Project Proposal

Web Technologies

Spring 2021

Submitted to: M IKRAM ULLAH KHAN
Submitted by: Muhammad Musawar Baig
Roll No: 2018-UET-NML-CS-03 (1802003)
Dated: August 14th, 2021.

Acknowledgement:

I would like to express my sincere gratitude to Mr. Jaleel Awan for his insightful comments, helpful information, and practical advice that helped me tremendously to properly complete Treeio.

Treeio

1. Motivation:

Our Earth is facing the disastrous effects of global warming. Trees provide a crucial line of defense against global warming. Trees are life-givers and lifesavers but they also need our help to grow and remain healthy.

There is a need in this modern world for special software applications (especially Mobile Apps) that can make the process of keeping track of plants/trees easy. The "Treeio" is a contribution to counter the effects of global warming by facilitating to keep track of the plantations/trees.

2. Similar Applications:

The following are a few similar mobile Apps that I have found on the google play store:

1. Android App TreeCorder by EcoMatcher (Registration function is not working)
2. Android App Treetracker by Greenstand and https://map.treetracker.org/
3. My Tree by Appropriate Evolve Pvt. Ltd. (Not active yet (April 18th, 2021))

From the above applications, “Treetracker” was the only one in working condition. Following are some of the major problems that I have identified with Treetracker:

1. **No Authentication:** The users can make an account and log in using random numbers or dummy emails.
2. **Poor User Interface:** The user can't understand what He/She is doing after opening the Application because the User Interface of the landing screen is not appropriate.
3. **Not Multilingual:** Only the English Language is supported.
4. **No Integration of Maps in the Application:** Maps are not integrated into the mobile application. Users have to leave the App and visit a separate Website to view the records of trees.
5. **Privacy Concerns:** Users have to take an awkward selfie the 1st time they log in to go on the landing screen.
6. **Very limited functionality:** Users can’t even update or delete their records.
7. **Complexity:** It’s hard to find an individual record on their website because their website shows the records of all their users on one screen.

3. Requirement Gathering and Analysis:

3.1. Functional Requirements:

3.1.1. Reliable Internet Connection:

Treeio will require a reliable internet connection to function properly. It will be needed to handle images as they are stored on a remote database. It will also be required to get proper information about the location of the user, to get the weather information for the application and for the Ads, etc.

3.1.2. Location:

Treeio will also require that the GPS location of the phone is on.

3.1.3. Authentication:

Users will have to authenticate themselves using one-time password (OTP) when they login.

3.1.4. Images:

Treeio will also need access to the Camera and Gallery of the mobile phone to get the images of plants/trees.

4. Nonfunctional Requirements:

4.0.1. Speak the user's Language:

As the majority of Pakistanis can't read English. That's why Urdu will also be added to the Mobile Application to reach broader audiences.

4.0.2. Performance:

The performance of the Application is good on a reliable internet connection. Users don't have to wait much to save, retrieve and update the records. The performance might suffer on a very poor internet connection but we have not tested our application with an unreliable/poor internet connection.

4.0.3. Easy to use interface:

The User interface needs to be easy and straightforward so that less-educated users can also use the application without any problem.

5. Database (Freebase):

5.1. Firebase Authentication:

Following are the records of users in firebase authentication view:

