# Overview

In teams of 2 - 3 people, develop a Java project around a topic of interest to you. The project may involve the expansion of something you have done in this class such as a previous assignment, but it may also be about a topic beyond the scope of this course that you want to learn more about. For example, you might do something with Android phones in Java, or you may be interested in games, databases, networks, computer graphics, security, encryption, web applications, or anything else that you would like to learn more about. Anything beyond the scope of this course will be your responsibility to research and learn about.

The scope of the project should be comparable to other course projects. There are some minimum requirements that are described below.

# Minimum Specifications

* The project must have a graphical user interface based on the Swing library in Java (except for Android projects). The main window must be derived from JFrame and have a menu bar and at least one of each of the following: buttons, text fields, labels, and list or combo controls. It must have a caption and its own icon. You should make a conscious decision about whether the window should be resizable to fit the requirements of your application.
* Use control borders appropriately to improve the appearance of your application. Provide for scrollbars if/where they are needed.
* The project must have appropriate listeners for events appropriate to your application.
* There should be an “About Box” which is derived from JDialog and more attractive and professional looking than a JOptionPane dialog.
* You should use icons, pictures, and/or other graphic elements in appropriate ways to enhance your user interface. The window layouts should be attractive and professional looking.
* You should use one or more text files in ways appropriate to your application. Use Java’s built-in dialogs such as JFileChooser, JColorChooser, and so forth as appropriate to enhance your project.
* Your program should handle any exceptions thrown in appropriate ways.
* The application should have an appropriate look and feel. Choose colors and fonts carefully, remembering that your application will be graded on a computer different than yours, and it may have a different screen resolution and different default color schemes and fonts.
* The application should make appropriate use of arrays and/or arraylists.
* Use good object-oriented principles in your design and implementation. Consider using inheritance, polymorphism, and other object-oriented principles where they are appropriate in your project.
* Be sure to include appropriate documentation with your project including the names of all team members with an indication of the projects components for which each team member was responsible.

# Submission and Presentation

Submit a short description of your project and a list of team members first. The description of the project should be detailed enough to show how it satisfies the minimum requirements of this assignment.

Submit your project in the usual manner. Be sure to include any instructions, text files, graphics files, sound files, and other non-code files that are needed to understand, set up, and use your project. Do not expect others to “read your mind” about the project’s purpose, setup, or usage. Include a “readme.docx” file if needed for setting up and running your project. Be sure you set it up and test it on a different computer before finalizing your submission materials.

In addition to the submission, each group will make a presentation to the rest of the class. The presentation should introduce the team, describe the purpose of the project and its interesting features, and demo the project for the class. The presentation and demonstration should show how the project meets the minimum requirements of the assignment. It may include a PowerPoint slide show. Each team member should participate in the presentation. The presentation should be of 30-45 minute duration.