TEMPLATED ANDROID APP ARCHITECTURE

JEREMY OOSTERBAAN AT PERFORMANCE ELECTRONICS, LTD.

PRESENTATION ROADMAP

- Problem & opportunity
- Overview of the proposed solution
- Design & development
    - User stories, charts, other design diagrams, etc.
- Project constraints & challenges
- Milestones & progress check.
- Expected deliverables by end of semester.
- Division of work & effort matrix
- Expo demo plans

PROBLEM

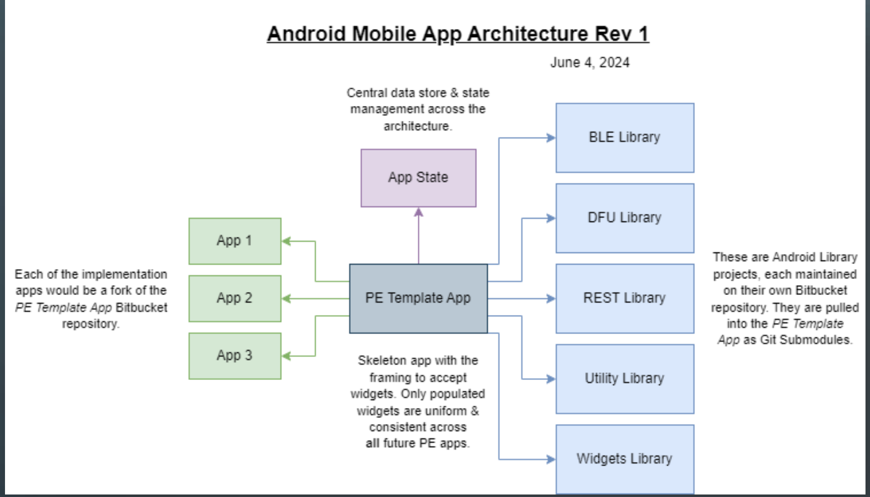
- To explain the problem my capstone project aims to solve, I first need to give some background on my employer, Performance Electronics, Ltd (PE). I am very lucky to be working there and having development funded by the company this year. PE is an embedded systems company, meaning we have both hardware & software components to our products. The software component(s) is largely comprised of mobile apps which talk to various hardware modules over Bluetooth Low Energy (BLE). Until now, the process of creating new mobile apps for each new project is incredibly tedious. We have to more or less start from scratch each time. There exists little to no reusability in our current mobile app structure, which creates a very inefficient development process.

MY PROPOSED SOLUTION

- The ultimate goal is to save development time & costs when creating mobile apps for new projects. The solution I proposed is a sort of “templated” structure where we have one consistent, generic framework or outline app. Additionally, we implement a collection of UI widgets that can be populated at will throughout the app. Each project would be a clone of the template app repository, but with a different “configuration” of widgets & themes. Moreover, we can aim for a less monolithic application and closer follow the microservices architecture, in which the various components of the app are distinct, independent libraries/modules that work together through a generic publisher/subscriber communication protocol.

DESIGN DIAGRAMS

- Microservices architecture (old revisions)



Android Mobile App Architecture Rev 1

June 4, 2024

Central data store & state management across the architecture.

