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.
Describe how you will implement Google Play Services.
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: AkshayMoorthy

Dawn 2 Dusk

Description

Dawn 2 Dusk is a simple app used to track everyday Sunrise & Sunset time based on the user location. The app also allows user to see the Sunrise & Sunset time on any date through the years.

Intended User

The app is for families to keep track of Sunrise and Sunset time in their location.

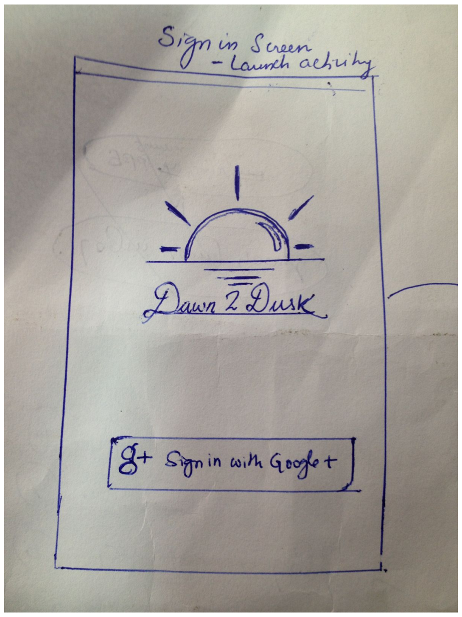
Features

- Authenticates user using Google Sign-in.
- Stores Sunrise and Sunset time data.
- Uses user location to give localised data.

User Interface Mocks

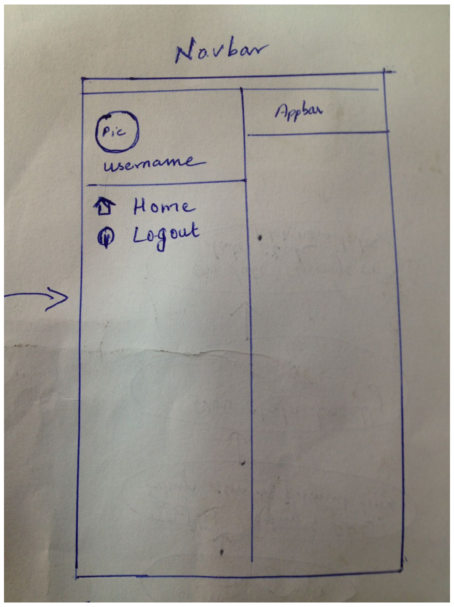
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



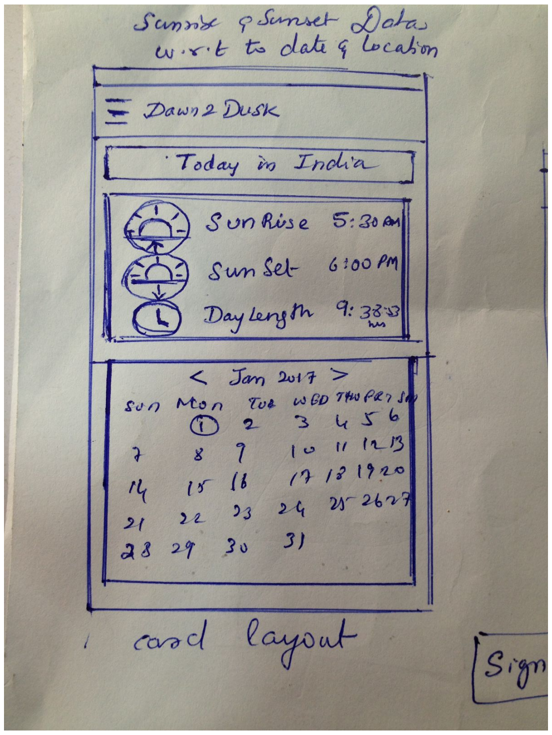
Sign in Screen is the launch activity that uses Google login for authentication.