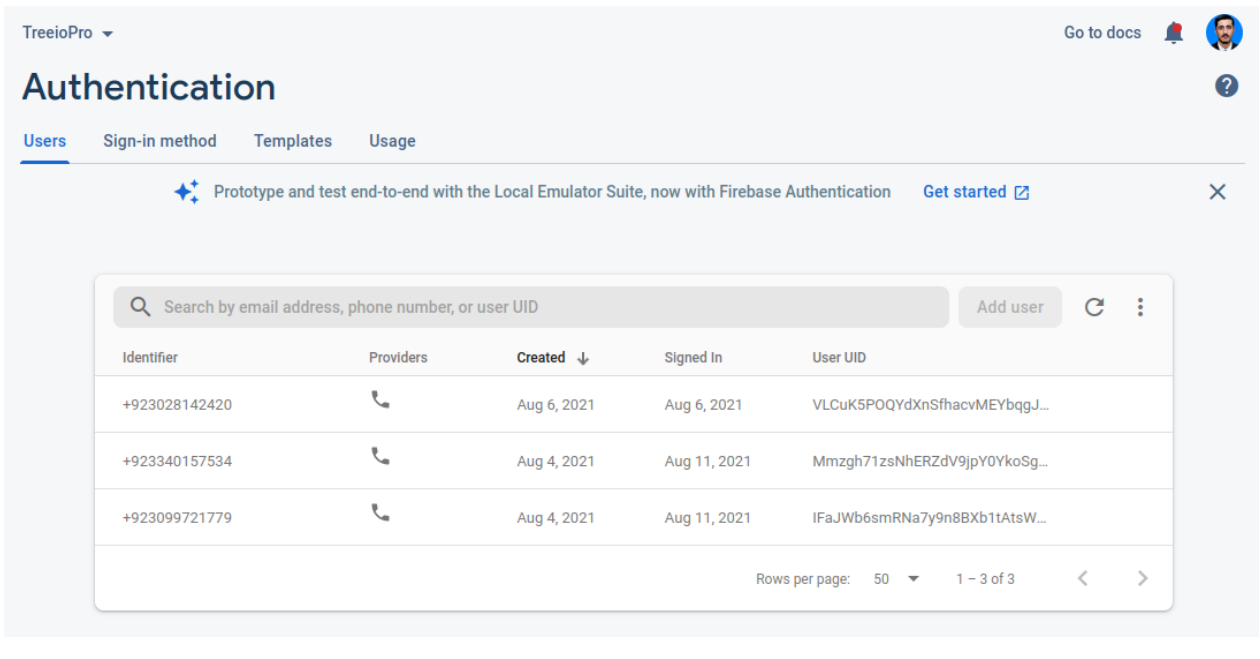


Fig: Firebase Authentication View

5.2. Firebase RealTime Database Schema:

Following is the Firebase RealTime Database Schema of Treeio:

