|  |
| --- |
| **DESIGN DOCUMENT OF ANDROID APPLICATION FOR ALL TIME FAVOURITE NEWS (ATF) APPLICATION** |
| **Chandrasekar Jayaraj** |

Table of Contents

[Description 3](#_Toc525898250)

[Intended User 3](#_Toc525898251)

[Features 3](#_Toc525898252)

[User Interface Mocks 4](#_Toc525898253)

[Screen 1 – Login - Main 4](#_Toc525898254)

[Screen 2 – Login - Enter 4](#_Toc525898255)

[Screen 3 – Trending – News 5](#_Toc525898256)

[Screen 4 – Favorite – News 5](#_Toc525898257)

[Screen 5 – Content News 6](#_Toc525898258)

[Screen 6 – Ad Mob 6](#_Toc525898259)

[Key Considerations 8](#_Toc525898260)

[How will your app handle data persistence? 8](#_Toc525898261)

[Describe any edge or corner cases in the UX. 8](#_Toc525898262)

[Describe any libraries you’ll be using and share your reasoning for including them. 8](#_Toc525898263)

[Describe how you will implement Google Play Services or other external services. 8](#_Toc525898264)

[Next Steps: Required Tasks 9](#_Toc525898265)

[Task 1: Project Setup 9](#_Toc525898266)

[Task 2: Implement UI for Each Activity and Fragment 9](#_Toc525898267)

[Task 3: Implement Login Activity 9](#_Toc525898268)

[Task 4: Implement Trending News Activity 10](#_Toc525898269)

[Task 5: Implement Favorite News Activity 10](#_Toc525898270)

[Task 6: Implement Settings Activity 10](#_Toc525898271)

[Task 7: Implement Content News Activity 10](#_Toc525898272)

[Task 8: Implement SnackBar Activity 10](#_Toc525898273)

[Task 9: Implement News Category Activity 10](#_Toc525898274)

[Task 10: Implement News Widget Activity 10](#_Toc525898275)

[Task 11: Implement TermsOfService and PrivacyPolicyActivity Activity 10](#_Toc525898276)

[Task 12: Implement Copyright Activity 10](#_Toc525898277)

[Task 12: Implement Common Tasks for all the above Activity tasks 3 – 12 11](#_Toc525898278)

[Task 12: Implement manifest and other services like google play and firebase 11](#_Toc525898279)

[Task 13: Implement error handlings, test cases, build variants and release tasks 11](#_Toc525898280)

[Task 14: Deploy and test 11](#_Toc525898281)

**GitHub Username**: jpchandru

ATF NEWS

# **Description**

ATF News is a news app that will provide various categories of news like trending, politics, tech, entertainment, sports, fashion and world news. App will have a provision for the user to register, bookmark their news card and share their news card. App provides a world news widget of a list of countries. User registration will store their news preferences instead of device id which makes the user to install the app in any devices and still retrieve their preferences.

# **Intended User**

Intended user for this app will focus on all passionate news readers.

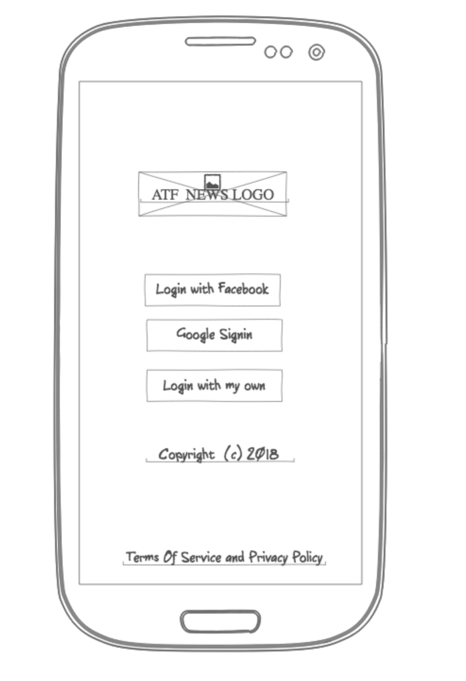
# **Features**

Main features:

