

GitHub Username: [dexter-69](#)

News Flash

Description

News flash allows users to read headlines and search for news and save offline for offline reading. The App offers intuitive design for easy reading, so the you never get tired of reading headlines or news of your choice.

.
Have little time to read ? no problem!. Save the news offline for later reading. Read headlines or search for one and read it in less than a minute. Stay updated!

Intended User

This app is for everybody who is enthusiastic about reading news or love to stay updated with latest news headlines.

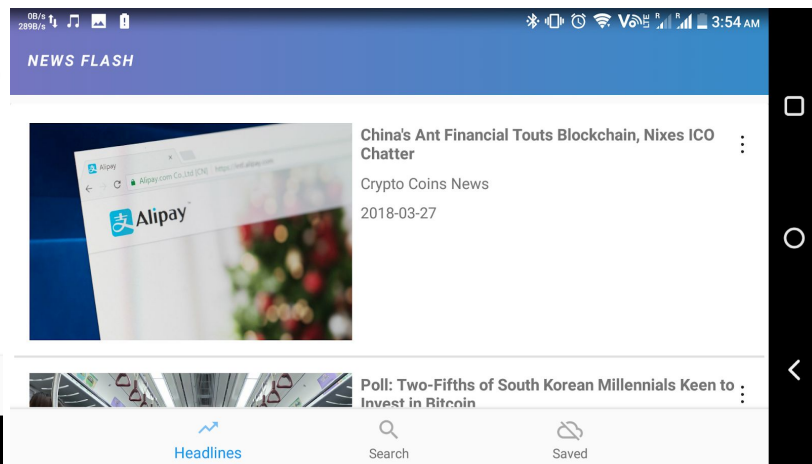
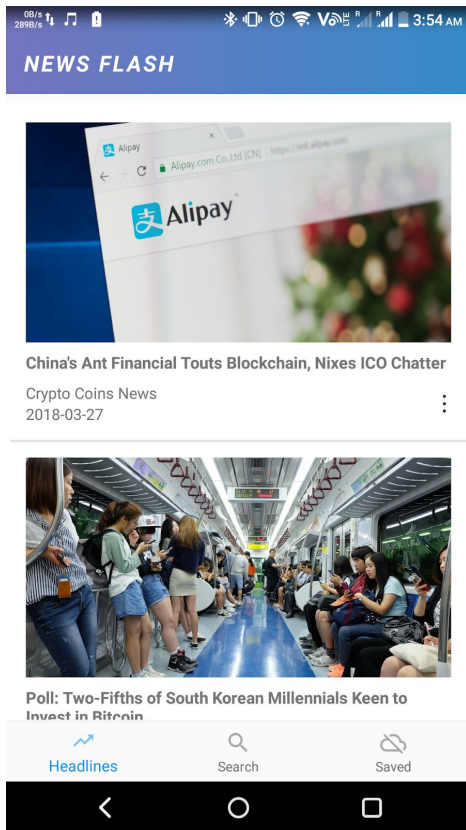
Features

- Headlines on top always!.
- Quick search .
- Intuitive design.
- Offline availability always!.
- News Headlines Widget.
- Sharing functionality.

User Interface Mocks

MainActivity (Main Fragment)

