



# LIBRARY APP

Andrea Lombardo 1893440

# TABLE OF CONTENTS

1. INTRODUCTION

2. AUTHENTICATION

3. FRAGMENTS (DETAIL -SETTING- PROFILE)

4. SERVER

5. GOOGLE BOOKS REST API

6. FUTURE WORKS

# 1-INTRODUCTION

## Requirements of the app:



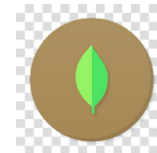
USER NEEDS TO CREATE ITS OWN VIRTUAL LIBRARY SAVING BOOKS READ AND THOSE WHICH STILL HAVE NOT FINISHED.



Create a personal node server able to maintain information of the user referred to the collection of books and information of each of them



USER NEEDS TO SAVE WHICH IS THE BEST BOOK



The node server implement REST CRUD operation that store information into a non-realtional dbms MONGODB



USER NEED TO SAVE SENTENCE OF A BOOK WHICH LOVES.



Firebase

The authentication of the user is done through another external dbms called Firebase that let to crate a user based on classic SIGN UP form or with Google Api service.



**Firebase**

# AUTHENTICATION

# AUTHENTICATION


► The authentication is done through Firebase. The sign in is done through

► 1) Email and password

► 2) Google

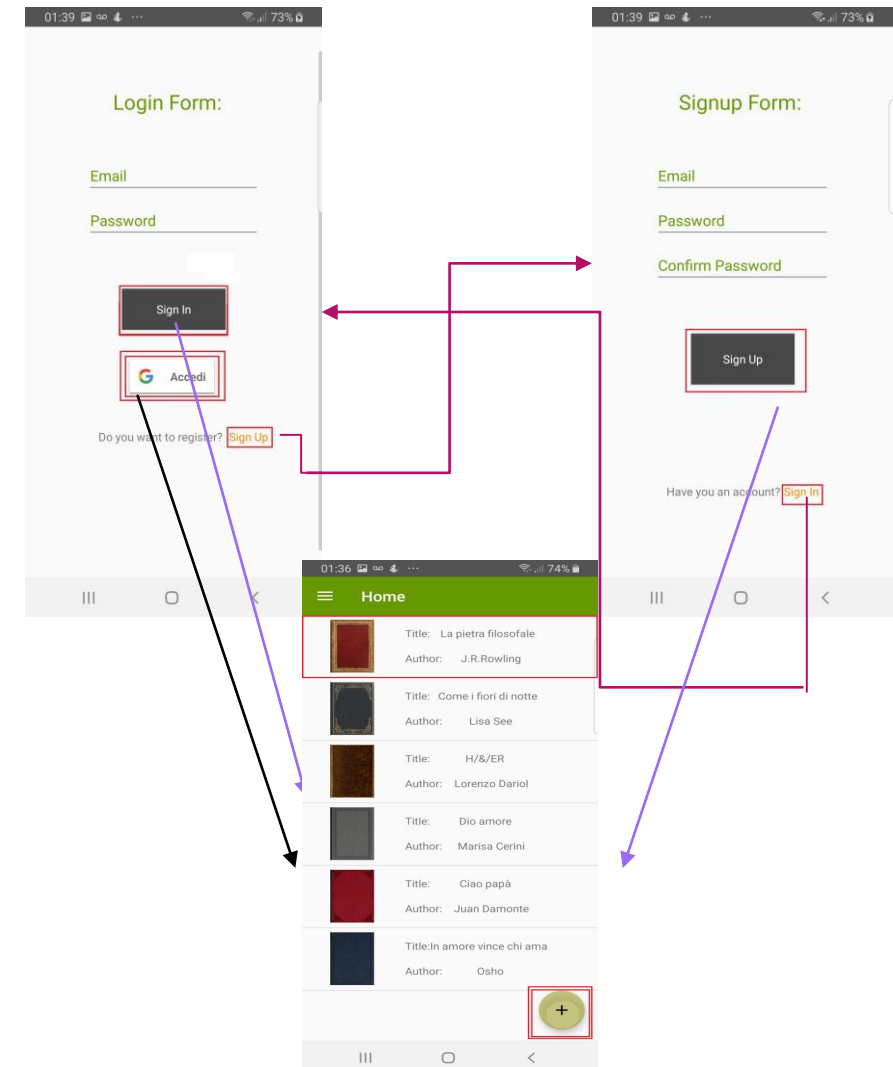
The orange Textfields are hyperlinks which lets us to jump from the Login activity to Sign up activity or viceversa.

