

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: deepakgarg0802

News Digest

Description

Stay updated with trending news. Read the best news from national & regional newspapers & websites. News is fetched from newsapi.org. It covers in-depth reporting of the current happenings in the country, gets you investigations and explained on news that matters by

showing details in the webview. The app gives you the option to bookmark any news so that you can retrieve it later. The app also has option to share the news.

Intended User

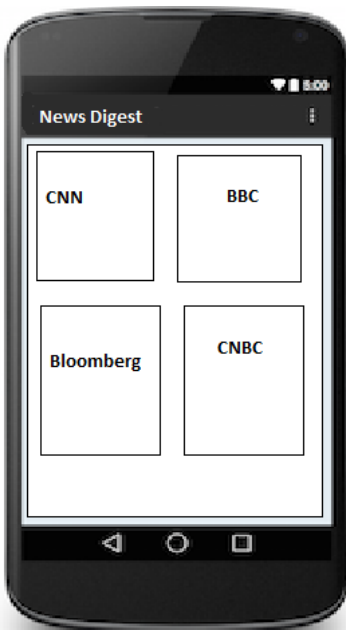
From students to professionals this app is for everyone who wishes to grasp all the news at one place. Feeds of all quality will be catered. Aspirants for government jobs can use it as a source of current affairs.

Features

- Thumbnails of different sources at the main activity
- Dig a little deeper by browsing particular source
- Details displayed in webview
- Bookmark any article
- Widget to show headline

User Interface Mocks

Screen 1



This is the Home Screen to display all sources. It will also contain option to view the bookmarks. Clicking on any tile will lead to a detail activity.

Screen 2



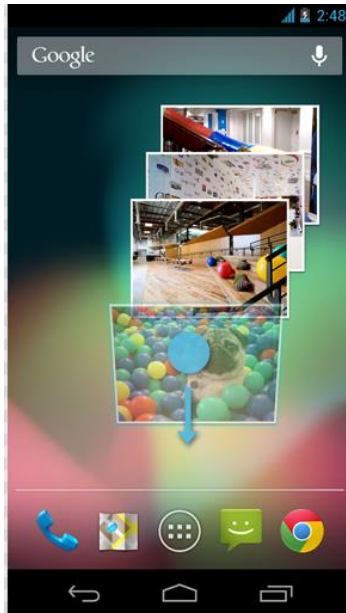
This is the detail activity. User can swipe left to right for more feeds of same kind. It has option to bookmark the article and ad displayed in the bottom.

Screen 3



This is the webview activity.

Widget



App widget will show bookmarked items on home screen like photo gallery widget. Clicking on any will open that news in webview.

Key Considerations

How will your app handle data persistence?

A content provider will be used along with sqlite to store data

Describe any corner cases in the UX.

There may not be any favourites marked by user. In that case message should be displayed. Also network access state should be displayed.

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

Glide to handle the loading and caching of images.
Volley to Loading data from API

Describe how you will implement Google Play Services.

Adwords : Interstitial ad between main and detail activity.

Firebase : Realtime analytics

Next Steps: Required Tasks

Task 1: Project Setup

1. Create a new project in Android Studio.
2. Add the Google Play Services dependency to the Gradle file.
3. Configure the AndroidManifest.xml to enable the various permissions necessary for accessing the internet and the user's location.

Task 2: Implement UI for Each Activity and Fragment

1. Build UI for MainActivity including list of sources
2. Build UI for the DetailActivity and Webview including
 - a. button to bookmark event for later viewing
 - b. button to share news.
 - c. a link to the news displayed in webview

Task 3: Implement Data Fetching and Storage

1. Login to newsapi.org to get a unique api key.
2. Test with api key in your browser that it works or not.
3. Implement content provider
4. Use RecyclerView to display data in list form.

Task 4: Error Handling

1. Find and handle errors with the functionality of the app including
 - a. When there is no access to a network
 - b. When there is no data returned from the API

Task 5: Allowing for Localization and Making the App Accessible

1. Make sure RTL layout switching is enabled for all layouts
2. Provide content descriptions for all icons and images.
3. Ensure the app can be navigated using a Dpad.

Task 6: Build a Collection Widget

1. Declare AppWidgetProvider in AndroidManifest.xml
2. Implement widget

Task 7: Polish App By Implementing Material Design Principles

1. Pick a color scheme with primary and accent colors.
2. Make sure text and images have enough space around them and are aligned along keylines.