App State

App 1

App 2

App 3

PE Template App

BLE Library

DFU Library

REST Library

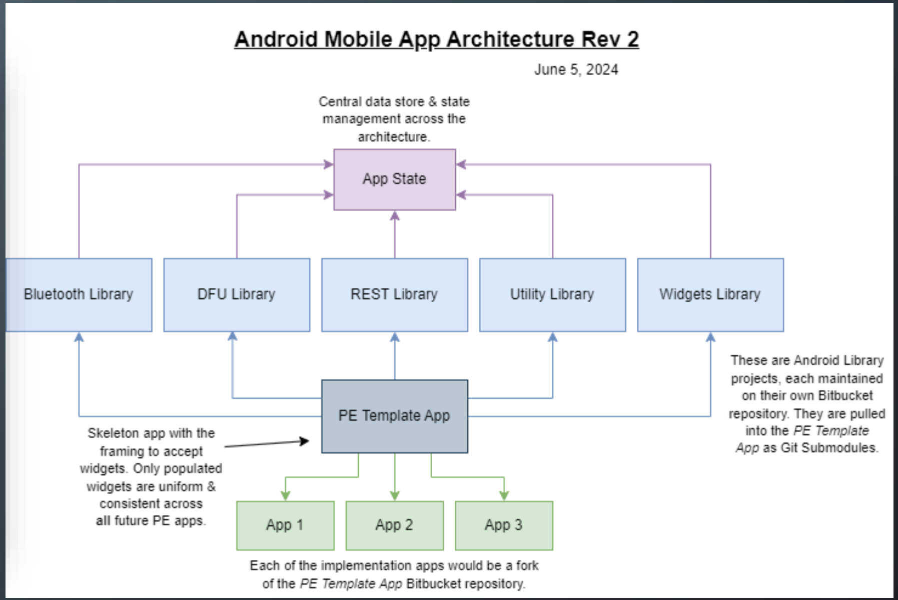
Utility Library

Widgets Library

Each of the implementation apps would be a fork of the *PE Template App* Bitbucket repository.

Skeleton app with the framing to accept widgets. Only populated widgets are uniform & consistent across all future PE apps.

These are Android Library projects, each maintained on their own Bitbucket repository. They are pulled into the *PE Template App* as Git Submodules.



Android Mobile App Architecture Rev 2

June 5, 2024

Central data store & state management across the architecture.

App State

Bluetooth Library

DFU Library

REST Library

Utility Library

Widgets Library

PE Template App

App 1

App 2

App 3

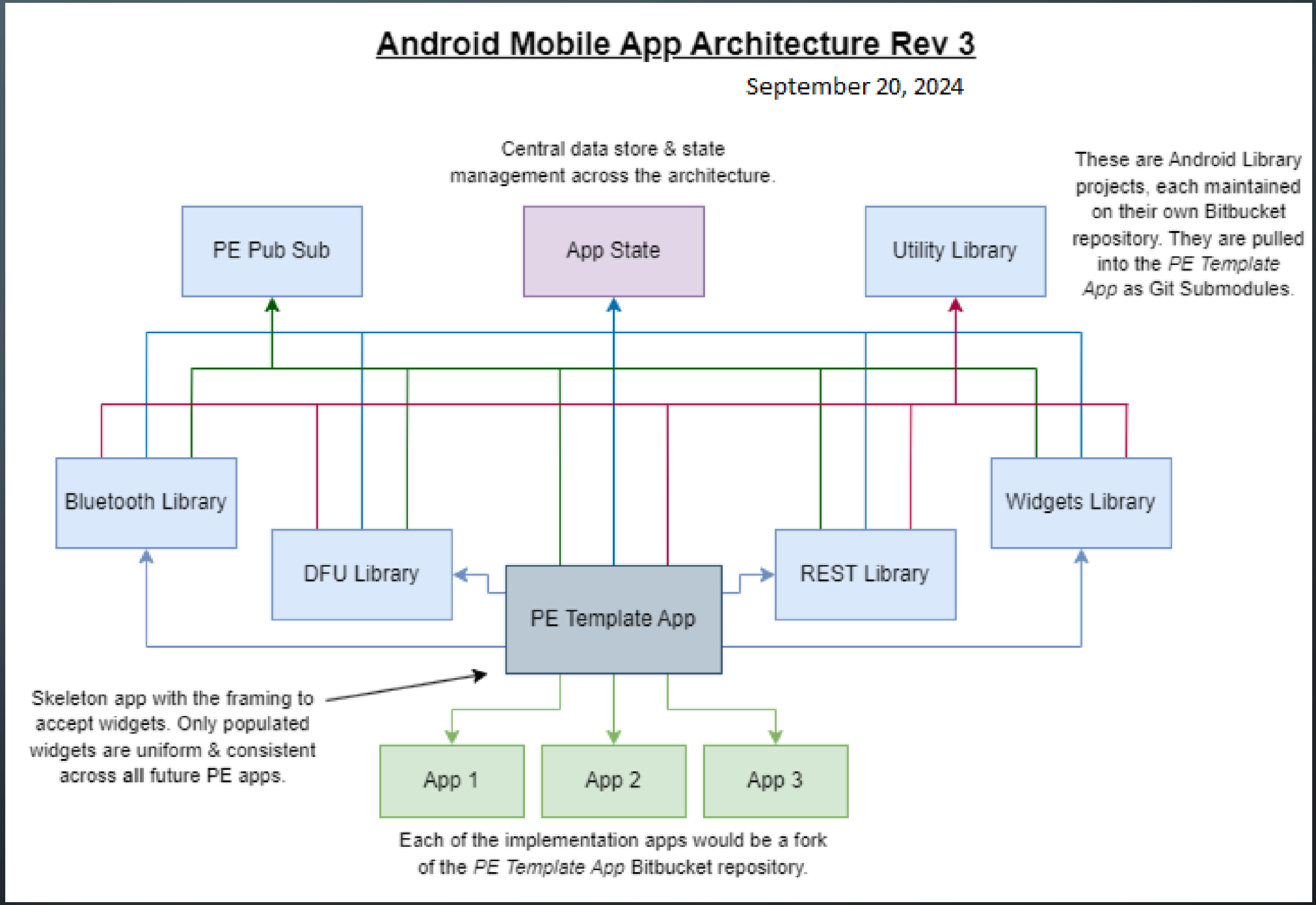
Skeleton app with the framing to accept widgets. Only populated widgets are uniform & consistent across all future PE apps.

