Group Project

Mobile Development with Android

# Introduction

The Programming Mobile Devices course project is a Scrum project with three sprints. Each sprint will last four weeks. Additionally, there is a Sprint 0, which is for planning the three sprints that follow.

You may self-select groups. One group member should host the project on his/her github.com account.

The project does not have one deadline at the end of the semester. This project has 12 deadlines, one per week for 12 weeks, for sprints 1 through 3. You should pull, update, commit, and push once per week, minimum. The source code repository is open for inspection by anyone.

# Sprint 0: Planning Sprint

## Group Assignment

Week 1/2:

* Form groups by using the discussion boards on Canvas.

Week 3:

* Design document due.
* Create online scrum board (scrumy.com or a similar alternative). List the URL in the design document.
* Create online source code repository (github.com).

# Sprint 1: Expand UI

Week 3 – Week 6

* UI
  + Create a Kotlin activity and/or fragment, with layout and ViewModel.
  + Add event handlers: button, motion, etc.
  + Link to LiveData, and Test LiveData connection with ViewModel.
* Integrations
  + Create ViewModel.
  + Test LiveData connections.
  + Find JSON sources to integrate.

Sprint 2, Use Android Device Features; Write Unit Tests.  
Week 7 – Week 10

* Integrate code review recommendations.
* Use one or more of the features of the Android device
  + Accelerometer
  + Threading
  + Orientation
  + Camera/media capture
  + GPS
  + Notifications
  + Contact List
* Consider persistence with either Room DB or FireBase.
* Write Unit Tests for each class.

# Sprint 3: Integrate

Week 11-Week 14

* Apply code review comments from Sprint 2.
* Integrate with one or more external systems. For example:
  + Firebase
    - OAuth Authentication
    - Cloud storage
    - Crashlytics/Fabric
  + Google App Engine: http://code.google.com/appengine/
    - Images
    - rotate, histogram
    - http://code.google.com/appengine/docs/java/images/overview.html
  + A web service.
  + REST-ful service.
  + Facebook, LinkedIn, Twitter, YouTube, Flickr, etc.
* Final Presentation:
  + "Sell" your program to your classmates.
  + Classmates evaluate, provide feedback, review.
  + Program must be marketable and easy to use.
  + Source code must be appropriately commented and readable, so that others can give a description of what the program is doing, without contacting the original author.
* Submit Grade Sheet (below) for Sprint 3.

# Teams and Roles

Each team should have three to five members.   Please submit the grade sheet with your project.  Each team member should submit both the "All Members" section and the individual section for that member.

# Grade Sheet

**All members:**

* Proper use of online Scrum tools and online source code repository (github.com) 10/10
* Code well commented, at class and method level 10/10
* Followed up on specific code review comments 10/10

List of at least three code review comments and corrections made:

My teammates accepted and denied code review comments and corrections. I don’t see any more corrections from code reviews that need to be made.

* Proper use of MVVM design pattern.  10/10
* Code in good form.  Uses standard Kotlin conventions.  Good use of packages, datatypes, etc.  Proper exception handling, when needed.  Null types are properly handled. 10/10
* Program works properly when deployed on AVD emulator.  10/10

**UI Specialist**

* Use both simple and advanced UI controls.  10/10
* Good use of XML layouts.  Layout uses material design, makes sense, is fluid, is viewable.  10/10
* Use LiveData to populate screen.  Business logic is not in Activity, and Activity logic is not in other layers.  10/10
* Use of at least one on board service (GPS, camera, notifications, contacts) 5/10

**Integration Specialist**

* Create the ViewModel and all backend classes, like DTOs and DAOs.  10/10
* Interaction with Firebase, Room Database, filesystem, or all.  Can insert, update, and delete data. 10/10
* Integrate with a RESTful service, and provide the results to the View using LiveData.   10/10
* Unit test coverage of non-UI features is at least 60% 10/10

**DevOps and Product Owner**

In addition to the tasks below, the Dev Ops and Product Owner should take the lead on the design document, due early in the semester.

* Enable team communications. 10/10
  + Schedule and host a Teams standup Thursdays at 6 PM.
  + UC hosts Teams for students You can set up a recurring meeting there.
  + At the first meeting, the team can choose a different meeting tool and different time. If nothing else is selected, assume Teams, Sunday at 8 PM.
  + Most successful groups meet twice weekly, so consider a second meeting.
  + Also, decide on other communications tools to use between meetings: Slack, Discord, Teams, Hangouts, etc.
* Maintain the CI/CD pipeline with CircleCI, or something similar. This pipeline should run unit tests and report the results. 10/10
* Manage GitHub branches. Assist team members with merge issues.  Create and merge branches as needed. Add code review collaborators to your project within 24 hours of request. 10/10
* Determine the Scope and Sprint Goals 10/10
  + Hold Retrospective meetings at the end of each Sprint.
  + Manage Scrum Board, Product Backlog on GitHub or a similar tool. Assist with any backlog items. Actively seek out and remove blockers. Research beyond class topics when necessary.