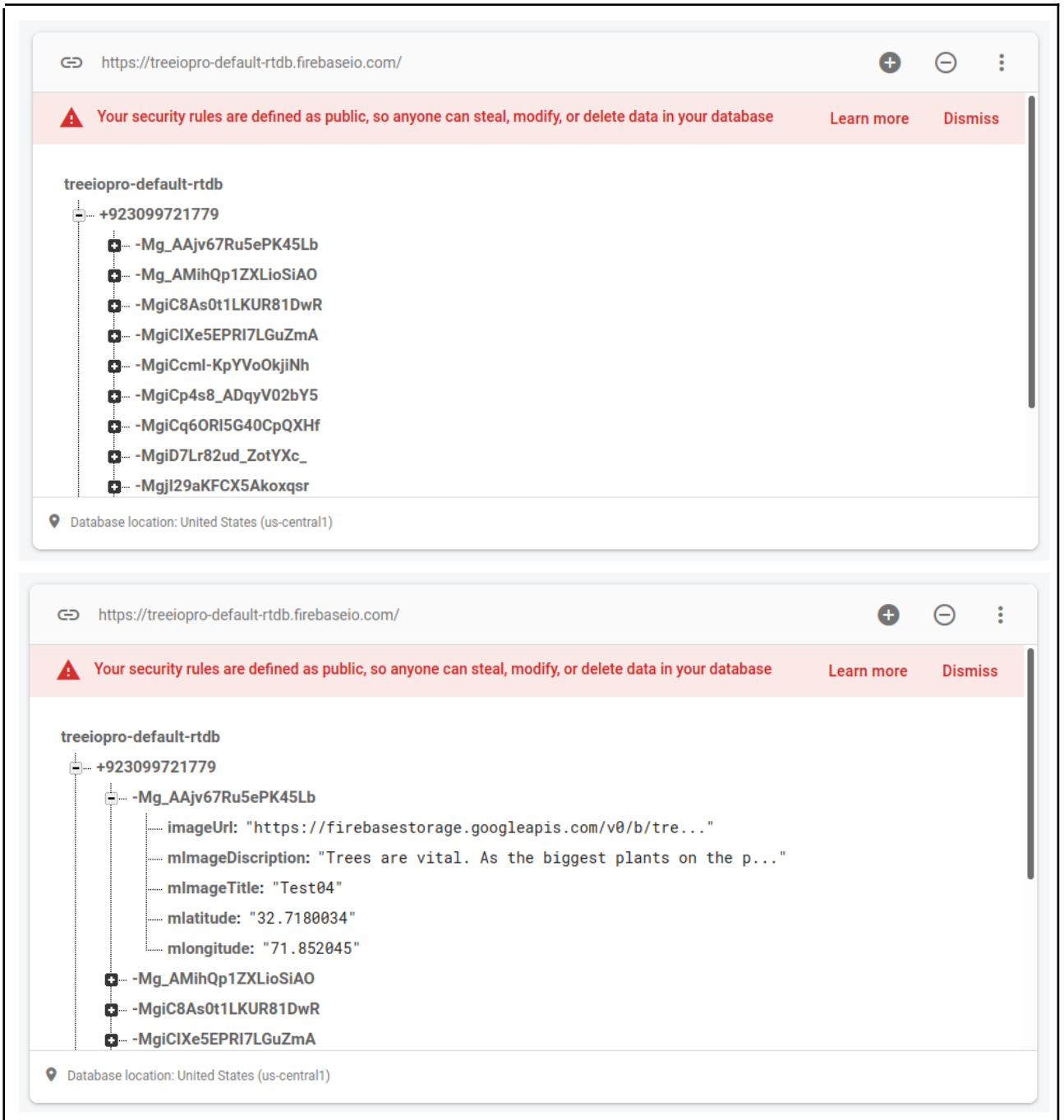


Fig: Firebase Database Schema

5.3. Firebase Storage:

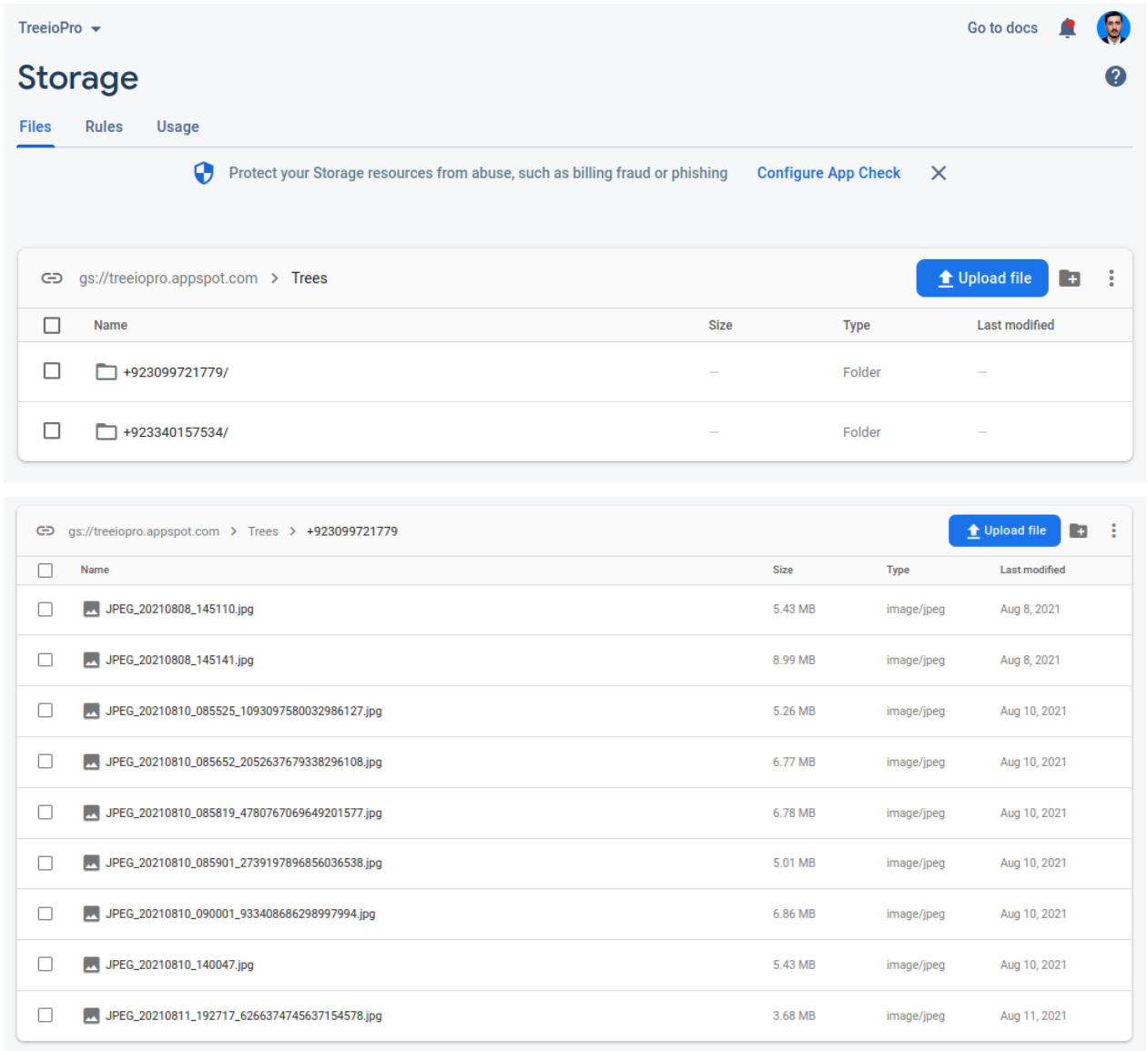


Fig: Firebase Storage View

6. Implementation of Features (Web App):

6.1. Implemented Features:

Following is the list of implemented features:

1. A user can login using email and password after firebase authentication.
2. A user can logout.
3. A user can view collective records on the Google map.
4. A user can also view records individually after clicking the records on the map. When a user clicks a record then the user can view the details (that includes image, title, and description of a tree) of the tree in the info window of the Google Map's marker.