After complete the form there's the possibility to create a user into Firebase or there's the possibility to verify that the email and password are correct for entering inside to the app using the buttons (Sign up or Sign in) .

Another important issue that let us to perform better the system is the possibility to enter into the system through the Google access click on 

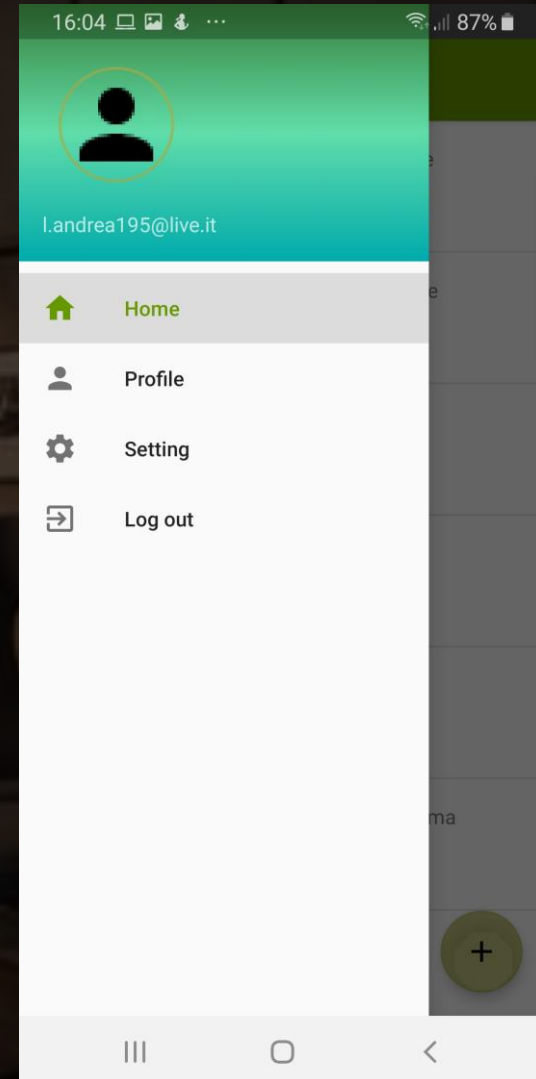


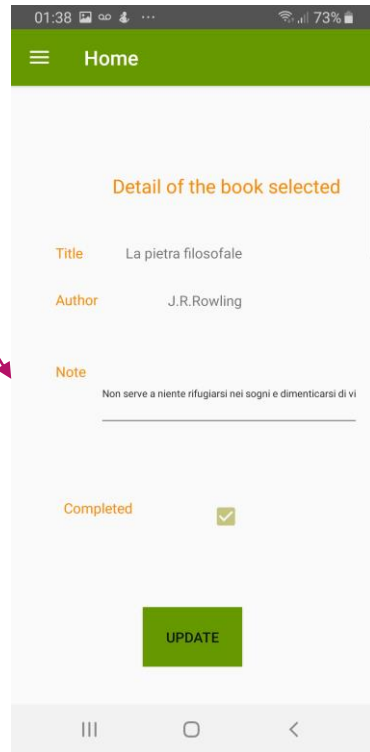
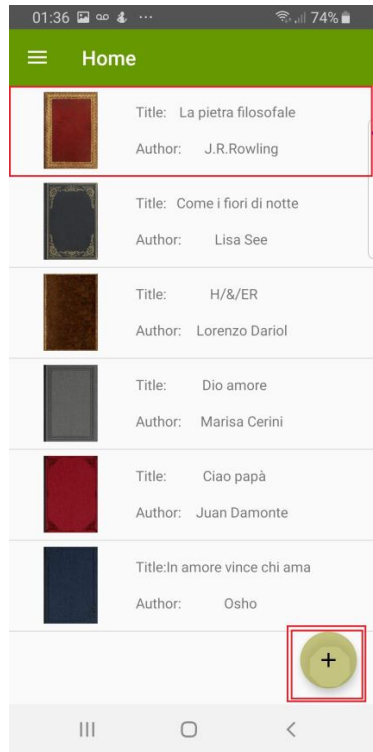
Firebase