These are Android Library projects, each maintained on their own Bitbucket repository. They are pulled into the *PE Template App* as Git Submodules.

Each of the implementation apps would be a fork of the *PE Template App* Bitbucket repository.

DESIGN DIAGRAMS CONT.

- Microservices architecture



Android Mobile App Architecture Rev 3

September 20, 2024

Central data store & state management across the architecture.

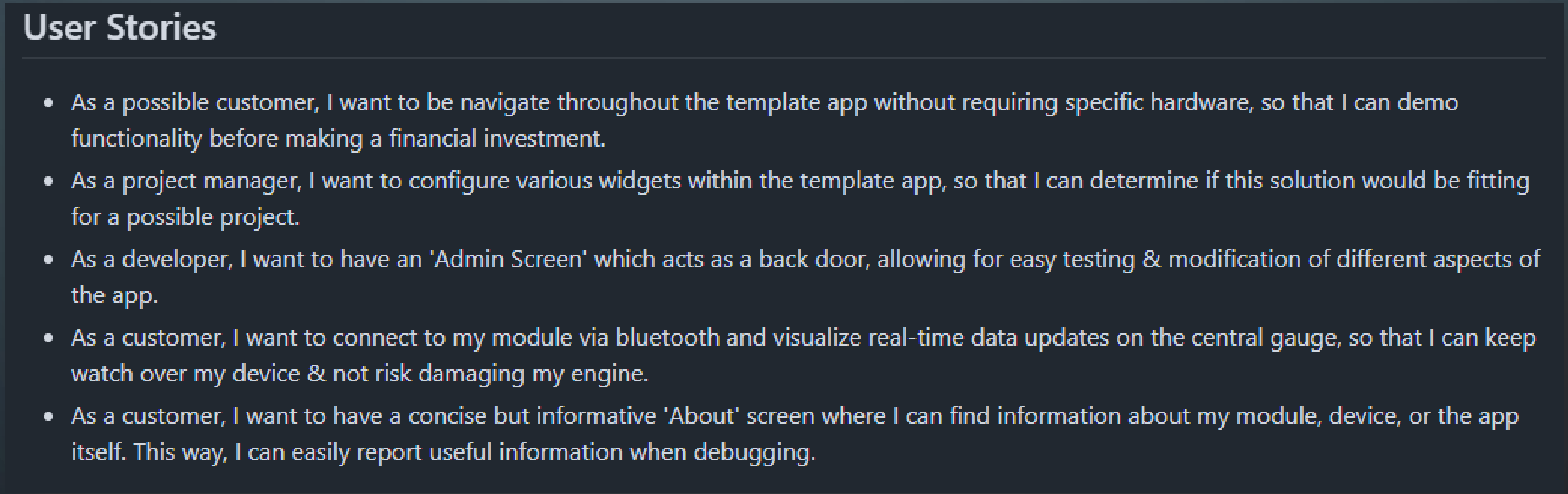
These are Android Library projects, each maintained on their own Bitbucket repository. They are pulled into the *PE Template App* as Git Submodules.

