[Description](#_sm4ra97uwo11)

[Intended User](#_aws88pzfmqca)

[Features](#_zheq5430xrpq)

[User Interface Mocks](#_giquerrw6g46)

[Screen 1](#_a4jdupabry3k)

[Screen 2](#_dpcbbkx5yry)

[Key Considerations](#_gvcvmae8jn8u)

[How will your app handle data persistence?](#_v8my7nhtvz0m)

[Describe any corner cases in the UX.](#_gw69vjn1ico0)

[Describe any libraries you’ll be using and share your reasoning for including them.](#_6yqqubmw5bs)

[Describe how you will implement Google Play Services.](#_qrxg682nywe6)

[Next Steps: Required Tasks](#_v518bncmggeg)

[Task 1: Project Setup](#_8oe8zpk3qsmp)

[Task 2: Implement UI for Each Activity and Fragment](#_rzllsk6uqztx)

[Task 3: Your Next Task](#_fdmohs7hes)

[Task 4: Your Next Task](#_umfwsvmx7tpn)

[Task 5: Your Next Task](#_kjidlkq4xm3u)

**GitHub Username**: ruchiverma6

# App Name:

News Bot

# 

# Description

**Summary:**

This app is a chat bot app which helps user to browse news from different categories at one place.

User just have to chat with news bot in a way like I want news from sports category, or I want news from any other category like politics then News bot will present list of news from sports section or from any other asked category.

Solution: User don't need to browse news headlines for different news categories.

Now user can surf news for different news categories at one place through chat interaction.

This app provides news for following categories: Sports, Entertainment, General, Politics, Gaming, Music, Business, Science and nature, and Technology

# Intended User

Almost for anyone interested in news

# Features

* Chatbot interaction to browse news for different categories
* Saves NEWS information in database
* App provides a widget to provide relevant information to the user on the home screen.
* App uses an async task to save data in Database
* App integrates a third-party library: Picasso, Retrofit, Luis client
* App validates all input from servers and users. If data does not exist or is in the wrong format, the app logs this fact and does not crash.
* App keeps all strings in a strings.xml file and enables RTL layout switching on all layouts
* App includes support for accessibility. It includes content descriptions.
* App integrates two Google services: Google Analytics and AdMob
* Each service imported in the build.gradle is used in the app.
* App theme extends AppCompat.
* App uses an app bar and associated toolbars.
* App uses standard and simple transitions between activities.
* App stores data locally by implementing a ContentProvider
* App uses a Loader to move its data to its views.
* App builds from a clean repository checkout with no additional configuration.
* App builds and deploys using the installRelease Gradle task.
* App is equipped with a signing configuration, and the keystore and passwords are included in the repository. Keystore is referred to by a relative path.
* All app dependencies are managed by Gradle.

