C490/C690 PROJECT REQUIREMENTS AND DUE DATES

Project proposal due:	07/27/2015, 8:00am, 10 points
Project checkpoint report due:	08/03/2015, 8:00am, 10 points
Project final report due:	08/11/2015, 8:00pm, 80 points

PROJECT OBJECTIVES:

- Apply and synthesize the knowledge and skills learned from this course.
- Gain the experience of developing a fairly complex mobile app
- Strengthen the understanding of OOP, Java programming, and Android programming.
- Promote problem solving and self-learning skills

PROJECT TOPIC AND REQUIREMENTS:

1) Project Topic

The topic choice is wide open. You can propose an app of any type: business, comics, communication, education, entertainment, finance, games, health & fitness, lifestyle, media, personalization, productivity, etc.

Here are a few things that might help you to decide what app to propose:

a) Download example apps coming with the textbook, import them to Android Studio, and try them out.

b) Try out the sample codes coming with the Android Studio (from Welcome screen, click on "Import an Android code sample"

c) Try out other open-source Android apps you can find. Textbook chapter 1 lists some of the useful websites.

d) Browse Google's Play Store to get inspired. The following websites might also help:

IdeasWatch: http://www.ideaswatch.com/startup-ideas/app?p=34

AppIdeas: https://newappidea.com/project/view-app-projects/all

2) Project Requirements

The project will be evaluated based on the quality of work and the workload. Here are some minimal requirements for your mobile app:

a) A fun project you can enjoy.

b) A concreate project you can learn a lot through completing it.

c) Apply as many as possible the techniques/skills you've learned.

d) The workload should be comparable to the total load of 3~4 of our programming assignments.

PROJECT PROPOSAL CONTENTS AND FORMAT:

The proposal should be about 2 pages giving a clear description of the app to be developed. It should contain at least the following elements:

1) The purpose and functionality of the app.
2) Draft GUI design, either through Android Studio, or other drawing/editing tools.
3) Major techniques that will be used in the design, and how you plan to learn those techniques, especially if they are not listed in our course schedule. Identify resources.
4) Your work plan.
5) (optional) A comparison with similar apps.

PROJECT CHECKPOINT REPORT CONTENTS AND FORMAT:

The report should be at least 2 page long, describing what have been done by the checkpoint. The following are expected:

1) The complete GUI design with Android Studio, including all the layouts and resources (strings, images, graphic design, audio/video, etc.)
2) You have done significant amount of research, feeling confident on what techniques to use in your project.
3) Finished 20% - 60% of the coding. You don't need to include the code in your report.

PROJECT FINAL REPORT CONTENTS AND FORMAT:

The final report of your project should include the following components:

1) A clear and complete description of the purpose and functionality of your app.
2) List and explain the major techniques you have used in the app. What difficulties you have encountered, how you overcome them.
3) A comprehensive test-run collection that is able to demonstrate the functionalities of your app and unfixed bugs if any. Or a video demo uploaded to YouTube.

What to submit:

1) The final report (including the test-run collection or YouTube link)

2) The zip file of your complete project (your code should be well commented).

Appendix: Characteristics of great apps (from Textbook)

GREAT GAMES

• Entertaining and fun.

• Challenging.

• Progressive levels of difficulty.

• Show your scores and use leaderboards to record high scores.

• Provide audio and visual feedback.

• Offer single-player, multiplayer and networked versions.

• Have high-quality animations.

• Offloading input/output and compute-intensive code to separate threads of execution to improve interface responsiveness and app performance.

• Innovate with augmented reality technology—enhancing a real-world environment with virtual components; this is particularly popular with video-based apps.

USEFUL UTILITIES

• Provide useful functionality and accurate information.

• Increase personal and business productivity.

• Make tasks more convenient (e.g., maintaining a to-do list, managing expenses).

• Make the user better informed.

• Provide topical information (e.g., the latest stock prices, news, severe storm warnings, traffic updates).

• Use location-based services to provide local services (e.g., coupons for local businesses, best gas prices, food delivery).

GENERAL CHARACTERISTICS

• Up-to-date with the latest Android features, but compatible with multiple Android versions to support the widest possible audience.

• Work properly.

• Bugs are fixed promptly.

• Follow standard Android app GUI conventions.

• Launch quickly.

• Are responsive.

• Don’t require too much memory, bandwidth or battery power.

• Are novel and creative.

• Enduring—something that your users will use regularly.

• Use professional-quality icons that will appear in Google Play and on the user’s device.

• Use quality graphics, images, animations, audio and video.

• Are intuitive and easy to use (don’t require extensive help documentation).

• Accessible to people with disabilities (http://developer.android.com/guide/topics/ui/accessibility/index.html).

• Give users reasons and a means to tell others about your app (e.g., you can give users the option to post their game scores to Facebook or Twitter).

• Provide additional content for content-driven apps (e.g., game levels, articles, puzzles).

• Localized (Chapter 2) for each country in which the app is offered (e.g., translate the app’s text and audio files, use different graphics based on the locale, etc.).

• Offer better performance, capabilities and ease-of-use than competitive apps.

• Take advantage of the device’s built-in capabilities.

• Do not request excessive permissions.

• Are designed to run optimally across a broad variety of Android devices.

• Future-proofed for new hardware devices—specify the exact hardware features your app uses so Google Play can filter and display it for only compatible devices