GitHub Username: tavishjain

ComicBuzz

Description

A lifesaver app for Comic Lovers !!!!!
ComicBuzz provides you all the information about your favourite comics, in whatever way you
want - Issue wise, Volume wise or any other way you want !!
It gives you the instances of all the characters in your favourite comics, the date they were first
seen into the comic, their date of exiting the comic, their real life creators, and much much more.

All thanks to the life saver Comic Vine API……………………...

Technical Descriptions

The app uses Java as the main programming language.
Strings, images, dimensions etc. all would be placed properly inside their respective folders.

App uses AsyncTask to sync the data once a day or as per the user

Intended User

People who have their heart based within comic books or keep a keen interest in reading
comics are the intended audience

Moreover, every single user would find a whole lot of things to explore in this vast collection of
comic books that one wouldn't finish up completely with the app !!!!!!!!!!!!!

Features

It would have a Main screen Activity that would show the user a list of Comic books available
according to their issues, or the volumes, as per the user
The user could select any of the comic book to see its details
The details include general info of the comic book, the description, list of characters, date
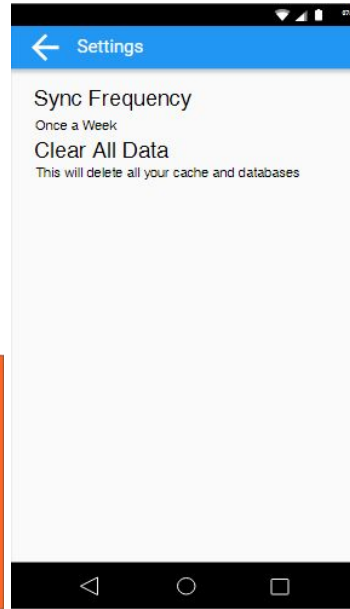released, date finished etc
Users can also see the information of each individual character, like when was this character
born, and his real life name and details , if not fictional , elsewise, its creators.
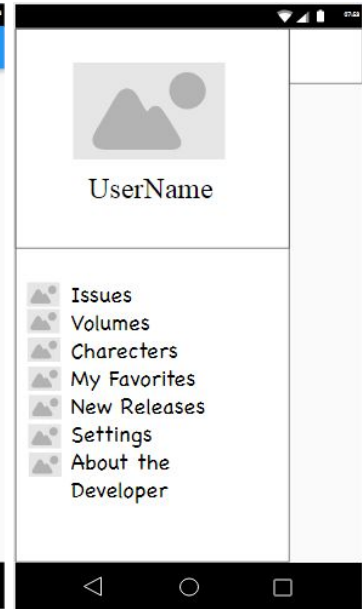User can search for his favorite series or volume.
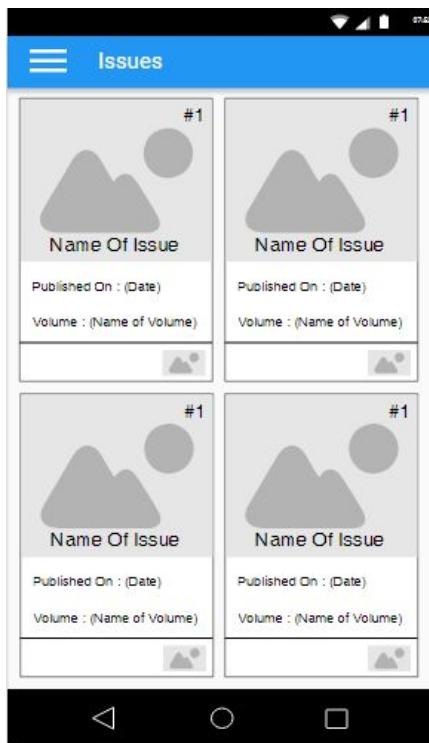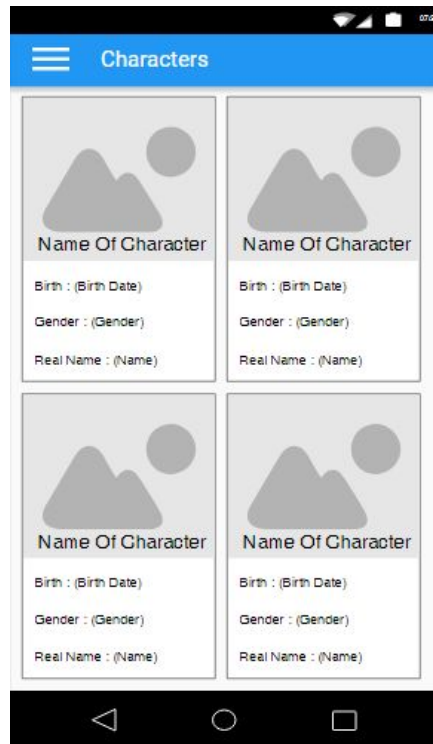
User Interface Mocks

Settings

Sync Frequency
Once a Week
Clear All Data
This will delete all your cache and databases

