GitHub Username: rajatparihar1994

inDewas

Description

inDewas is a local news app for our city Dewas. It provides all the summarised stories contain only headlines and facts, to help you stay informed about our city. This App is easy to operate and user can also choose the types of information which they want to read.

Intended User

This app is for all the user who can understand English and are the resident of Dewas. This app is only for android platform so user must have a android device. This app is for all types of user like Students, Businessman and woman, Families etc.

Features

:

- Show news or facts in English.

- Show the images related to the content.
- If internet connection is slow then user can also turn off the image so the app can work perfectly even in slow internet connection.

User Interface Mocks

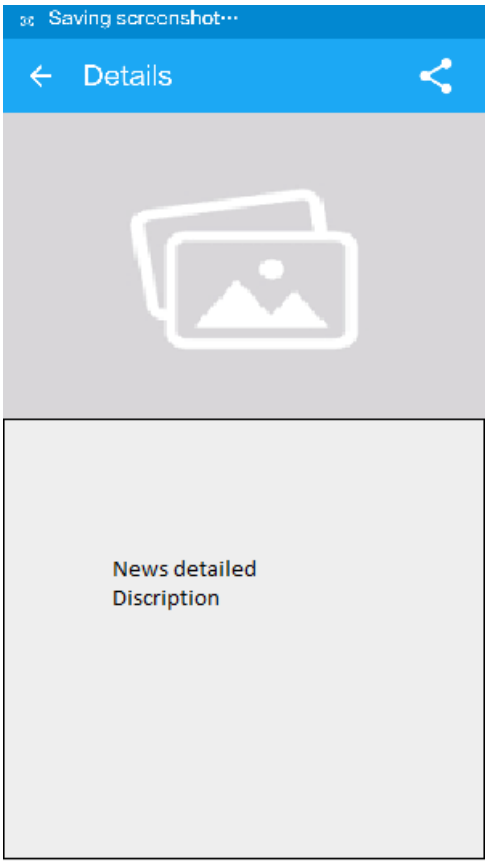
MainActivity

This is the first screen which contains the List of all the news and it will be the MainActivity. From this screen we can redirect to the detailActivity and thorough the menu user can select the option of showing the images etc.

MainActivity screen content

- News Image
- News Headline
- Menu Option

DetailActivity



This is the basic concept of the second screen which is DetailActivity, it will provides the Detailed view of the news including image. Through this Avtivity the user can share the news with different people using the share button in the menu as action.

DetailActivity screen content

- News Image
- News Headline
- News Detailed Description
- News share button.

Widget

The app will also have widget which will allow the user to set it on the homescreen and scroll the news without opening the app through the widget the user can navigate to the detailed news app screen.

Key Considerations

How will your app handle data persistence?

For backend this app will fetch data from the Firebase. This app is connect to the firebase whenever the app is start and will fetch new data from the cloud. The app structure is according to the Firebase structure, it means each parent or child node contains a POJO in the app so they can fetch the content and use them very easily.

Describe any corner cases in the UX.

This app contains two activities MainActivity and DetailActivity. MainActivity is the launcher activity so on back press it pause the app. DetailActivity is to view the detailed information of the selected news, on back press it goes to the MainActivity.

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

This app use Picasso to handle the loading and caching of images.

Describe how you will implement Google Play Services.

This app use Firebase and it's dependent Google Play Services

Dependencies:

com.google.firebase:firebase-core:10.0.1 for Analytics
com.google.firebase:firebase-database:10.0.1 for realtime Databse
com.google.firebase:firebase-storage:10.0.1 for storage..

Next Steps: Required Tasks

First I will connect this app to Firebase.Then create the UI part according to the content I want to deliver to the user and then create POJOs of every parent, child and sub-child present in the Realtime database.
The main feature of this app is to show the latest news to the user so onStart(), this app will fetch all the latest news from the Firebase cloud and then store them into ArrayLists and set them to the adapter.It will also fetch the images related to the content. But Realtime database only contain the image name. The images will be fetched from the Firebase Storage ,if showimage option is checked otherwise it can will not fetch the image.
This app requires internet connection to run at the first time, after that the top 15 news will be saved to the user local database and will get updated as soon as the user connects the internet.

Task 1: Project Setup

This app will be developed in steps:

First configure the app and add the required library and dependencies.

- Configure Firebase to the App
- Paste the “google-services.json” file in the app folder
- Add the Picasso library
- This app requires Internet to run so add Internet-Permission in AndroidManifeast.xml

Add the required resources and layouts

- Add all the strings to the strings.xml file
- Add all the styles or color to the styles.xml and color.xml

Creating support packages

- Create model package for storing the POJOs of the news.
- Create utils package for any support file.

Task 2: Implement UI for Main Activity and Fragment

- Build UI for MainActivity
- Build UI for Fragment_Main
- Build the Transition for activity calling
- Add animation for icons

Task 3: Implement UI for DetailActivity

Create DetailActivity.java. This activity provide the description of the news clicked.

- Create layout
- Add icons and images
- Create elements

Task 4: Add SharedPreferences Data

Create sharedPrefrence to store the current settings

- Store the value for the showing of images in sharedPreferences.
- By default the app will show the images.

Task 5: Update the MainActivity

In this phase I update the code of mainActivity for better working on offline mode :

- Check the internet connection first.
- If internet not available and value on sharedPrefrence is exist, then show the stored news.
- If internet available fetch new feeds and update the value of sharedPrefrence.
- For the first time if the user dosent have internet conection , then show a Toast “internet is required for very first time”

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**”