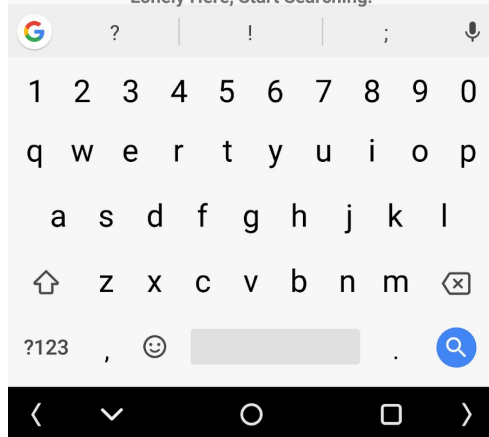
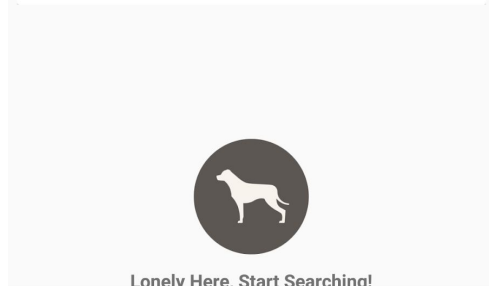
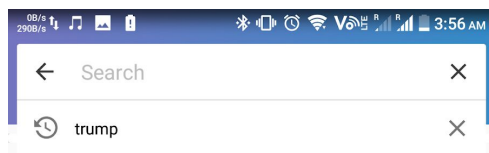
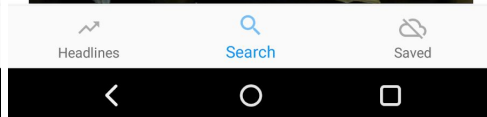
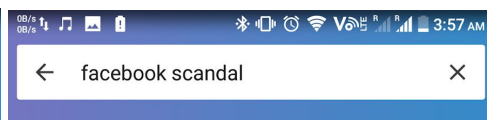
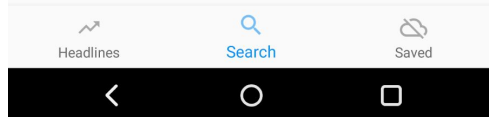
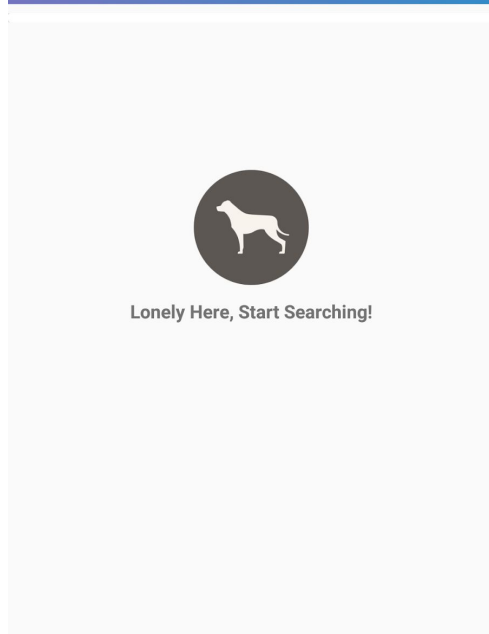
Portrait Mode



(Landscape Mode)

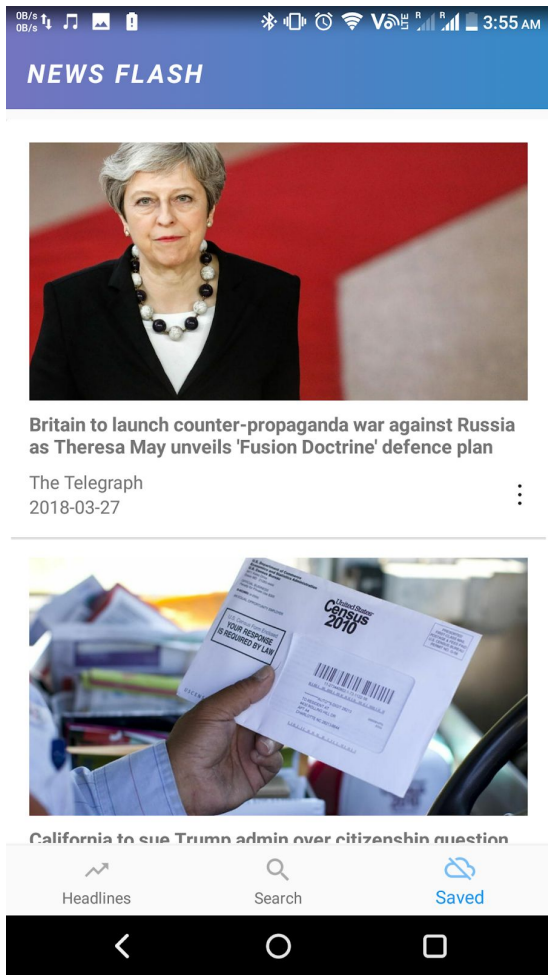
Main screen user will see on opening ap

Search Fragment



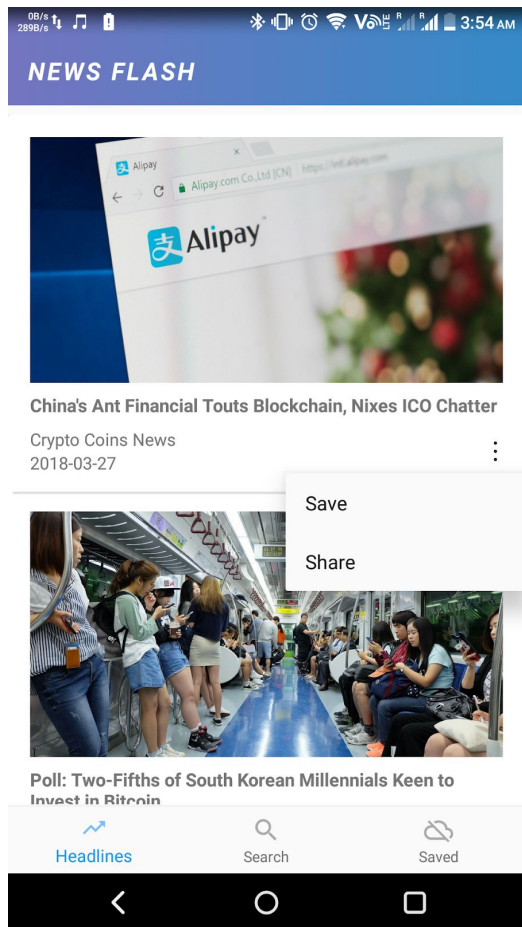
Search Fragment allows user to search(query) news

