

How to Use this Template

1. Make a copy [File → Make a copy...] s
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

Horoscope- Present and Future

Description

Horoscope app gives you a quick way to get daily, weekly, monthly and yearly horoscope updates on all the twelve zodiac signs.

-Easy to install and use.

-Professional forecast

Find your western astrology zodiac sign:

Aquarius (January 20 - February 18)

Pisces (February 19 - March 20)

Aries (March 21 - April 20)

Taurus (April 21 - May 20)

Gemini (May 21 - June 21)

Cancer (June 22 - July 22)

Leo (July 23 - August 23)

Virgo (August 24 - September 22)

Libra (September 23 - October 22)

Scorpio (October 23 - November 21)

Sagittarius (November 22 - December 21)

Capricorn (December 22 - January 19)

Intended Users

This app can be used by all who wish to get their horoscope update within 1 minute of their time.

Features

- Provides daily, weekly, monthly, yearly updates of all twelve zodiac signs.
- Provides Company location on google maps.
- Ability to check your daily horoscope updates from home screen Widget
- Share your prediction with Android sharing features.

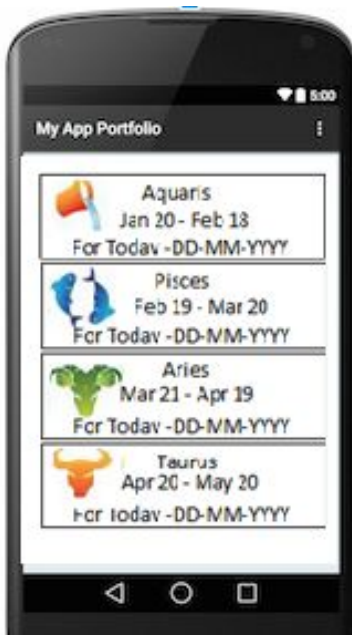
User Interface Mocks

Screen 1 - Home Screen



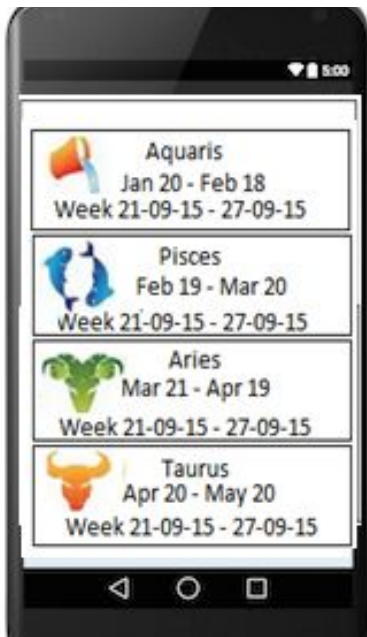
Home Screen - Lets the user select Horoscope based on day, week, month or year. User can also select Map to view nearest astrologer location.

Screen 2 - Horoscope for Today List Page



Get Horoscope for the day .The navigation is from the Home Page.

Screen 3 - Horoscope for the Week List Page



Get horoscope Details for upcoming Week for all the star signs. The navigation is from the Home Page.

Screen 4 - Horoscope for the Month List Page



Get Horoscope for the present Month for all the star signs. The navigation is from the Home Page.

Screen 5 Horoscope for the year List Page



Get Horoscope for the Year for all the star signs. The navigation is from the Home Page.

Screen 4 - Horoscope Details Page



Get Details of the Horoscope for the Day,week ,month or the year depending on the previous selection.

Share your horoscope result with family and friends using other apps.

Key Considerations

How will your app handle data persistence?

Data persistence will be handled by the app using the content adapter and Sync Adapter. The App will use the RSS feed from the <http://www.findyourfate.com/rss/horoscope-feed.asp> - site. All the network operations will be placed in separate module so that the Activity and the home screen Widget can use the service to update data from the update service call.

Describe any corner cases in the UX.

The app launches with the home screen where the user can navigate to Listing page depending on selection on the home screen and then to the details page depending on the selection of the listing page. Back button help user to navigate from the details page to listing page to home page.

User can select contact us to check location of company in google maps.

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

- Picasso to handle the Image Library.
- Google Maps for company Location.
- Google Admob For Ad Banners.

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

Add library dependencies to build.gradle file

- com.android.support:design
- com.android.support:recyclerviewv7

- com.google.android.gms:playservicesmaps
- com.squareup.picasso:picasso:2.5.0

Get api keys

- Get key for Google Maps service.
- Get key for Google AdMob Service.

Create git repository

Add .gitignore file.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Home Screen.
- Build UI for Home Screen fragment.
- Use
- Build UI for Listing View Screen.
- Build UI for List View Screen fragment.
- Build UI for Details Views Screen.
- Build UI for Details View fragment.
- Build UI for Company View screen with Google map.
- Build Google ads using Addmob to display in Activity.

Task 3: Implement Content Provider

Define the Horoscopebydate table to store horoscope data for a day.

- Star Sign
- Title
- Description_Today
- Description_ThisWeek
- Description_ThisMonth
- Description_Thisyear

Classes to create:

- HoroscopeContract
- HoroscopeOpenHelper

- HoroscopeProvider

Task 4: Build Cursor Adapter , Cursor Loader and updateservice

- Build class Article loader and extend from Cursor Loader.
- create updateservice and extend inentservice use the class to get data from service and update the db tables.
- use content Loader to update on Activity.

Task 5: Connect Data with UI

- Connect ContentProvider with Activities and Fragments via Loaders
- Create HoroscopeHelper to keep all interactions with ContentResolver within it.
- Handle Error case for all data views.
- Use Add Banners and update in the Activity.

Task 6: Create a Home Screen Widget

- Create a Home Screen Widget and get the data from Content provider for Todays horoscope and update the UI.

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**"