Skeleton app with the framing to accept widgets. Only populated widgets are uniform & consistent across all future PE apps.

Each of the implementation apps would be a fork of the *PE Template App* Bitbucket repository.

DESIGN DIAGRAMS CONT.

- User Stories



User Stories

- As a possible customer, I want to be navigate throughout the template app without requiring specific hardware, so that I can demo functionality before making a financial investment.
- As a project manager, I want to configure various widgets within the template app, so that I can determine if this solution would be fitting for a possible project.
- As a developer, I want to have an 'Admin Screen' which acts as a back door, allowing for easy testing & modification of different aspects of the app.
- As a customer, I want to connect to my module via bluetooth and visualize real-time data updates on the central gauge, so that I can keep watch over my device & not risk damaging my engine.
- As a customer, I want to have a concise but informative 'About' screen where I can find information about my module, device, or the app itself. This way, I can easily report useful information when debugging.

More design diagrams can be found on the GitHub repository: https://github.com/JPEG1/Senior-Design/tree/main

PROJECT CONSTRAINTS

- Developers & development resources
  - Small company means much more tedious & careful in resource allocation
  - Long list of tasks (see TaskList.md on GitHub) for solo developer
- Hardware
  - Products implementing BLE adds a whole layer of complexity
  - Asynchronicity & independent publisher/subscriber message processing protocol
  - Collaboration with the hardware department