# NAVIGATION DRAWER

- ▶ With a navigation drawer with fragments I 've the possibility to see the email where I'm logged in, then the Home, the settings, the profile and the possibility to Logout and so going to the Login Form.
- ▶ The Home lets to enter to the fragment designed with the ListView of books
- ▶ The Profile lets to enter to a fragment characterized only by a sequence of information of the user logged in.
- ▶ The setting let us to set some preference of the user.





# DETAIL OF A BOOK SELECTED

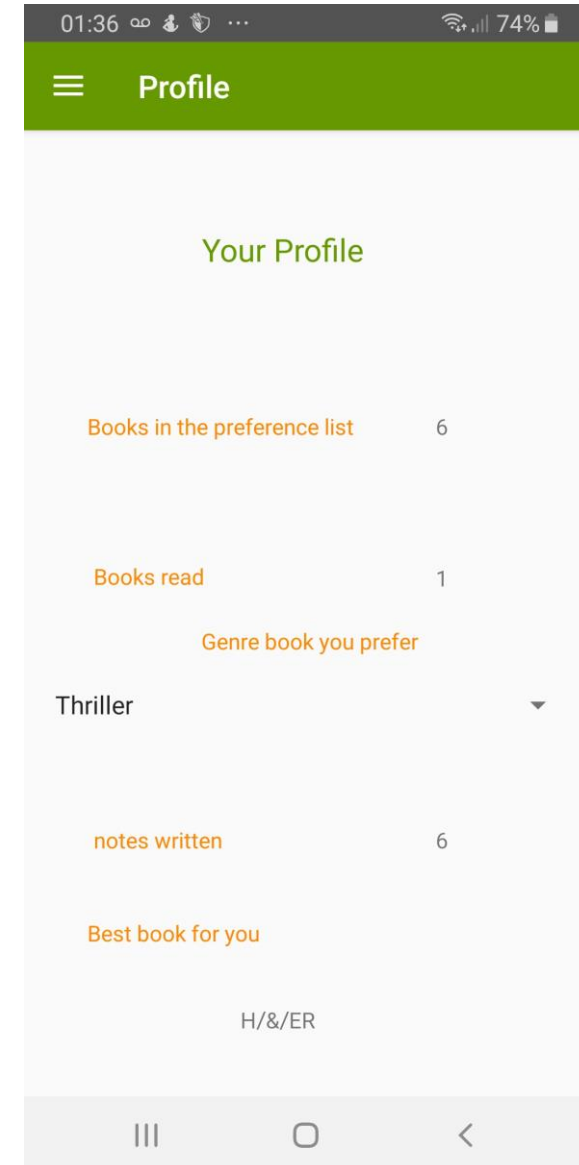
► If we click on the book there's the possibility to save more information about the book selected like the sentence we prefer and fill the checkbox if we've completed otherwise we leave it as empty.

► Then we can modify the information with the UPDATE button.