# 

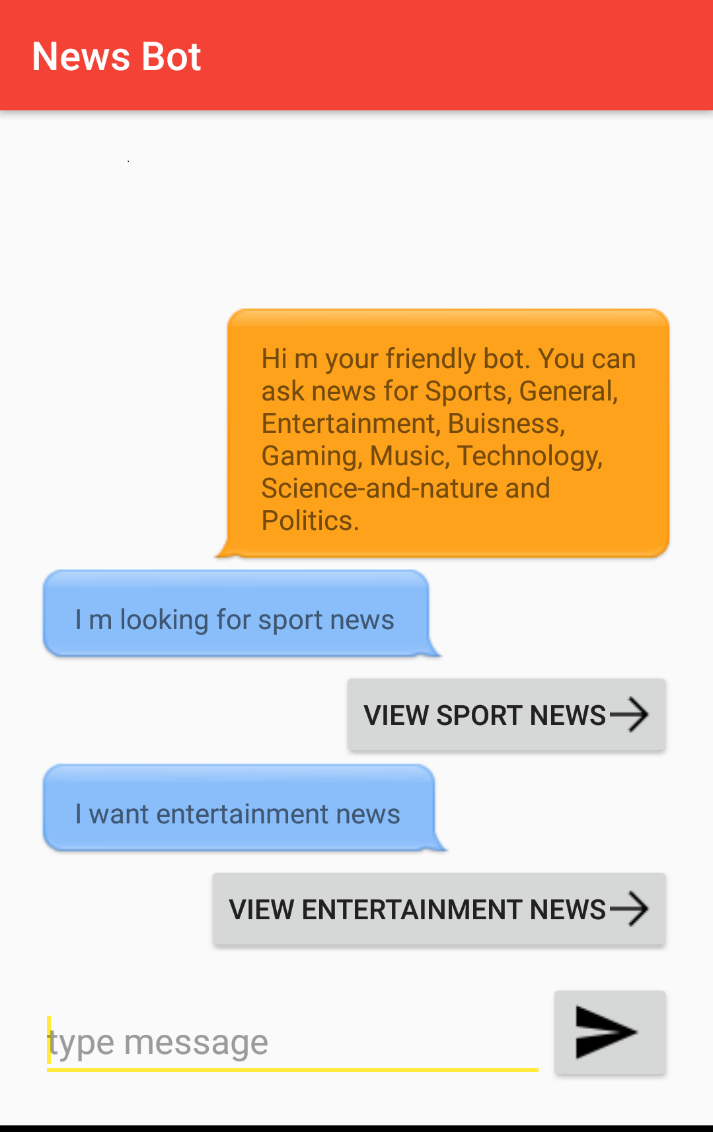
# User Interface Mocks

## Screen 1



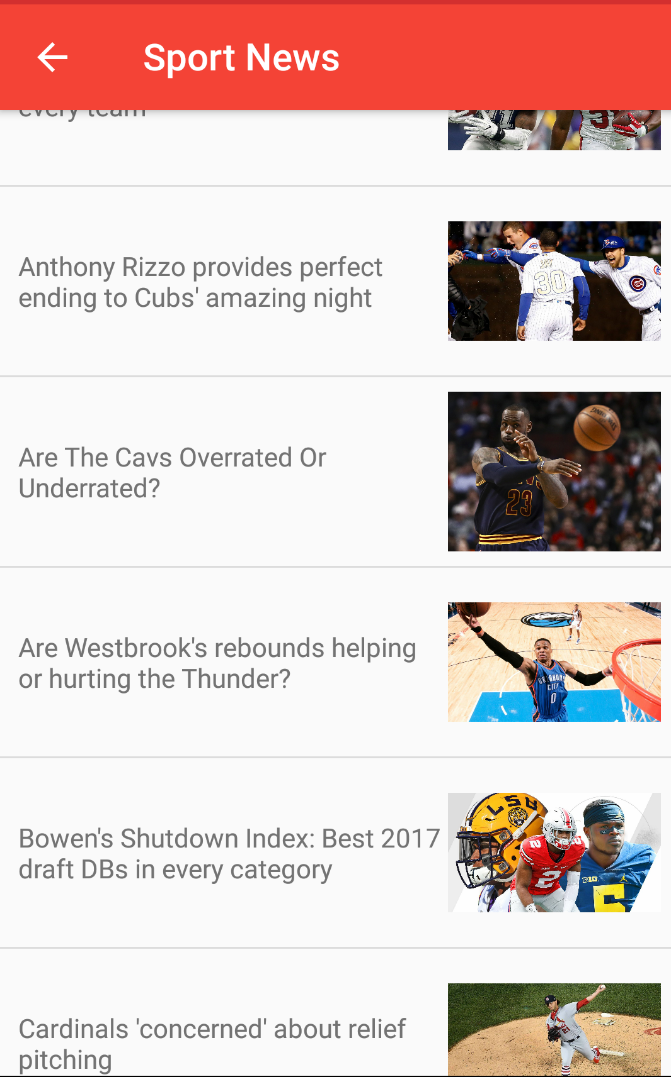
News Bot app loading screen

## Screen 2



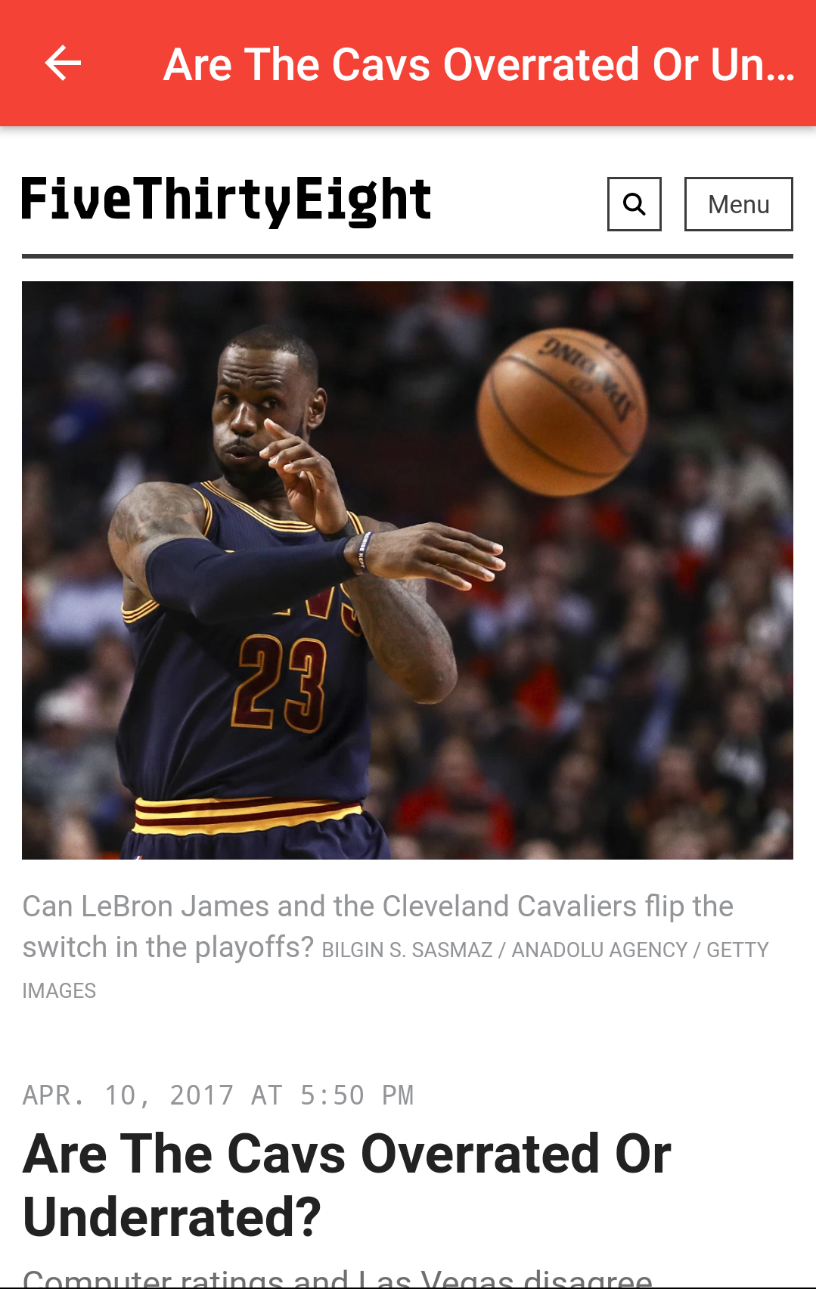
Chat Screen for asking news for different categories.

## Screen 3



News list screen for sports category.

## Screen 4



News detail screen.

## 

## Screen 5

## C:\Users\v-ruchd\AppData\Local\Microsoft\Windows\INetCache\Content.Word\NewsBot_widget.png

NewsBot widget UI.

# Key Considerations

### How will your app handle data persistence?

I will use content provider and sqlite db

### Describe any corner cases in the UX.

For example, how does the user return to a Now Playing screen in a media player if they hit the back button?

On first time launch of app, loading screen would be visible to user and after that Test Ad screen will be visible. But from second time of app launch, user will be landed to chat window screen instead of going to loading and test add screen.

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

1) Picasso library:

to handle the loading and caching of images.

2) LUIS Client library:

LUIS stands for Language Understanding Intelligent Service. This is designed to provide you with an easy way to create models, which allow applications to understand user command.

LUIS Client library is getting used for language understanding that provides intent classification and entity extraction for human interaction in chat bot.

3) Retrofit:

Networking library for making network calls

