How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
2. Name your document file: “**Capstone_Stage1**”
3. Replace the text in green

GitHub Username: vrking39

MyHoroscopeDaily

Description

Ever feel the hassle of going to your web browser just to check the daily prediction for your horoscope? Even worse is when you go for these searches you still have to scroll around just to find the daily prediction section that you are looking for. If you don’t like that hassle, then MyHoroscopeDaily is for you! This app is an easy to use horoscope app to get your daily horoscope prediction.

Intended User

This app is intended for users who want to get a daily update of their horoscope easily accessible on their mobile devices.

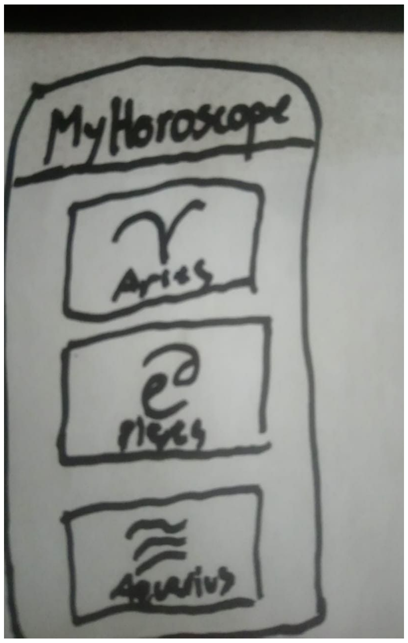
Features

List of main features of the app.

- Access to daily predictions for each astrology sign.
- Notification to remind you about checking your daily horoscope.
- Option to add a widget to your home screen to see daily predictions from there.
- The app will be solely written in the java language.
- The app will make use of stable versions of dependencies, gradle, and Android studio.

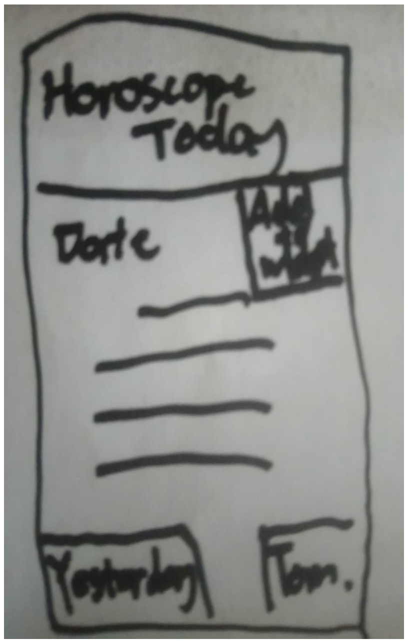
User Interface Mocks

Screen 1



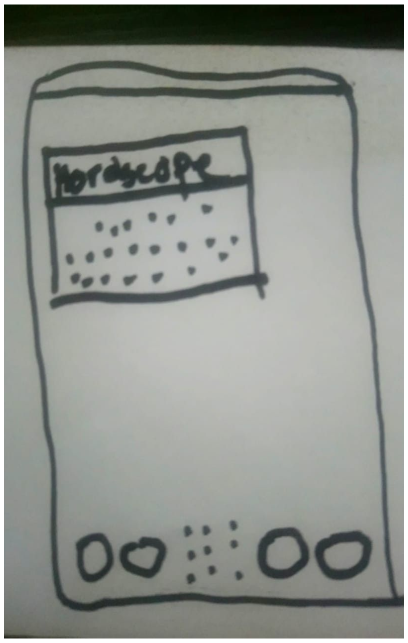
At launch, the user will be prompted to this home screen that contains a recycler view. The view contains all the astrological signs the user can choose from.

Screen 2



This screen will be launched when the user selects an astrological sign. This will contain the daily prediction for the selected sign. It will also post the date today. It will also include two buttons to look at predictions from yesterday and for tomorrow. An add to widget button will also be included.

Widget Screen



This widget will contain the horoscope selected by the user and the daily prediction of it. On click, the app will be launched to the screen of the selected horoscope.

Key Considerations

How will your app handle data persistence?

The app will locally save the selected astrological sign to display on the widget.

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

Retrofit library to be used for making API calls.

Picasso library to be used for putting images in the app.

ViewModel library will be used to preserve data on screen rotation or when app is put in background

Describe how you will implement Google Play Services or other external services.

WorkManager to schedule daily notifications and widget.

Google ads services to place ads at the screens of the app

Next Steps: Required Tasks

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

Task 1: Project Setup

List the subtasks:

- Configure libraries that will be used in the app
- Find an API that will give data about horoscope
- Add permissions such as internet

Task 2: Implement UI for Each Activity and Fragment

List the subtasks:

- Build UI for MainActivity that will show a list of all existing astrological symbol
- Build UI for DetailActivity that would show the daily prediction of the selected horoscope
- Detail activity should contain a “Yesterday” and “Tomorrow” button to navigate to the predictions for the other days.
- Should have a widget button to add the astrological sign to the widget.

Task 3: Understand the API

List the subtasks:

- Check the API
- Build a model for the data taken from the API
- User Retrofit to get the data

Task 4: Implement methods

List the subtasks:

- Create classes that will display data to the UI
- Use Intent to launch the detail activity with the right data.

Task 5: Implement widget

List the subtasks:

- Create a class to implement the widget

Submission Instructions

- After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]

    - Make sure the PDF is named “**Capstone_Stage1.pdf**”
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

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