Screen 2



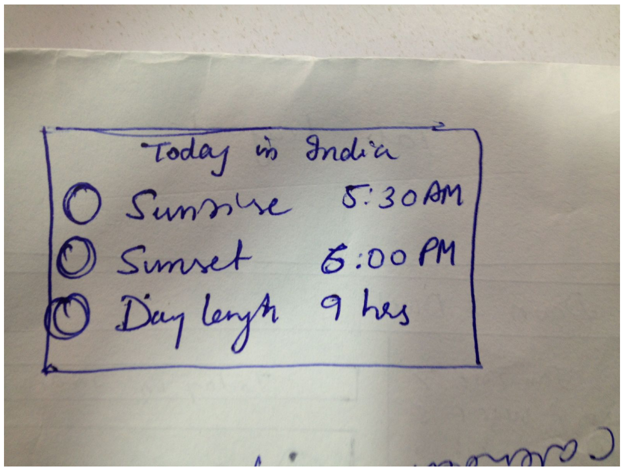
NavBar to navigate to home screen that displays localised Sunrise and sunset data, or logout to launch activity.

Screen 3



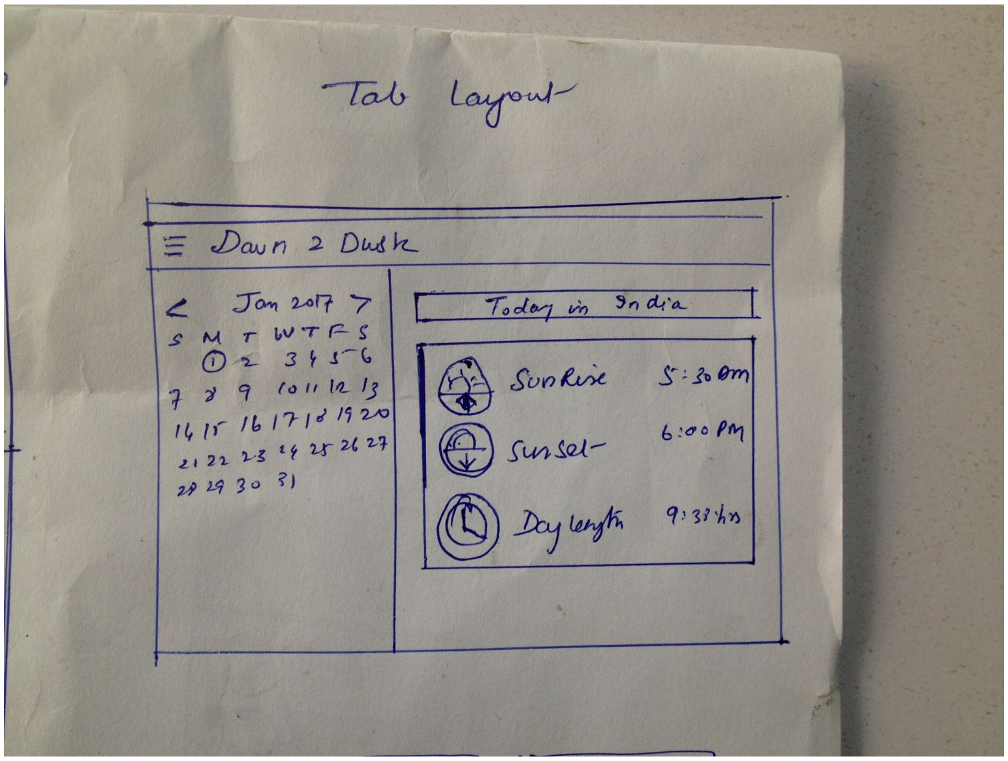
Main Activity or HomeActivity that uses card layout to display the user location based Sunrise, Sunset and Day-length data corresponding to the date chosen in the calendar below(Calendar view is a third party library).