Offline Fragment



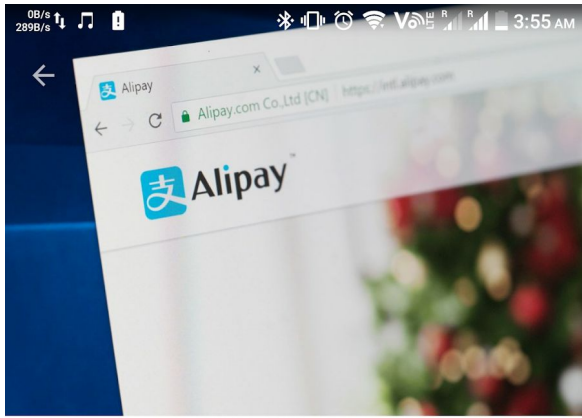
In this screen (fragment) user can view saved news articles without internet!

Options on News Card:



User can tap on 3 dots and instantly save or share news article

Detail Activity



China's Ant Financial Touts Blockchain, Nixes ICO Chatter

Join our community of 10 000 traders on Hacked.com for just \$39 per month. Jing Xiandong, Ant Financial's chief executive, called the blockchain "a cornerstone of trust" but dashed any hopes that the payments leader will pursue an initial coin offering. Xiand...

Source : Crypto Coins News

Published : 2018-03-27

Read Full Story : [Here](#)



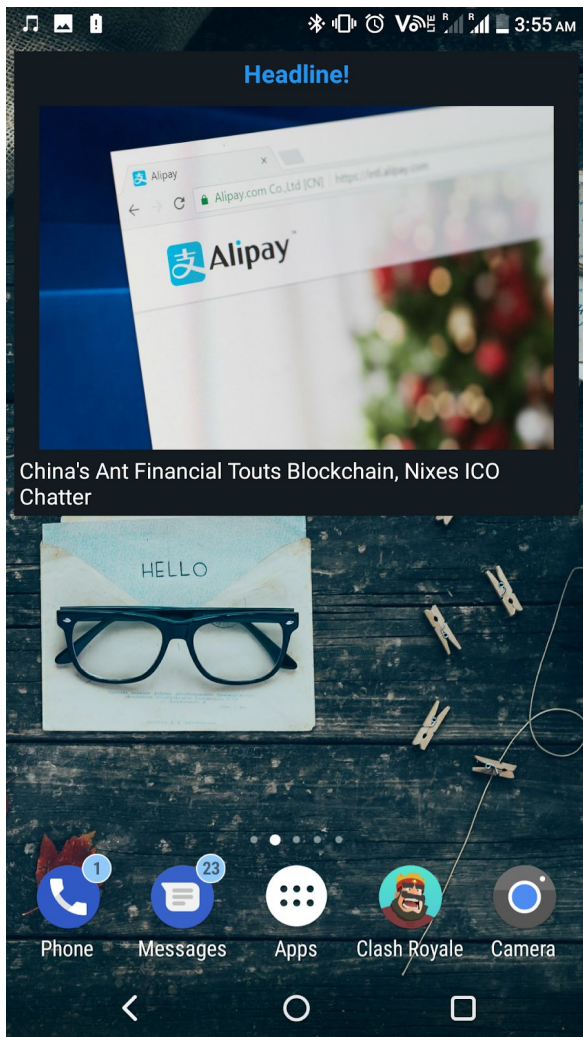
Detail Screen for News Article

Empty View:



