

How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: “**Capstone_Stage1**”
3. Replace the text in green

Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it “**Capstone Project**”
3. Add this document to your repo. Make sure it’s named “**Capstone_Stage1.pdf**”

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you’ll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: Your GitHub username here

JuiceBoard

Description

JuiceBoard is a Dashboard of battery status of all your devices!

Ever had multiple devices / tablets and when you need them you realize they are dead? Or ever had a loved one keeps letting their battery down and becomes unreachable? JuiceBoard to the rescue.

The app keeps track of battery status and alerts of all your devices in one dashboard. Attend the device that is dieing soon and Go Charge it!

Intended User

Anyone who has to keep track of battery status of their or someone else's device.

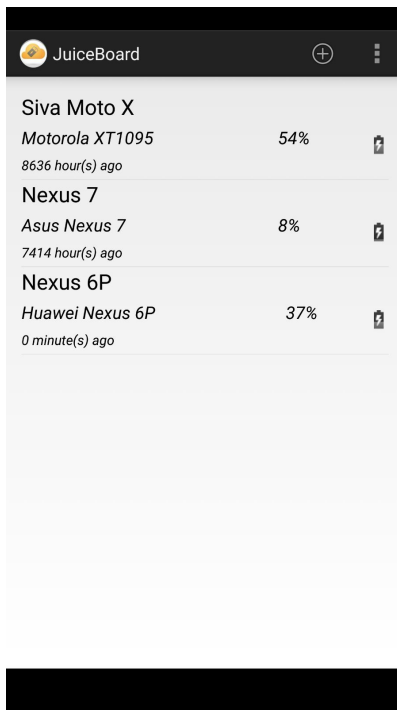
Features

- All android devices (soon iOS)
- Sends notifications for low battery of oter
- Other features like that

User Interface Mocks

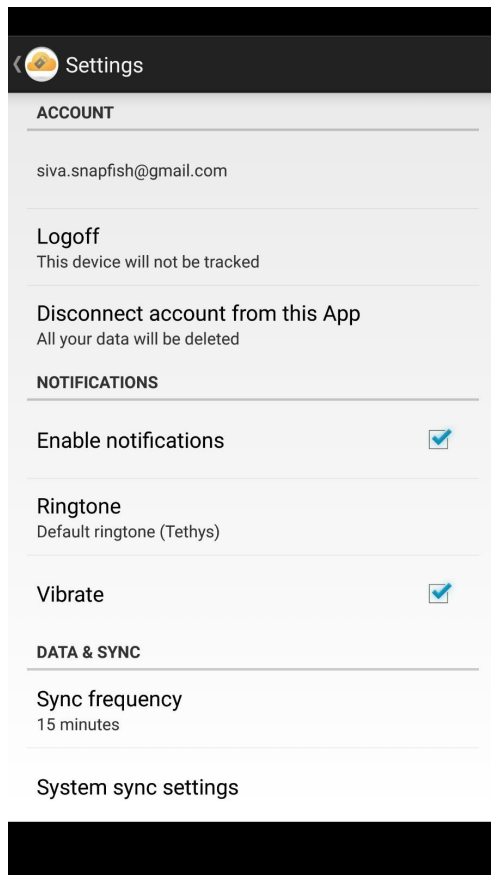
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



This is the main screen with list of all signed devices.

Screen 2



This is the Settings screen with basic usual settings.

Key Considerations

How will your app handle data persistence?

The app will persist data via Content Provider / SQL and use cloud (Parse.com) to store as well.

Describe any corner cases in the UX.

The UI is simple. Has a dashboard and settings page.

Describe any libraries you'll be using and share your reasoning for including them.

Will use Parse.com SDK. May use other simple libraries for UI look and feel.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

- Create a new Android Project Setup
- Setup Parse SDK integration dependency for the app module.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for Settings
- Create (or make use of existing) a Fragment to show the list of devices.
- Pull from cloud (user Parse.com SDK), persist locally and display the list of devices with battery status on the UI.

Task 3: Your Next Task

Create Sync Adapter for data syncing and notify when a device is low on juice.

Describe the next task. List the subtasks. For example:

- Use a Sync Adapter to pull from cloud periodically.
- Display a notification if a device is in critical status.

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]

2. Create a new GitHub repo for the capstone. Name it “**Capstone Project**”
3. Add this document to your repo. Make sure it's named “**Capstone_Stage1.pdf**”