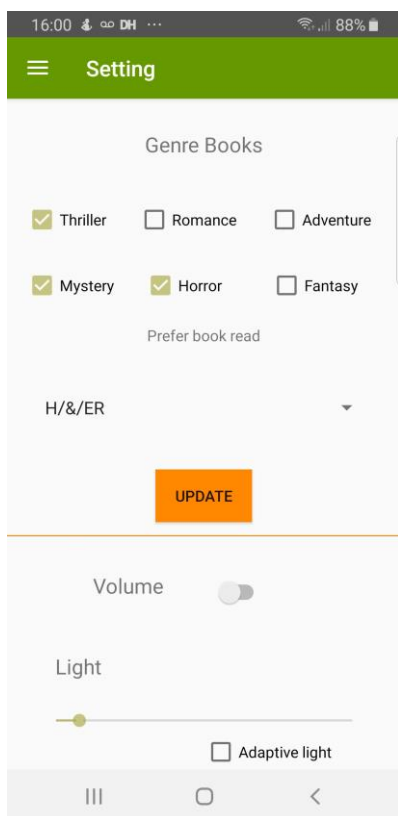


# PROFILE FRAGMENT

► The Profile fragment contains all the information set using the Home fragment and the detail fragment (the number of notes, and the number of books read) and the Setting fragment (the genre books and the best book for you).







► The part above of the divider let us to set information related on preference book so the set of genre books that a user can prefer and which is the best book which the user has read. Click on the UPDATE we make a PUT operation.

► The below part of the divider instead let us to play a song from the MediaPlayer class and manage the sensor of the light.

► The light sensor is enabled if we fill the checkbox otherwise if the checkbox is empty we can modify the light based on the Seekbar which is as a progression bar that fix with a single thread the functions of the System of the phone that we need to enable before entering into the Setting Fragment.

# SETTING FRAGMENT

# SERVER

►1) create a link from the personal server to mongodb database where we need to specify the user and the password for entering inside to the db through node.

►2) create crud operations with rest in the server:

►2.1) a post-> for creating a new user

►2.2) a put->for updating the document selected giving in input the email of that user

►2.3) a get-> for obtaining the document with that email as json file

►After developing the node server we need to care about how the android app is able to communicate with the mongodb instance? Usign the crud operation of the server .

►Retrofit is the class through which API interface(UserService) is turned into callable objects with the support of the converter Gson.

►Then according to the kind of CRUD operation we create a model class

►User(for Get)

►UserPost(for Post)

►UserPut(for Put)



►IMPORTANT: We need to specify from the android activity the port ad the address where we need to link for accessing to the the mongodb dbms.

►Port is specified into the server node

►Address is specified in press in prompt of command the value "ipconfig" and then we can take from it the value of the address.