### 

### Describe how you will implement Google Play Services.

Google Analytics: I will implement google analytics to track how many times user landed to chat screen, News list screen and News detail screen and how many times user sent messages, and also will track how many times user saw news result.

ADMOB: I will implement admob to show test InterstitialAd

at first launch of app.

# 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

* Download android studio
* Import android project for News Bot app
* Create API key for accessing News open API at following site: https://newsapi.org/
* Configure following libraries:

Picasso

Retrofit

LuisClient

* Configure Google Analytics

## Task 2: Implement UI for Each Activity and Fragment

List the subtasks.

* Build UI for MainActivity
* Build UI for ChatActivity
* Build UI for NewsListActivity
* Build UI for NewsListFragment
* Build UI for NewsDetailActivity
* Build UI for NewsDetailFragment

## Task 3: Creating Layouts and Data Binding to UI

Subtasks:

* Create layouts for Loading screen, Chat screen, News list screen, News detail screen for phone and tablet both
* Bind data from database to UI using cursor loader

## 

## Task 4: Integrate Retrofit library

Subtasks:

* Create endpoint for fetching news API
* Create common client class for accessing retrofit instance
* Execute retrofit calls for different api

## Task 5: Implement Sqlite DB implementation

Subtasks:

* Write SQLiteOpenHelper class to manage sqlitedatabase instance
* Create Columns and tables
* Write code for insert/update/query of data from database

## Task 6: Implement Google Analytics

## Task 7: Implement AdMob to show test ad