UserName

- Issues
- Volumes
- Charecters
- My Favorites
- New Releases
- Settings
- About the Developer

New Releases

Volume	Issue Name
Name	Name
Name	Name
Name	Name
Name	Name
Name	Name
Name	Name
Name	Name

Widget

Settings Screen

Navigation Bar

Issues

#1
Name Of Issue
Published On : (Date)
Volume : (Name of Volume)

#1
Name Of Issue
Published On : (Date)
Volume : (Name of Volume)

#1
Name Of Issue
Published On : (Date)
Volume : (Name of Volume)

#1
Name Of Issue
Published On : (Date)
Volume : (Name of Volume)

Issues Activity

Volumes

Name Of Volume
Count Of Issues
Publisher : (Name of Publish
Start Year : (Year)

Name Of Volume
Count Of Issues
Publisher : (Name Of Publish
Start Year : (Year)

Name Of Volume
Count Of Issues
Publisher : (Name Of Publish
Start Year : (Year)

Name Of Volume
Count Of Issues
Publisher : (Name Of Publish
Start Year : (Year)

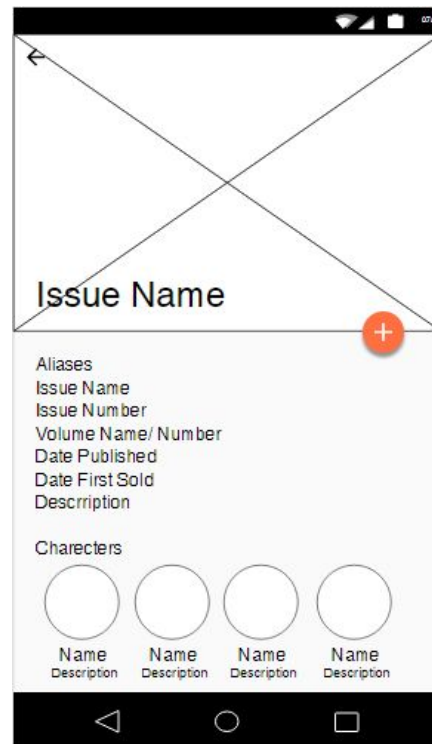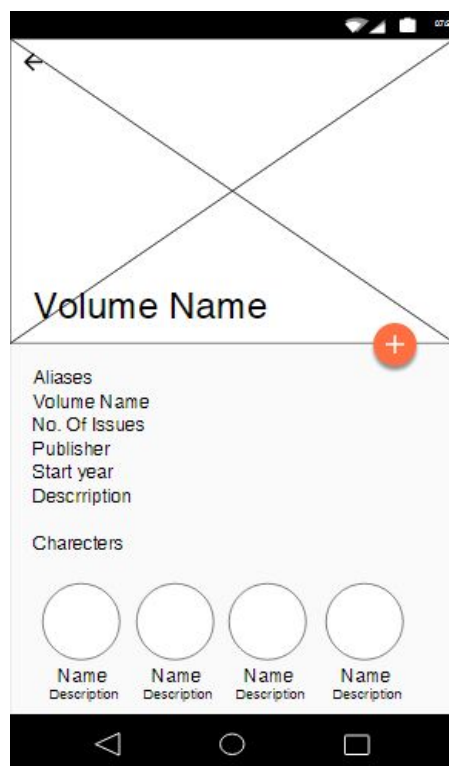**Volume Activity**

Character Activity

Issue Details

Volume Details

Character Details

Key Considerations

How will your app handle data persistence?

My app will use Room along with LiveData and ViewModel along with SharedPreference to save information.

Describe any edge or corner cases in the UX.

The app would fail to load the map in case of internet connectivity not available and would also show a snackbar containing the same info.

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

Picasso – Image loading and Caching - `v2.71828`
Butterknife – View Binding library – `v8.8.1`
Android Studio – `v3.1`
Gradle – `v4.9`
Retrofit - `v2.0`
Firebase - `v16.0.1`

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

I would be using Firebase to take a track of the users account, on what basis are they enrolling or logging into the app, to see the number of users using my application.
Also I plan to add an AdMob activity as this app would be available for free currently.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Creating a new Project in Android Studio and then configuring all the dependencies and adding the permissions in Manifest.

Task 2 : Adding all Model Classes and set up Retrofit

Next , I will add all the Model Classes(or the POJO classes) and set up the interface for the Retrofit Client so that I can make the Network requests to the API.

Task 3 : Designing All the UI and layouts

Next I would design the layouts , including the layout for navigation drawer , the involved activities or fragments and the widget

Task 4 : UI tests

Next I would add the UI tests and test my UI using the espresso library

Task 5 : Create the Settings Screen

The settings screen would include certain user specific settings like notification frequency , setting a theme and so on

Task 6 : Implement Google Play Services

Next I would implement the Google play services like Firebase for user authentication and google admob for showing ads.

Task 7 : Create a widget

Towards the end , I would create a widget that would show user the latest releases in the market

Task 8 : Including Accessibility support

I would constantly try to add accessibility support like content descriptions to images , RTL layouts etc.