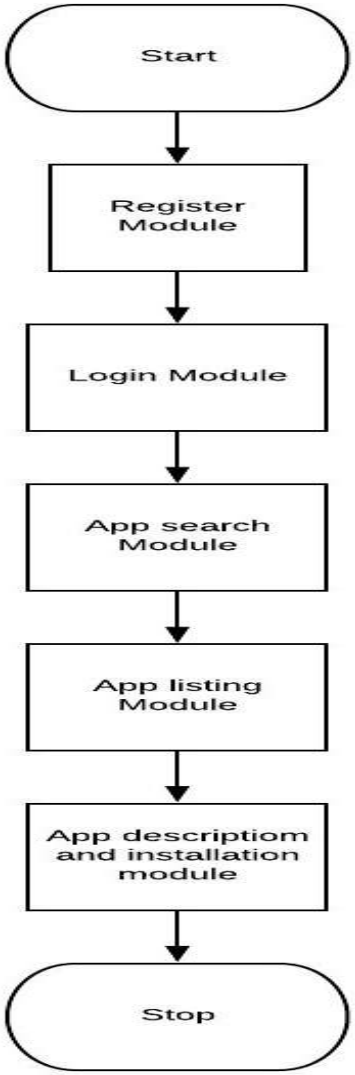
Abstract:

- The usage of smart phones has been increasing exponentially. This has resulted in a wide range of applications in the google play store.
- We see a lot of applications with similar names and features the naïve users would get into confusion by installing the applications which don't meet their requirements.
- Eventually results in wasting lot of time and cost of the user.
- To this problem we come up with a solution and that is :

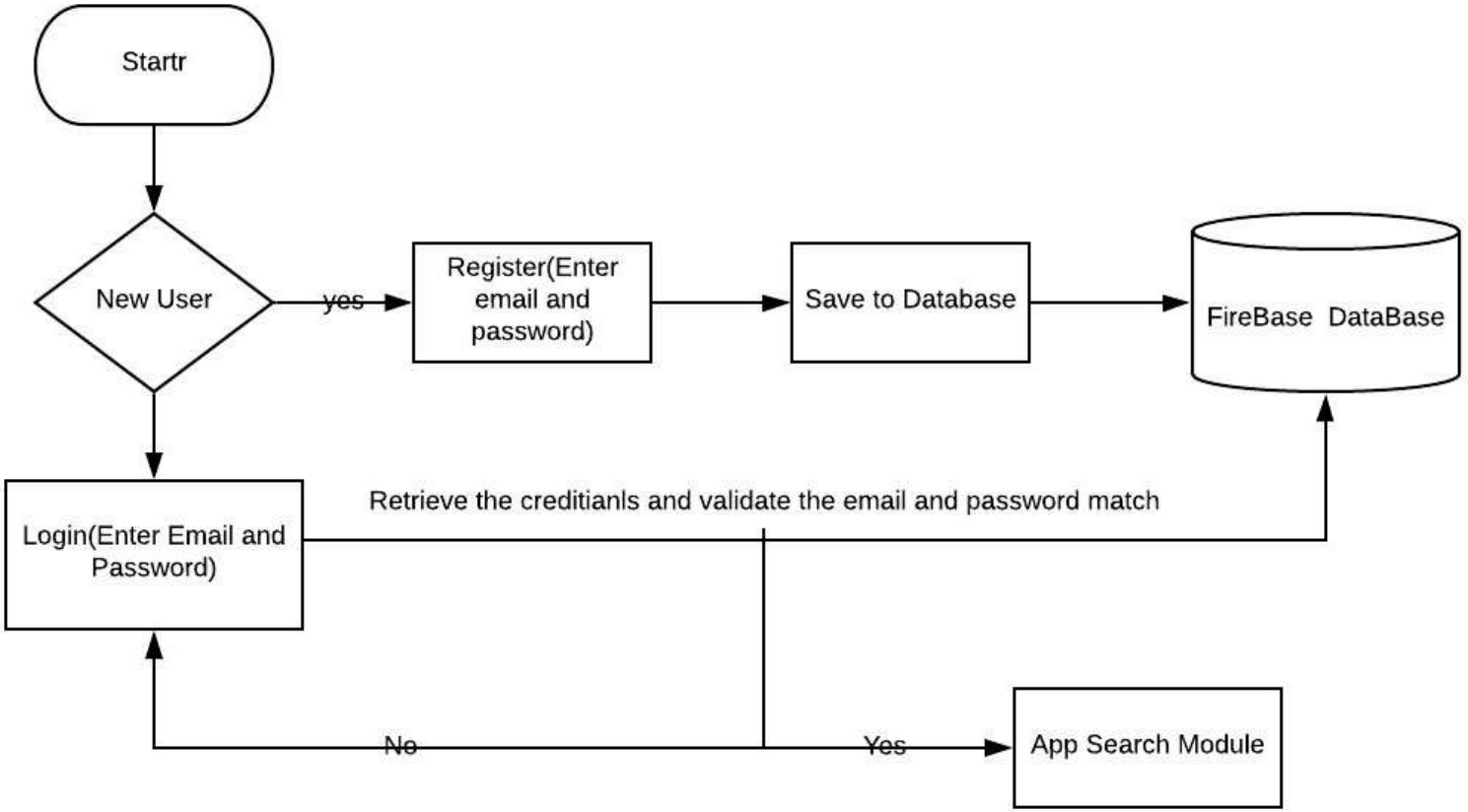
"SeWiCo" -SEarch the best WIth COnvenience