6.2. Key Components/Libraries:

Following is the list of key components/libraries used in the Treeio:

1. Google Maps API
2. Flask
3. Ajax, JQuery
4. Session, Json
5. HTML, CSS, JavaScript
6. Bootstrap
7. Prebase, Python
8. Firebase Database

7. Major Screens (Web App):

Following are the major screens of Treeio (Web Interface):

7.1. Login Page:

Note: Kindly use the following credentials for testing:
email: abc@gmail.com
password: 11111111
or
email: saif@gmail.com
password: 22222222

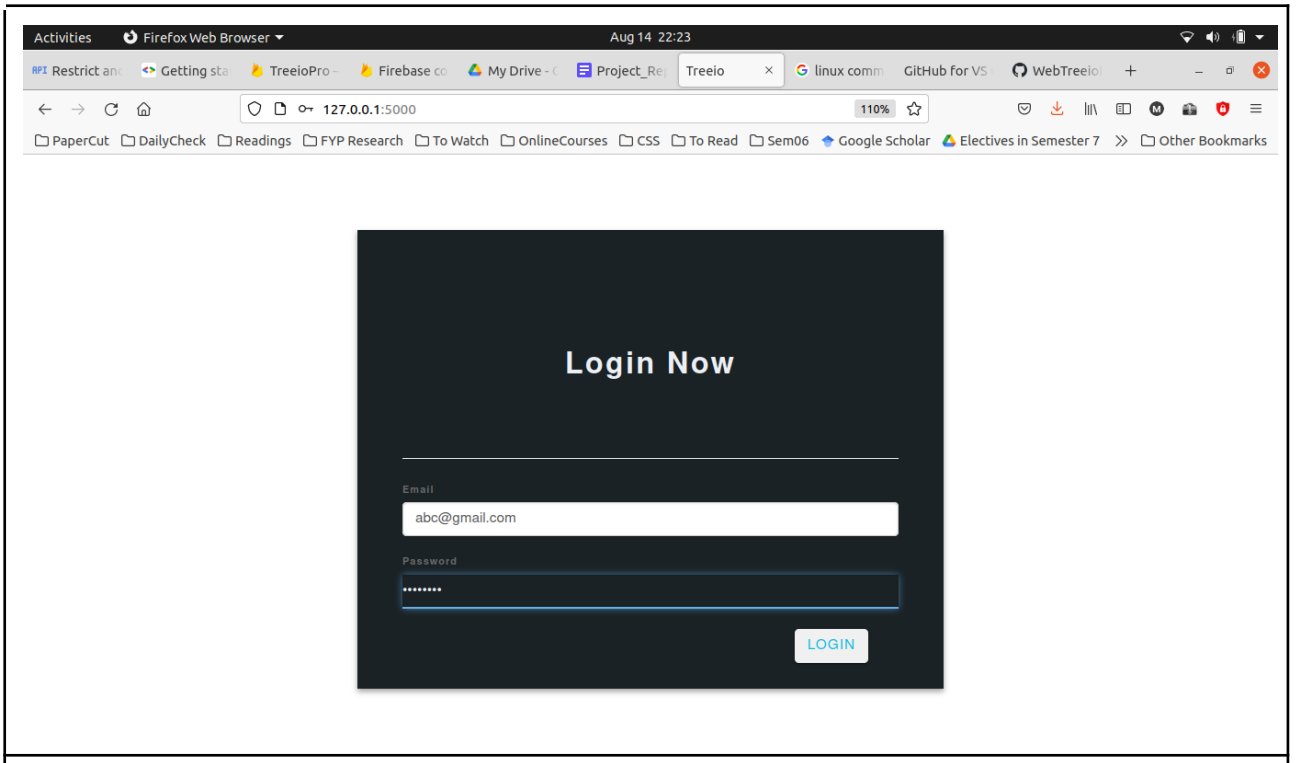
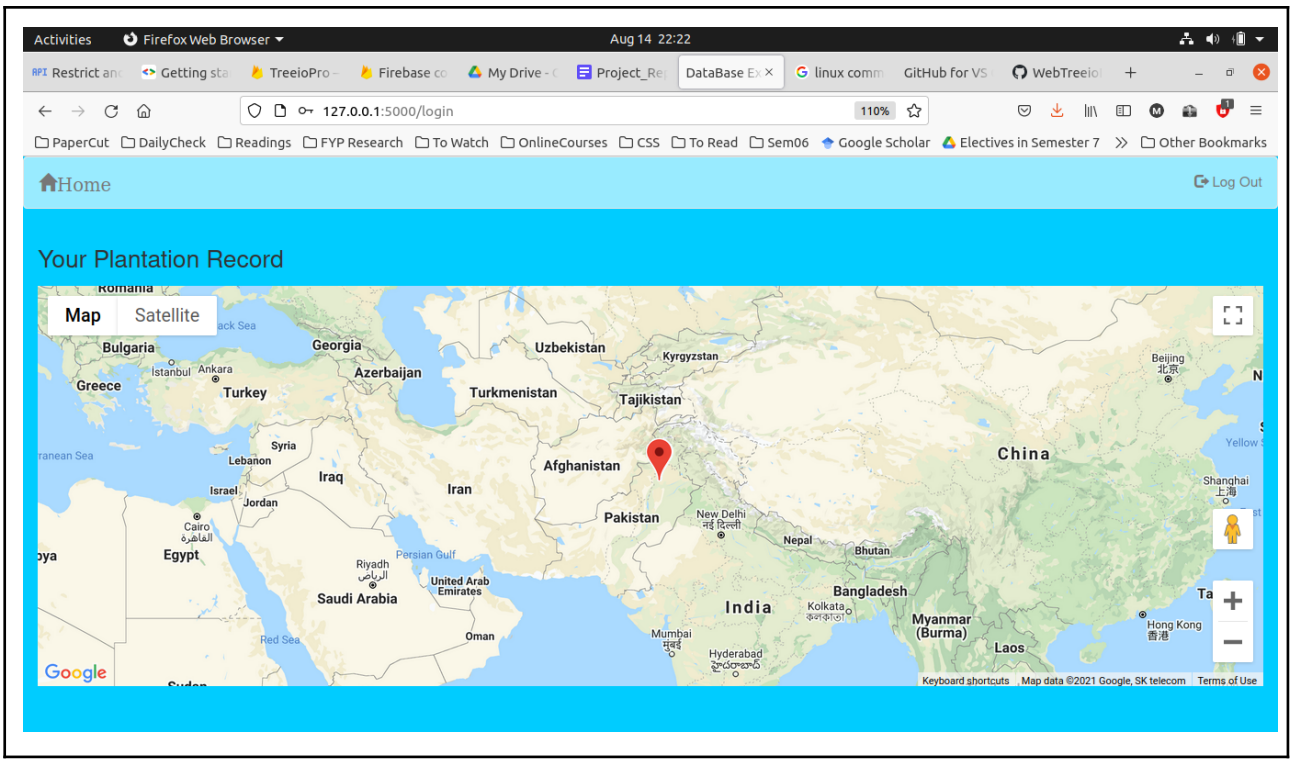