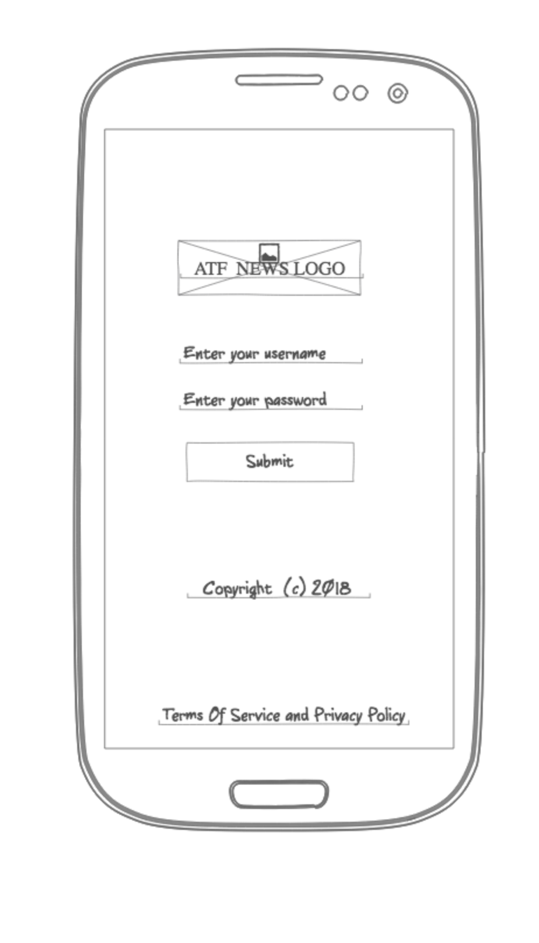
* User registration through Google signin or Facebook signin or self registration.
* Display news based on user’s preference over either current location or international.
* Bookmark news card into their favorites section will allow readers to read at a later time.
* Share news card.
* Display advertisements.
* Send alerts/notifications about breaking news.
* Widget of all country names to read their selected country news.
* Settings menu to modify their email, location based news and alert frequencies.

