for using this server are because it is open source and because some members of our group have previous experience with it. Another reason is because we expect there will be support for this enviroment available. Both from the university and from the contributers to the glassfish project. Glassfish has many more features than Tomcat, the other open source option.

Spike testing was carried out and it was found that this peice of software was easy to use and appropriate for the nature of our project.

1.1.2 Google App Engine

The main reason we didn't choose this software is because it prooved unreliable in tests. This software is also closed source and using it would mean that you rely upon google when the application is released.

1.1.3 Apache Tomcat

Tomcat was not as fully featured as Glassfish, and no one in the group has ever used it before, so there would be a steeper learning curve for them and there would be no "in-group" support for using it.

1.2 High Level Archetecture

1.2.1 Version Control

For version control we are using Git. Git is a distributed version control system, which some members of the group already have experience with. Distributed version control systems give a slightly different development pattern which suited the qualities of a group better than SVN.

Version control systems we considered:

- Git
- Bazaar
- Subversion

1.2.2 Integrated Development Enviroment

We have decided to use the NetBeans IDE, because it is available free and it is the preference of the majority of the group. Modules are available for NetBeans to help with Version Control (Git) and JUnit.

IDEs considered:

- Eclipse
- NetBeans

1.2.3 Documentation Tool

We decided to use LaTeX because it is widely supported, there is a template provided, and because it was prefered by the majority of the.

Methods of documentation we considered:

- LaTeX
- Open Office/Libre Office
- Microsoft Word

1.3 Description of Target User

The target user will be young people. Typically aged between 11 and 16. We will have to make sure that no complicated language is used without good reason and we will have to make sure that all content is appropriate. Other things to consider, are:

- Make sure that there are no really lengty tasks to do
- Make sure that it will fit around the lifestyle of a young person of that age. ie. Around school, limited access to a computer.