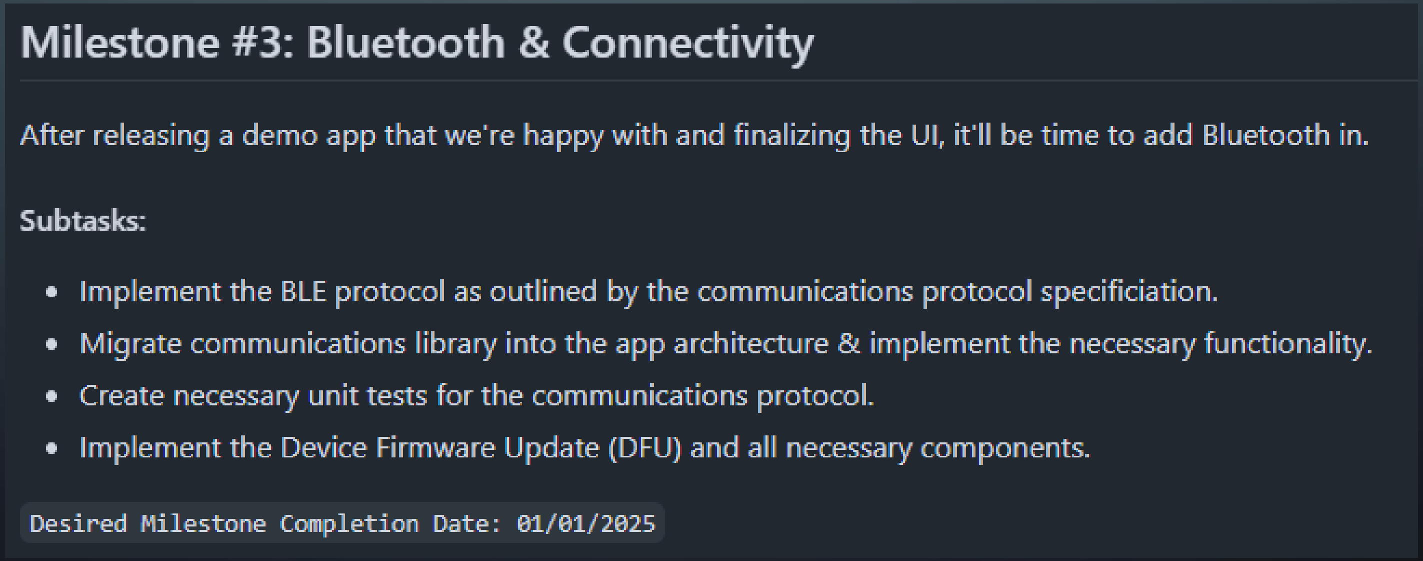
PROJECT MILESTONES

- Milestone #1: Template App Architecture Design
  - Designing all aspects of the apps such as communication processing structure, device firmware update (DFU) process, PE pub sub, widgets spec, etc.
- Milestone #2: Demo Mode UI App
  - Template app framework with completed collection of widgets.
- Milestone #3: Bluetooth & Connectivity
  - Communication processing library & BLE library.
- Milestone #4: Comprehensive Test Framework
  - Unit tests for each individual microservice and system tests for entire architecture.
- Milestone #5: Finalized Documentation & Deliverables
  - Final documentation for expo in Spring 2025.

- More comprehensive & detailed breakdown of milestones can be found on GitHub.

PROGRESS CHECK

- Currently, as of the week of October 20, 2024, I am working on project milestone #3. I am presently working on a comprehensive test suite to enable BLE communication. This goes in line with the timeline that I set at the beginning of the semester. I aim to have this milestone completed by the end of 2024.



Milestone #3: Bluetooth & Connectivity

After releasing a demo app that we're happy with and finalizing the UI, it'll be time to add Bluetooth in.

Subtasks:

- Implement the BLE protocol as outlined by the communications protocol specificiation.
- Migrate communications library into the app architecture & implement the necessary functionality.
- Create necessary unit tests for the communications protocol.
- Implement the Device Firmware Update (DFU) and all necessary components.