Fig: Login Page

7.2. Landing Page:



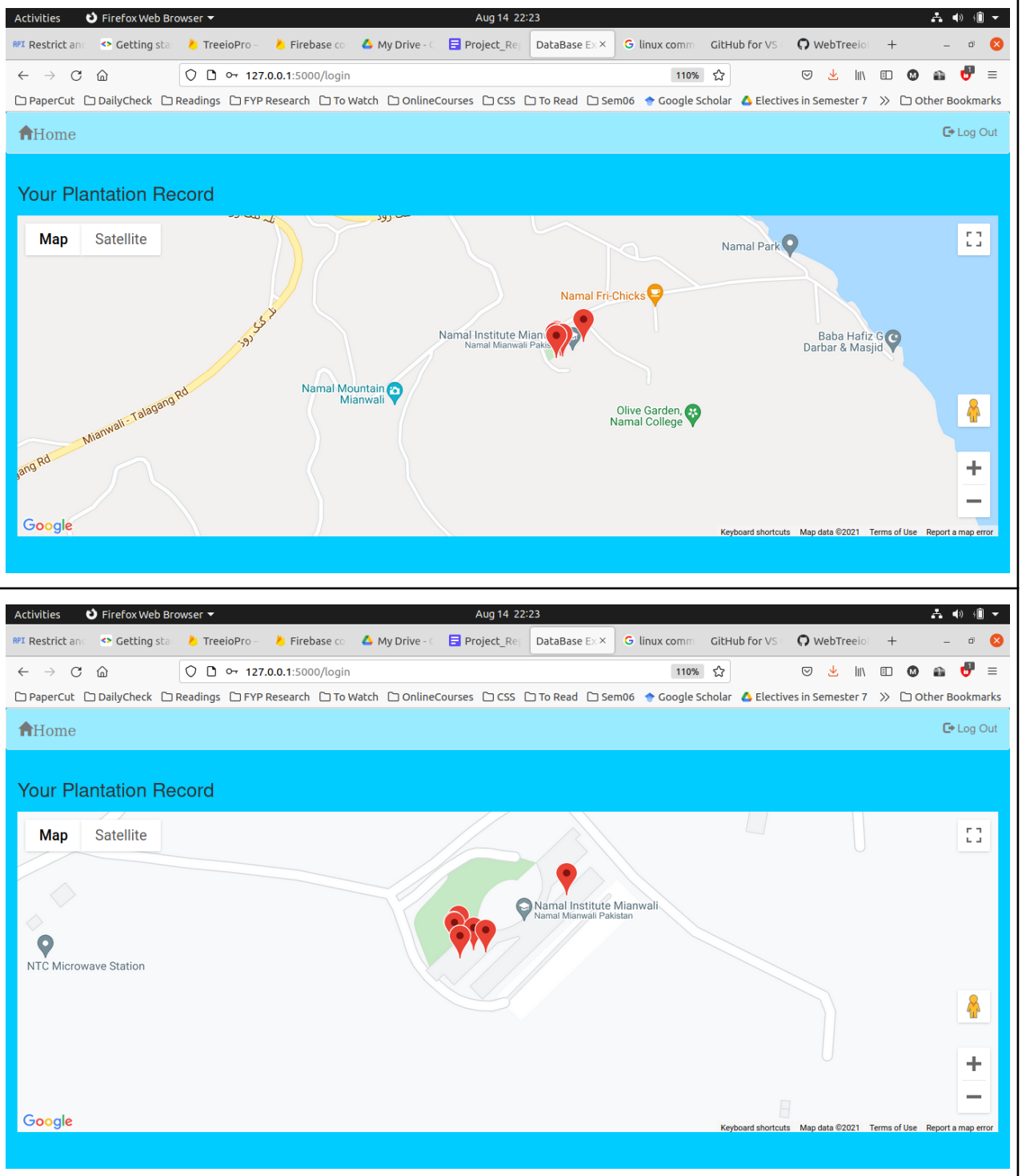


Fig: Landing Page, Each marker corresponds to a single tree.

8. Github Links:

The code of Treeio (Mobile App) can be accessed via this link. **The code of Treeio (Web App) can be accessed via this link.**

9. Initially Proposed Features (Mobile App):

Following is the list of initially proposed features:

1. It will store the location (Google location pin or coordinates) of a tree. (newly planted or
2. old)
3. It will track the progress of that Tree/plant by storing pictures, age, type, and other
4. significant details.
5. Users will be able to track his/her trees on a map.
6. Users will be able to track the progress of his/her trees from pictures that will be stored
7. in a timeline.
8. Users will be able to log in and log out.
9. Users will be able to assign themselves tasks like Aerating or Watering day and time.
10. Functionality to alert the user for his/her tasks.

10. Prototype (Mobile App):

I have used paper prototypes for this project. I have learned the concept of “paper prototypes” in my course on Human-Computer Interaction(HCI). I preferred paper prototypes because I find paper prototypes less time-consuming compared to making them via a software tool. They helped me to get quick feedback from my project supervisor (Mr. Jaleel Awan) and refine the requirements of my application. Paper prototypes can be accessed on the following link using the email address belonging to Namal Institute:
Link

Note: Kindly ignore the quality of the prototypes. I can understand that they are not very professionally made but They worked for me and helped me a lot in my Project. They also included key points that were discussed in the meeting that I had with my project supervisor.

11. Implementation of Features (Mobile App):

11.1. Implemented Features:

Following is the list of implemented features:

1. A user can login using a one time password (OTP).
2. A user can logout.
3. A user can add/update/delete a record about his/her trees/plants.
4. A user can view collective records on the Google map.
5. A user can also view records individually after clicking the records on the map .
6. Application is multilingual (English & Urdu)
7. The developer can earn from Ads as well.

11.2. Key Components/Libraries:

Following is the list of key components/libraries used in the Treeio:

1. Google Maps
2. Open Weather API

3. Volley
4. Glide
5. Picasso
6. Gson
7. Camera/Gallery to get images
8. Google Banner Ads

12. Major Screens (Mobile App):

Following are the major screens of Treeio:

12.0.1. Welcome Screen:

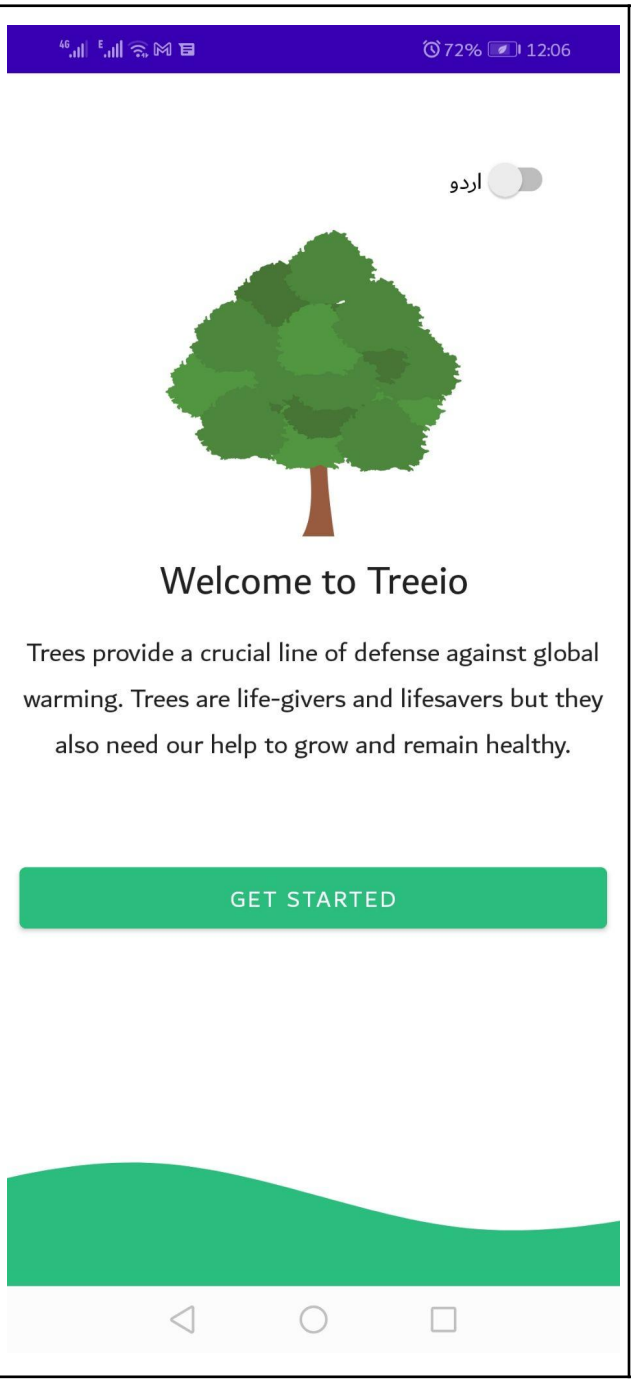


Fig: Welcome Screen

12.0.2. Login Screen:

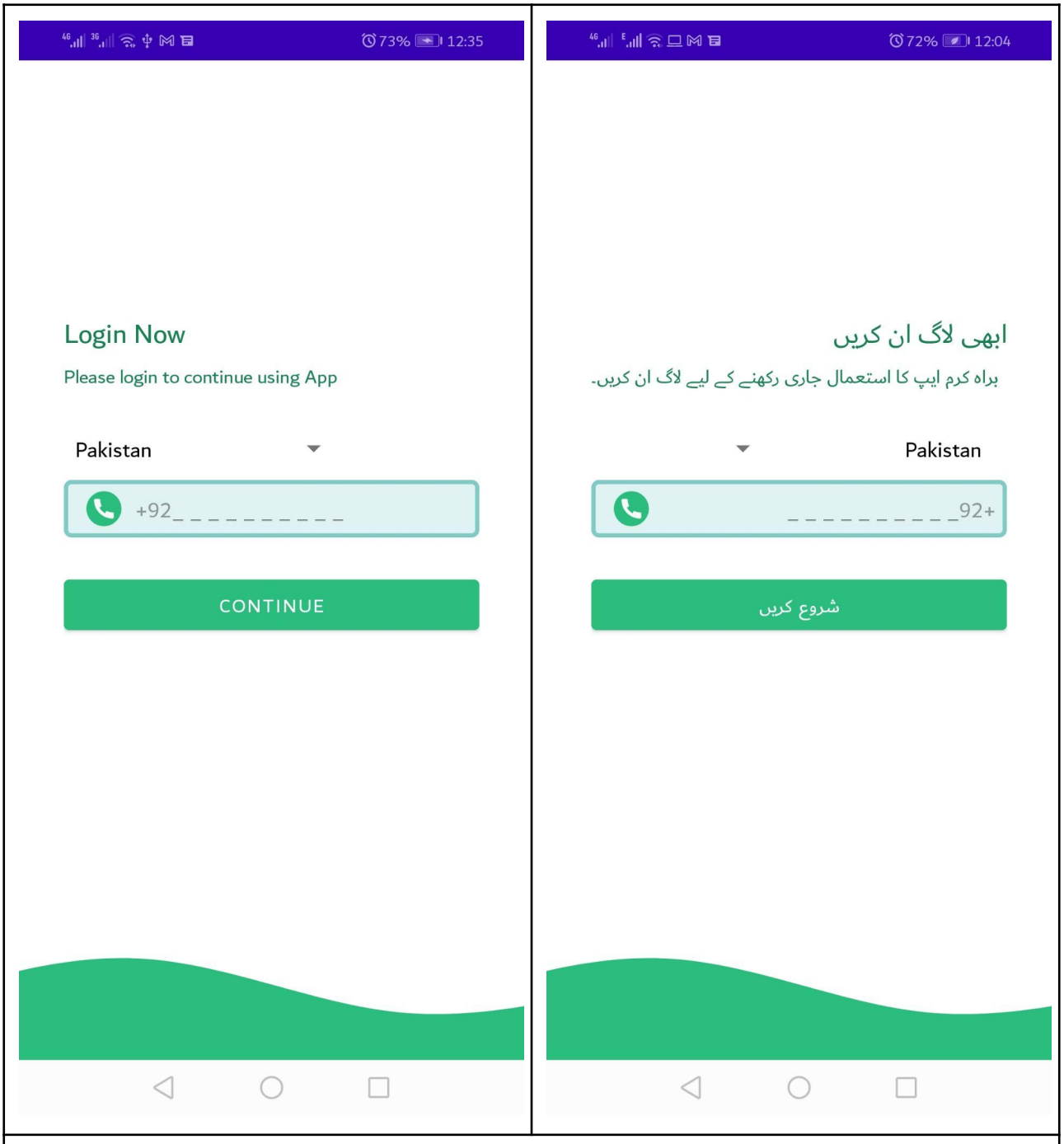