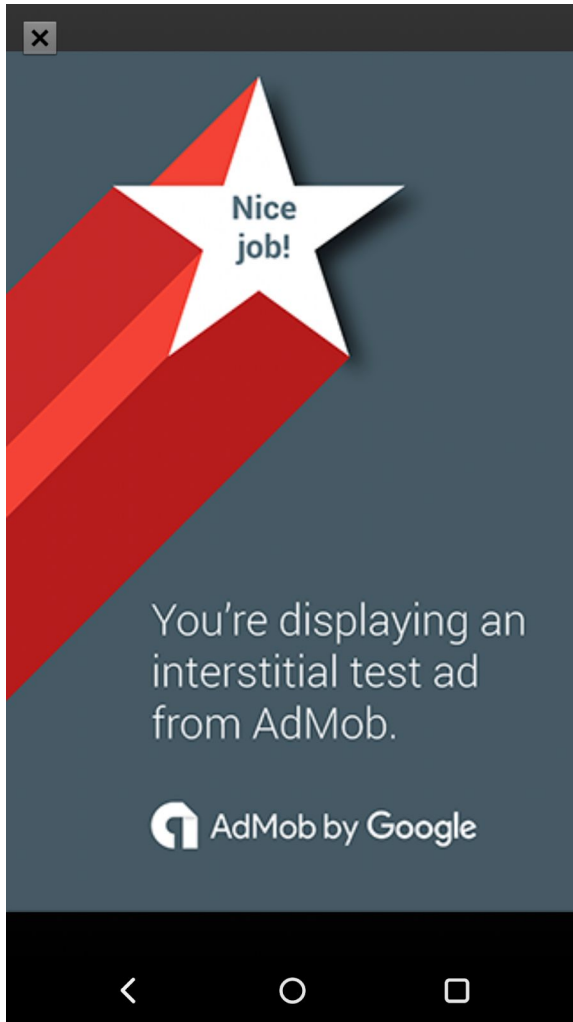
This is presented when user has network connectivity issues

Widget:



Widget for Top News Headline (Refresh Every 30 mins)

Admob:



Admob Dialog

Key Considerations

How will your app handle data persistence?

Data Persistence Using Content Provider (sqlite db).

Describe any edge or corner cases in the UX.

- When user rotates before data is loaded from network and populated in recycler view, it should not restart loading again.
- While loading user's network connection goes off.

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

- Butterknife : For Injecting View
- MaterialSearchbar : Library implementing material search bar just like google play app.
- Glide : For image loading and caching.
- Retrofit : Handling api calls.
- Gson : Converting objects into json for saving in database.
-

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

- Google Admob : I'll be implementing Google admobs as Interstitial Ad and will display on launch of app and after the data has been loaded.
- Google Analytics : I'll be implementing analytics in each fragment to see which fragment is most used by the user.

Next Steps: Required Tasks

Task 1: Project Setup

- Create a new project using Android Studio wizard.
- Add all the required dependencies and sync gradle.

Task 2: Implement UI for Each Activity and Fragment

- Mainactivity will contain three fragments:
 - Headlines fragment .
 - Search Fragment .
 - Offline Fragment.
- Detail Activity .
- Widget Layout.
- Create layout with qualifiers for different orientation and screen widths.

Task 3: Create Model Classes:

In this task, model classes for news article is created that extends from Parcelable interface.

Task 4: Implement RecyclerView

Once the base layout is created, create a recycler view that will be common to all mainactivity fragments.

- Create Custom recyclerview adapter that will bind news article data to views.
- Create ViewHolder for holding views that recyclerview will contain.
- Implement Click listener for recyclerview items.

Task 4: Create Content Provider

Create content provider (sqlite db) for data persistence.

- First create a contract.
- Create a class that extends from SQLiteOpenHelper and implement required methods.
- Then create content provider and implement all methods.
- Lastly add provider to manifest.

Task 5: Create Retrofit Interface

In this task, a retrofit interface will be created which will contain all the required api call methods. Then create a retrofit builder class that will return a retrofit object.

Task 6: Create Widget:

Create Widget Provider class.

- First create a class that extends from AppWidgetProvider.
- Override methods to update widget in some period.
- Create a retrofit instance and make a api call for data to be filled in widget.
- Add receiver to manifest that will listen to updates for this widget.
- Create remote views and bind data to widget

Task 6: Handle / Persist States:

Properly handle data according Android Guidelines on config changes and properly override methods to save / persist data.

- Use onSaveInstanceState to save state of various things like recyclerview state, scroll position, etc.
- Use onCreate / onRestoreInstanceState to restore above saved state.

Task 7: Proper feedback to user:

Errors like network connectivity or failed api calls provide user proper message to display on screen to notify user.

- Provide a retry button for retrying connection to network.
- Provide a offline button to load offline data on no internet connection.

Task 7: Asynchronous Operations:

Do heavy operations on background thread rather than on main thread to provide uninterrupted experience to user.

- Use Loaders to load data from database to views.
- Use AsyncTask to query and save data to database.