# **User Interface Mocks**

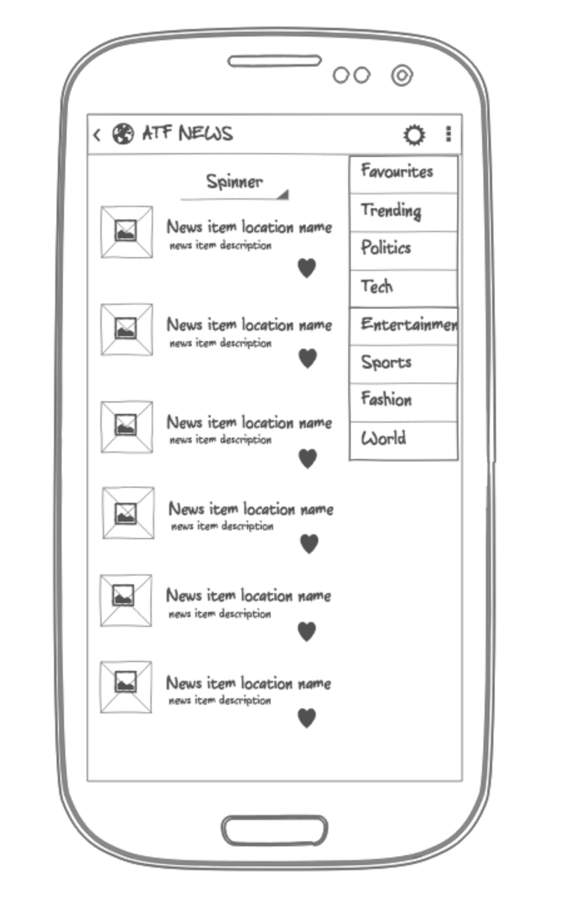
## Screen 1 – Login - Main



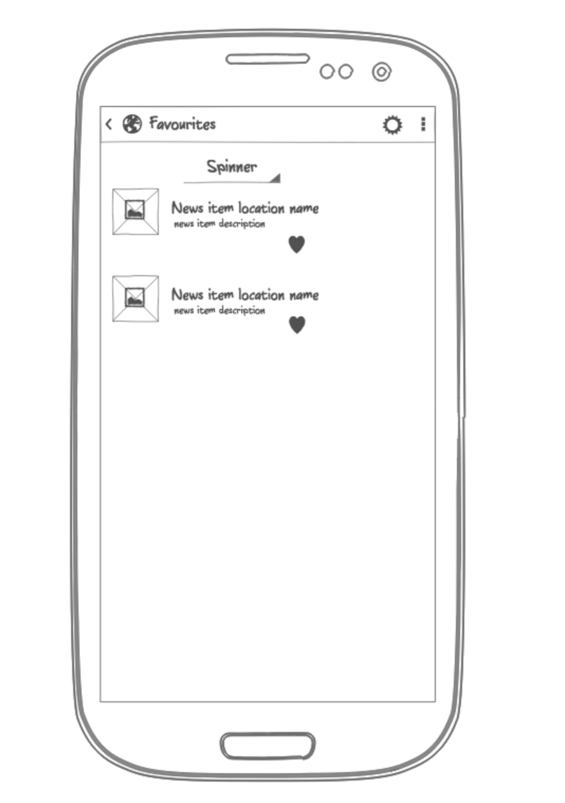
## Screen 2 – Login - Enter



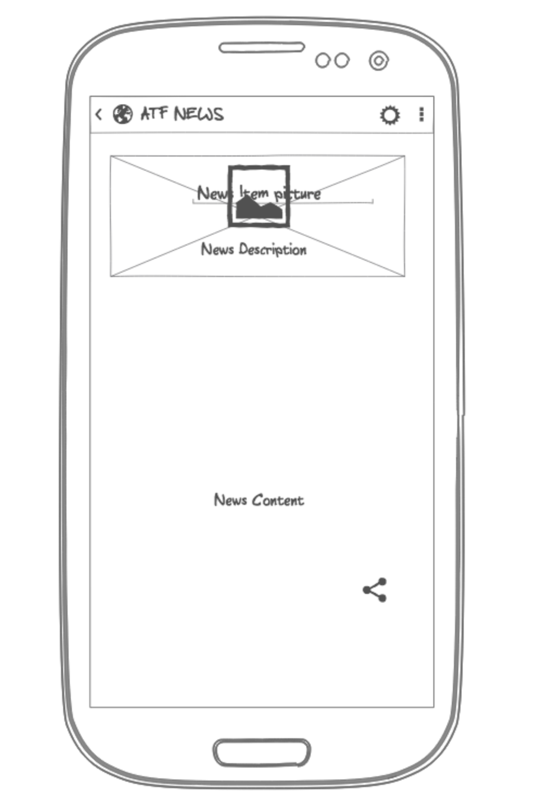
## Screen 3 – Trending – News



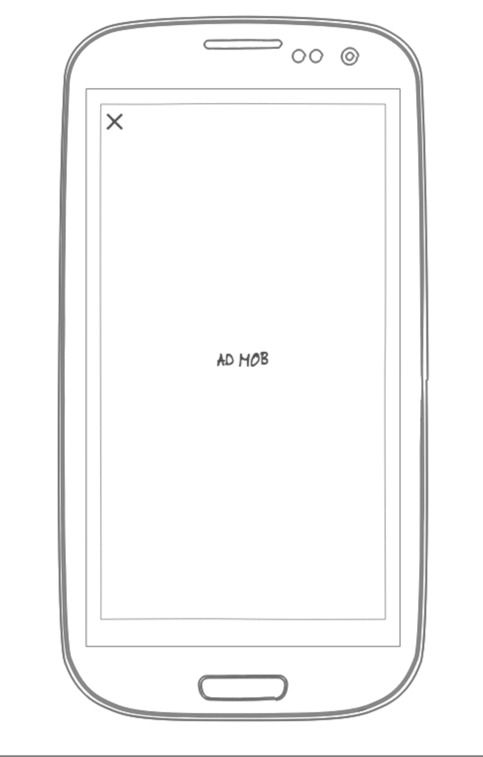
## Screen 4 – Favorite – News



## Screen 5 – Content News



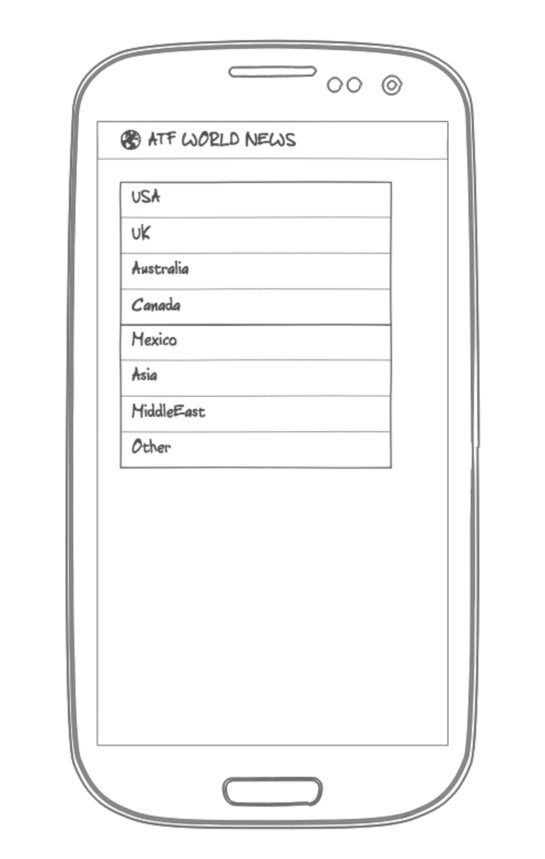
## Screen 6 – Ad Mob



Search 7 – Settings



Search 8 – Widget containing country list



# **Key Considerations**

### How will your app handle data persistence?

Data persistence will be handled by Rooms database API and use its LiveData mechanism as well. Databases will be loaded and handled by a running GCE as well.

### Describe any edge or corner cases in the UX.

* Whenever user faces an internet connectivity issue, then the app will display a snack bar that will notify about loss of internet and it will have a retry link.
* During loss of internet, user will still be able to browse the content as offline as the app will be using Volley API caching strategies.
* During the loss of internet, all the items inside Settings menu will be disabled like Email field, location based news selection field and alert frequency field.
* During loss of internet and user clicks on any item that are not previously cached then the user will be notified with a toast message and also a snack about No Connectivity message.
* During loss of internet and user tries to login then app will show a snack bar with Internet connectivity issue and a Retry link
* If the news content doesn’t have an image to load then a No image available placeholder will be applied to the image block.
* If the news content text has more than 1000 words then it will be truncated during initial page load to avoid page crash.

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

* Picasso will be used to load images – Version: 2.71828
* Volley will be used to handle the responses from the webservice call in-order to retry if failure and cache responses as well for offline loading – Version: 1.0.0
* ButterKnife will be used for data binding resources – Version 8.8.1
* Room database persistence library to store user information’s and preferences – Version: 1.0.0
* GSON to parse JSON responses – Version: 2.8.5
* Design support library for material design – Version: 27.1.1
* Google Play services for ad: Version: 11.8.0
* App Engine plugin to run backend services via google cloud environment – Version: 1.3.3
* Espresso for unit testing – Version: 3.0.2

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

* Google play services will be used to load advertisements and location services.
* Firebase services will be used for google signin, facebook signin and sending breaking news alerts.
* Google Cloud Environment service will be used to expose a webservice url in-order to retrieve user information’s from Room database.

