Tools

1. Programming Language

This application will be written in the Java programming language. Specifically, Java version 1.7 will be used via the Java Platform Tools Standard Edition 7 Development Kit provided by Oracle.

2. Integrated Development Environment

The open source Eclipse integrated development environment, available at http://www.eclipse.org/, will be used for this application. The EGit plugin for Eclipse, available at http://www.eclipse.org/egit/, will be added to the standard Eclipse installation to enable integration with the Git version control system. The M2E – Maven Integration for Eclipse plugin, available at http://www.eclipse.org/m2e/, will be used to package the application into a single executable file.

3. Version Control

All code changes will be tracked using the Git version control system, enabling collaborative coding and a central repository for the master version of the code. The master repository will be stored on Google Code at the following URL: https://code.google.com/p/farkle-csc478/. At the completion of each version of the application, the master branch will be forked, effectively saving a snapshot of the code for that version.

4. Generating the Installer

The .msi installer for this application will be created in MS Visual Studio 2012 using the InstallShield plugin made available by Flexera Software, http://www.installshield.com. The .exe created from the Maven build in eclipse will be used to create this installer.

5. Documentation Editor

All documentation will be initially drafted and maintained in Microsoft Word 2013. Collaboration on these documents will be facilitated via a shared folder on Google Drive.

6. Scheduling

Project scheduling will be completed in Microsoft Project 2013. At the conclusion of the project, the Gantt chart generated in Microsoft Project will be submitted in pdf format.

7. Uniform Modeling Language Tool

The overall architecture of the application will be modeled in Microsoft Visio 2013 using the standards of the Uniform Modeling Language. At the conclusion of the project, the UML diagram will be submitted in pdf format.

8. Online Meetings

Online meetings will be conducted via Google Hangouts twice weekly. Google Hangouts allows desktop sharing, and video conferencing among multiple users. The conference calling line owned by the company Curtis works for will serve as a backup option if any team member is unable to use Google Hangouts for any scheduled meeting.

9. File Sharing

A shared folder has been set up on Google Docs to facilitate the sharing and storage of all files developed for the project that are not stored on Google Code.

Standards

1. Process Model

The application will be developed using the incremental process model with three increments scheduled for completion prior to submitting the final project.

2. Overall Architectural Model

The application will be developed using a variation of the model-view-controller architectural pattern. In this variation, the controller will receive input from the view, send it to the model, receive a response from the model, and update the view. In essence, the controller is responsible for all communication between the model and view.

3. Code Documentation Standards

All source code documentation will follow the Javadoc standard allowing for automatic generation of formatted source code documentation in the Eclipse integrated development environment.