`Desired Milestone Completion Date: 01/01/2025`

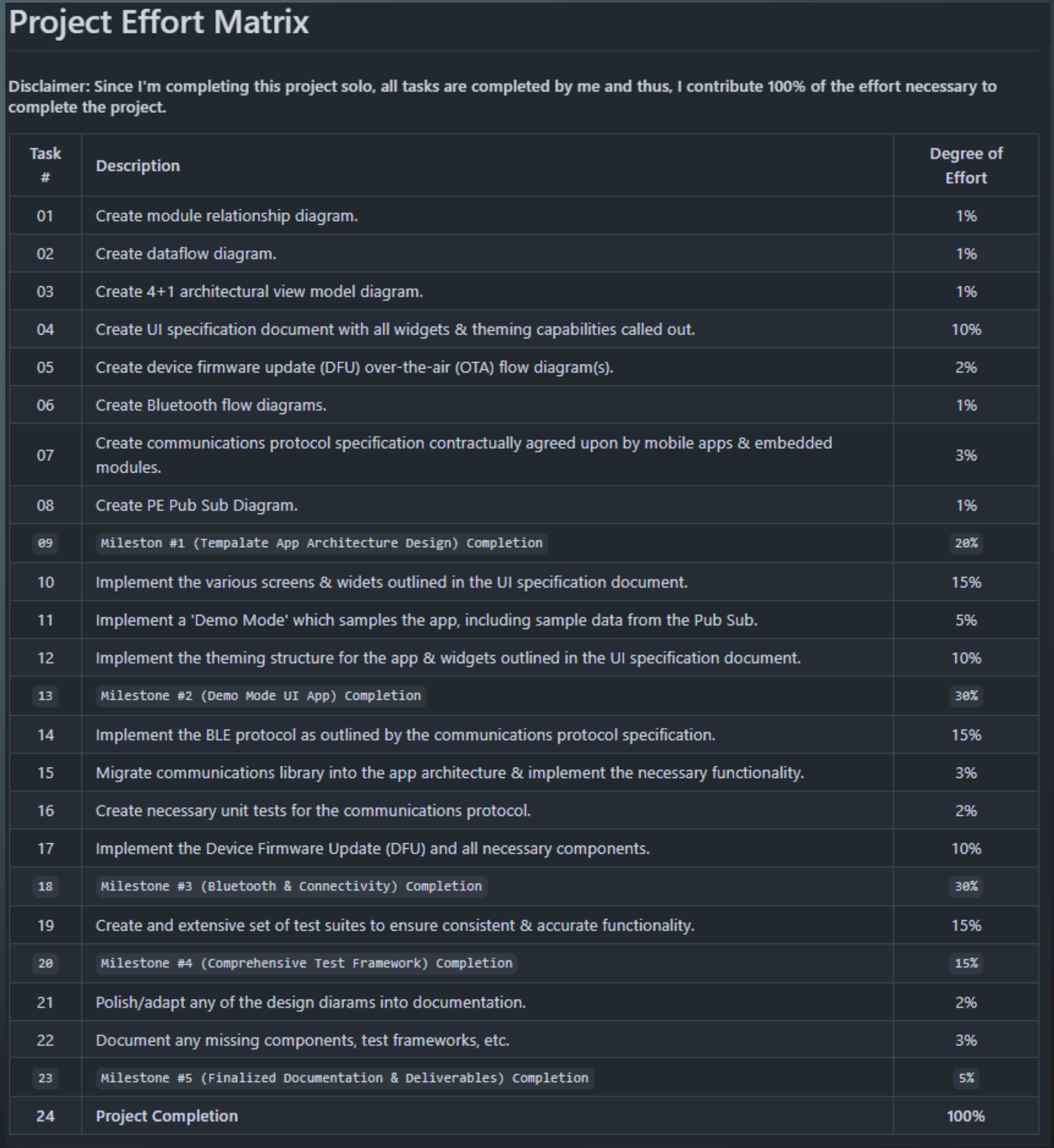
END OF SEMESTER DELIVERABLES

- My timeline outlines that I'll be nearing the completion of milestone #3 by the end of this semester. Thus, there isn't a list of concrete deliverables that I plan on supplying at the end of the semester. The semester break for me, is nothing significant in the scope of this project. I'll continue development past the end of the semester in order to reach the end goal next April/May.

DIVISION OF WORK

- Since I’m completing this project solo, 100% of project effort falls on me.

The effort matrix on the right can be found on GitHub.



Project Effort Matrix

Disclaimer: Since I'm completing this project solo, all tasks are completed by me and thus, I contribute 100% of the effort necessary to complete the project.

Task #	Description	Degree of Effort
01	Create module relationship diagram.	1%
02	Create dataflow diagram.	1%
03	Create 4+1 architectural view model diagram.	1%
04	Create UI specification document with all widgets & theming capabilities called out.	10%
05	Create device firmware update (DFU) over-the-air (OTA) flow diagram(s).	2%
06	Create Bluetooth flow diagrams.	1%
07	Create communications protocol specification contractually agreed upon by mobile apps & embedded modules.	3%
08	Create PE Pub Sub Diagram.	1%
09	Mileston #1 (Tempalate App Architecture Design) Completion	20%
10	Implement the various screens & widets outlined in the UI specification document.	15%
11	Implement a 'Demo Mode' which samples the app, including sample data from the Pub Sub.	5%
12	Implement the theming structure for the app & widgets outlined in the UI specification document.	10%
13	Milestone #2 (Demo Mode UI App) Completion	30%
14	Implement the BLE protocol as outlined by the communications protocol specification.	15%
15	Migrate communications library into the app architecture & implement the necessary functionality.	3%
16	Create necessary unit tests for the communications protocol.	2%
17	Implement the Device Firmware Update (DFU) and all necessary components.	10%
18	Milestone #3 (Bluetooth & Connectivity) Completion	30%
19	Create and extensive set of test suites to ensure consistent & accurate functionality.	15%
20	Milestone #4 (Comprehensive Test Framework) Completion	15%
21	Polish/adapt any of the design diarams into documentation.	2%
22	Document any missing components, test frameworks, etc.	3%
23	Milestone #5 (Finalized Documentation & Deliverables) Completion	5%
24	**Project Completion**	**100%**

EXPO DEMO

- Subject to change
- Will need one (or more) mobile apps
  - Some of which have already been released on Google Play Store
- Need hardware to communicate with each app being used
  - i.e. 2 different mobile apps = 2 unique modules
- Since the project is the architecture itself, the best demo I can give is showcasing how this architecture is beneficial. Highlighting specific project functionality isn't necessary. Rather, showing different apps using the same architecture (saving development time) is key.

THANKS FOR YOUR TIME!

Check out more documentation on GitHub.