# **Next Steps: Required Tasks**

## Task 1: Project Setup

* App will be written in Java 1.7
* App will be written using Android Studio 3.0.1, Gradle 3.1.4 and all libraries will use stable versions. Stable version of libraries has been mentioned in this document under library section.
* Create a multimodule project structure that contains modules like app, backend, androidlibrary and a java library.
* Create project based gradle file and gradle files for each module.
* Implement the dependencies in each gradle files that contains free and paid versions in it.
* Implement release tasks inside gradle file.

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

* Create layout file for main login screen.
* Create layout file for login entry screen.
* Create layout for AppBar.
* Create layout for menu that contains all news categories.
* Create layout for Trending News display screen.
* Create layout for Favorite News display screen.
* Create layout for Settings screen.
* Create layout for AdMob Screen.
* Create layout for Content-News display screen with FAB in it.
* Create layout for snackbar.
* Create layout for widget display.
* Create layout for About.
* Create layout for Contact us.
* App keeps all strings in a strings.xml and enables RTL layout switching on all layouts.
* App will support accessibility by having content descriptions.
* App will support accessibility by implementing navigation using D-Pad if time permits.

## Task 3: Implement Login Activity

* Implement LoginActivity to login into app using Firebase service for FB, google authentications or self signin operation and send email after successful registration.

## Task 4: Implement Trending News Activity

* Implement TrendingNewsActivity to retrieve trending news via a webservice call using Volley.

## Task 5: Implement Favorite News Activity

* Implement FavoriteNewsActivity to retrieve favorite news from user favorite news database.

## Task 6: Implement Settings Activity

* Implement SettingsActivity to retrieve user settings from user info database running in GCE. Handle no internet connectivity issue to disable list items in settings screen.

## Task 7: Implement Content News Activity

* Implement ContentNewsActivity to retrieve news content.

## Task 8: Implement SnackBar Activity

* Implement SnackBarActivity for internet connectivity retry actions.

## Task 9: Implement News Category Activity

* Implement NewsCategoryActivity to retrieve news based on each menu item selection.

## Task 10: Implement News Widget Activity

* Implement NewsWidgetActivity to display all country names and clicking on each item will display its respective country news in a screen.

## Task 11: Implement TermsOfService and PrivacyPolicyActivity Activity

* Implement TermsOfServiceAndPrivacyPolicyActivity to retrieve the information from database.

## Task 12: Implement Copyright Activity

* Implement CopyrightActivity to retrieve the information about copyright from database.

## Task 12: Implement Common Tasks for all the above Activity tasks 3 – 12

* Implement gradle dependencies for Javalibrary
* Create JavaLibrary to implement all Entity and DAO classes.
* Create Async tasks inside App to retrieve information’s from webservices. This will include remotely hosted news api webservices and webservices exposed for local db value retrieval.
* Implement gradle dependencies for androidlibrary
* Create Androidlibrary to implement all display operations to the layout.
* Implement gradle dependencies for backend web service running in GCE.
* Implement backend service to expose a webservice url to retrieve information’s

## Task 12: Implement manifest and other services like google play and firebase

* Create entries into AndroidManifest.xml
* Implement Google Play Services for AdMob
* Implement Google Play Services for FAB
* Implement Google Play Services for location services
* Implement Alerts via Firebase services (If time permits)

## Task 13: Implement error handlings, test cases, build variants and release tasks

* Handle Error Cases
* Create Build Variants
* Create keys
* Create signing and release tasks
* Create unit and instrument test cases.

## Task 14: Deploy and test

* Deploy and test the application in phone and tablet