# Google BOOKS REST API

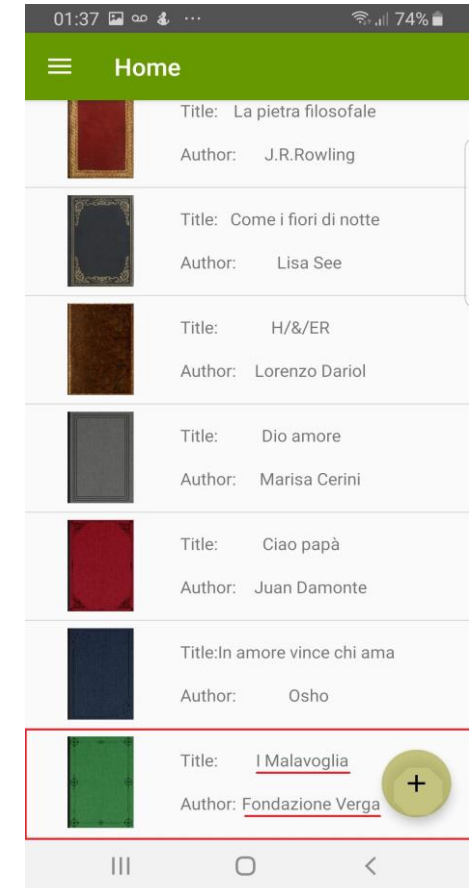
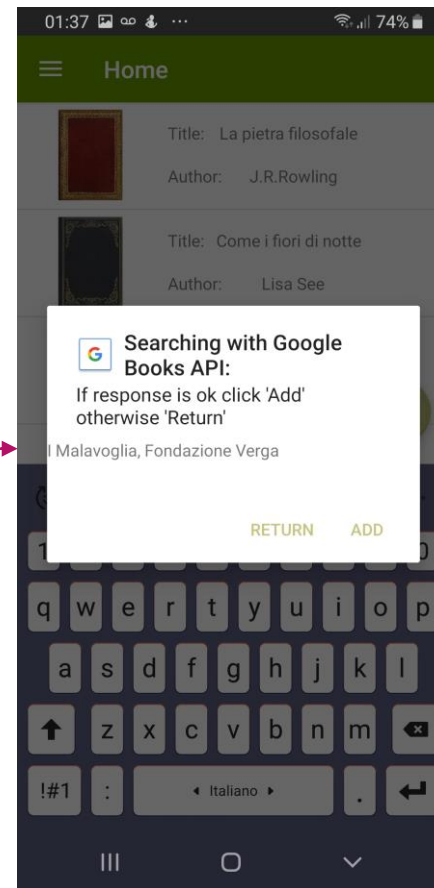
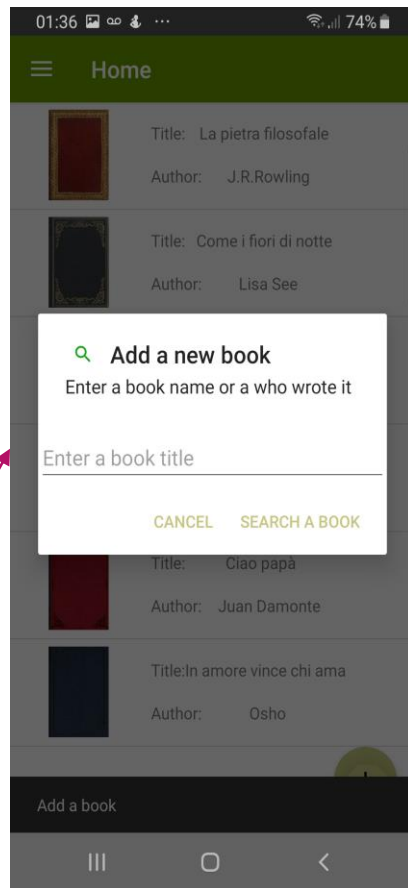
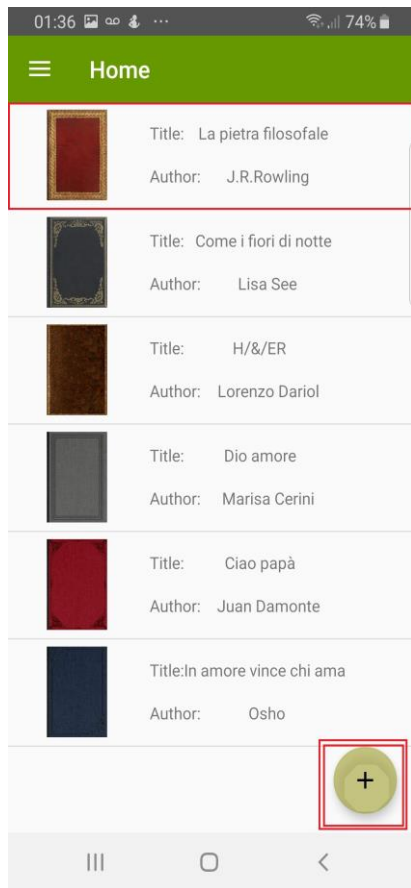


# Google BOOKS REST API

- ▶ Use the Google Book Search Data API to get and modify Book Search content via feeds. Submit full-text searches for books and get book information, ratings, and reviews
- ▶ In particular clicking on the floating button we try to add a book. According to it we're able to add into our virtual library a book.
- ▶ The user tries to look for the researched book based on the Title or Author and then the API let us to obtain the information need to insert it in the library.
- ▶ If the book is not right guessed by the API then ther's the possibility to not insert it, otherwise we can add it into our virtual library and then after a couple of second appear in the ListView with the CustomListView that set it in the right way.



# DEVELOPMENT –GOOGLE BOOKS API





# FUTURE WORK

- ▶ It could be interesting improve the following app with:
  - ▶ Possibility to add a book through the bar code (there's a Google API book that let us to research it)
  - ▶ Possibility to see an abstract of the book
  - ▶ Possibility to add a review
  - ▶ Possibility to create a list of suggest book based on the genre.