- The project helps to predict dynamically a better application that could serve the user requirements based on the past reviews, ratings, resource consumption, genres, content rating and support
- An android application that Displays top 5 applications for a given key search that reduces hassle for the user.
- If the user is already using a app in specific category which does not meet his requirement it also displays the top 5 similar apps that meets the user requirements.
- Provides a link to install the application of users choice.

SeWiCo Application:



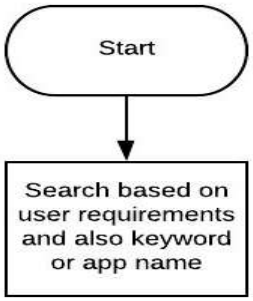
Register and Login Module:



APP SEARCH MODULE

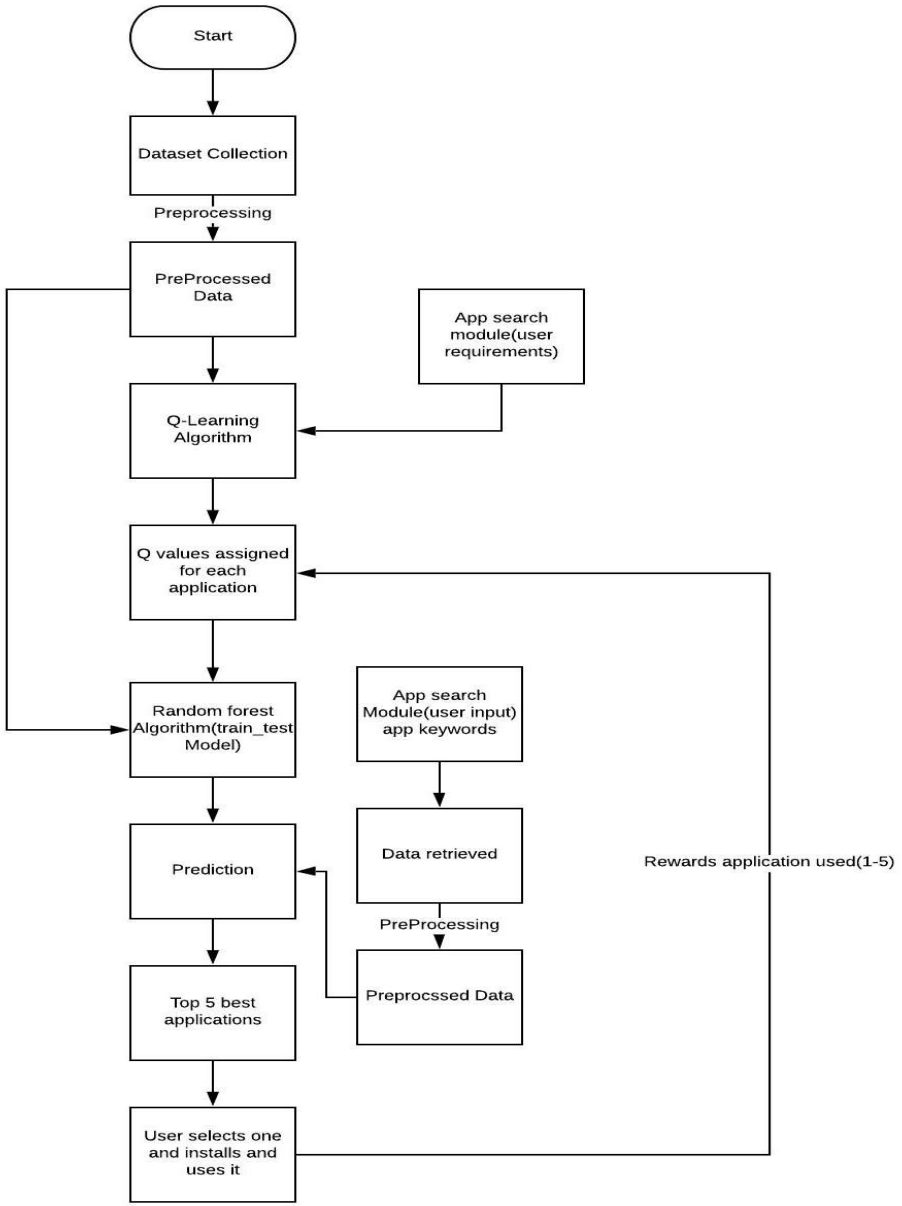
We have implemented this module using a python library named play_scrapper.
If the user knows the app name or keyword related to app, then the user enters keyword to search and also his requirements and intrests.
If the user finds out app required the app name can be selected then all the apps related to the name selected will be retrieved otherwise all the apps of each suggestion are also retrieved and all the details of apps retrieved and the features of apps are stored in datastorage.
The features include:

1)App title
2)Category
3)Content rating
4)Current version
5)Editors choice
6)Free
7)In App Purchases(iap)
8)Iap range
9)Installs
10)Price
11)Required android version
12)number of reviews
13)number of ratings
14)rating score
15)Size



Process applied in predicting the top 5 best applications:

In the process of processing input and displaying top 5 best applications futher process is applied:



Dataset Collection:

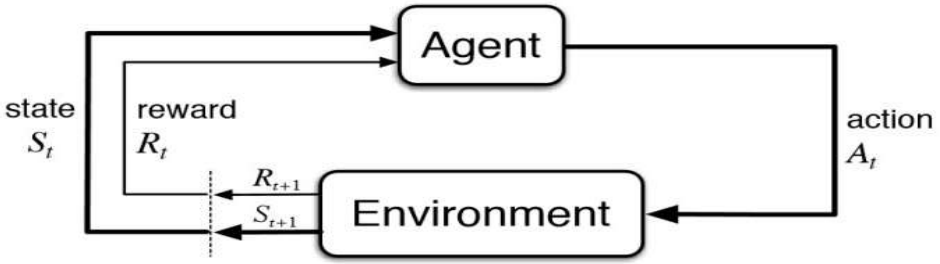
- Dataset is collected from playstore using concept “Web Scrapping” implemented in python.
- 10k+ apps of 59 categories are scrapped from playstore with 26 features:

App title,Category,Content rating,Current version,Editors choice,Free,In App Purchases(iap),Iap range,Installs,Price,Required android version,number of reviews,number of ratings,rating score,Size,App id, Description,Developer,Screenshots,Last updated,url,vedio

PREPROCESSING:

- The dataset obtained is not in required form to apply algorithm so the data is cleaned and preprocessed and all the features are converted to required form

Q-Learning Algorithm:



Equation1 : $$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_{t+1} + \lambda \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right]$$

Random Forest Algorithm:

- Random forest algorithm is applied for the dataset after assigning q values for each application
- The model is trained and tested and a model is built.
- The model takes the data from App search module which is obtained and predicts q values for each app and assigns it.
- The top 5 apps with maximum q values are predicted as output.

Learning process of recommendation system:

- The user selects the app from the 5 and installs it and uses it the user gives the reward for the app(1/0)
- The reward is updated to the q values and when next time the app is predicted gives better prediction then present output
- It is a continuous learning process.

APP Listing and installation module:

- The top 5 best apps are predicted as output based on user inputs and requirements.
- When a user selects an application link is provided which redirects him to install the application and also summary of functionalites of application is shown.

Reward module:

- Functionality is provided for users to give reward to application after using it only

Future Enhancements:

- Providing functionality for the user to search in their own language
- Developing a software for Appstore and also windows store.