Widget UI



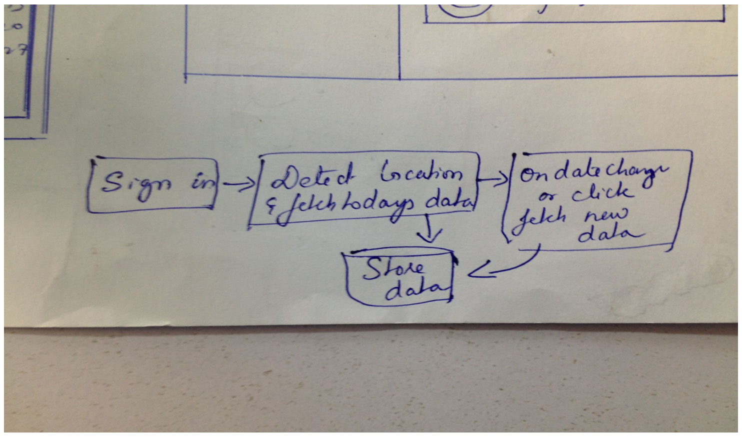
Widget for the Dawn 2 Dusk app has a simple UI that displays current date and locations Sunrise, Sunset and Day-Length Data.

Tablet UI



Tablet View has two sections in the screen, where the left section shows the calendar view to select date and the right section shows the corresponding Sunrise, sunset and daylength data for the user location in cardlayout.

App Flow


Sign in
Detect location & fetch todays data
On date change or click fetch new data
Store data

The user signs in using google account and the app detects user location. The app then fetches today's sunset, sunrise and day-length data, stores it locally, loads and then displays it in cards. When the user selects a new date or when the date is changed, the new data is fetched from server and then cached locally.

Key Considerations

How will your app handle data persistence?

The app will use Content Provider to store and retrieve the Sunrise, Sunset and Daylength data.

Describe any corner cases in the UX.

On back press the app exits as there is only one main screen.

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

Glide to handle the loading and caching of images.
Material CalendarView Library to add calendar view to the UI for selecting dates through the years.

Describe how you will implement Google Play Services.

Google Sign In API for logging in and authenticating the app user and getting the user data for local use.
Location API for detecting the device location and sending it to endpoint to get localised data from the server.

Reverse GeoCoding API to get the User Area from the detected Latitude and Longitude values.

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

- Test API endpoints.
- Configure Google Sign In API.
- Configure Material Calendar View Library
- Configure Location API
- Configure Android Design and Support Library to implement Material Design.

Task 2: Implement UI for Each Activity and Fragment

- Build Login Page.
- Build HomeActivity.
- Build AppBar for HomeActivity.
- Build NavBar for the HomeActivity.
- Build HomeFragment with cardviews and Calendarview in the HomeActivity.
- Build alternate layout fragment for tablet view.

Task 3: App Flow and Behaviour

- Implement Google Sign In and store user info using shared preferences.
- Populate user info in NavBar.
- Add Menu items to NavBar.
- Link UI to Nav Bar.
- Implement GLIDE to load user image.
- Fetch the current date on Homeactivity creation.
- Implement Location API to detect current location during HomeActivity creation.
- Implement Material Calendar View to get the date on click or change and mark the selected date.
- Implement AsyncTask by passing the location and date to the respective API for the Sunrise,Sunset and Day-Length data.

Task 5: Data Handling

- Parse the JSON response received
- Cache the data using SQLlite.
- Implement Content Providers to Access Locally stored data.
- Implement Loaders to move data to its view.
- Update the cardviews with the data.
- Implement GLIDE to load images.

Task 6: Views

- Check if the app updates view for respective dates.
- Check if all the errors are handled and app doesnt crash.
- Check if pressing back button closes the app.
- Check if all the data is stored and retrieved correctly.
- Build and Implement widget for the app.
- Build and implement tablet view of the app.

Task 7: Run, Test and Upload

- Check if app meets all specifications given.
- Build, run and test the app.
- Sign the app and generate a signed build.

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**"