Fig: Login Screen

12.0.3. Verification Screen:

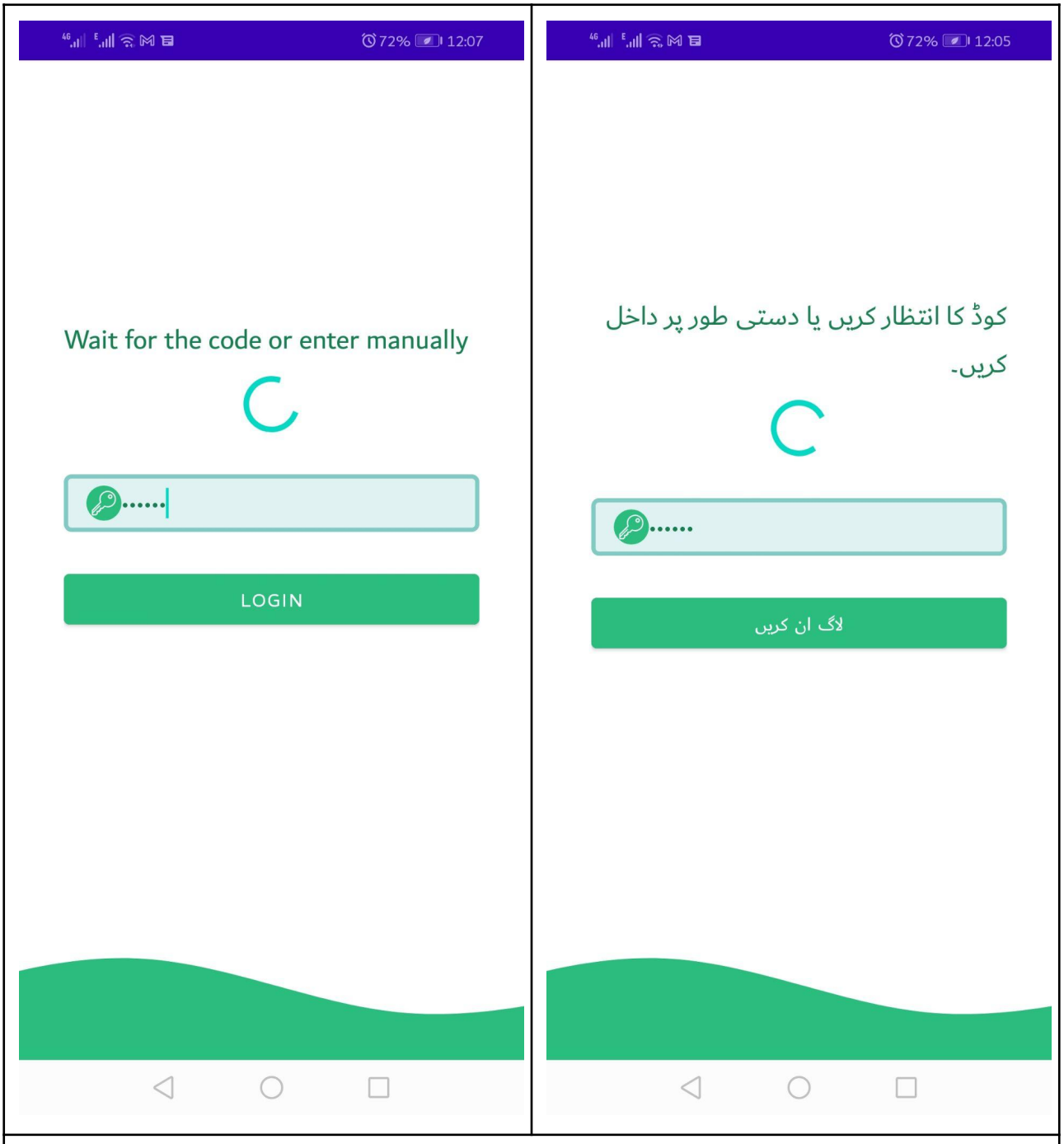


Fig: Verification Screen

12.0.4. Main Screen:

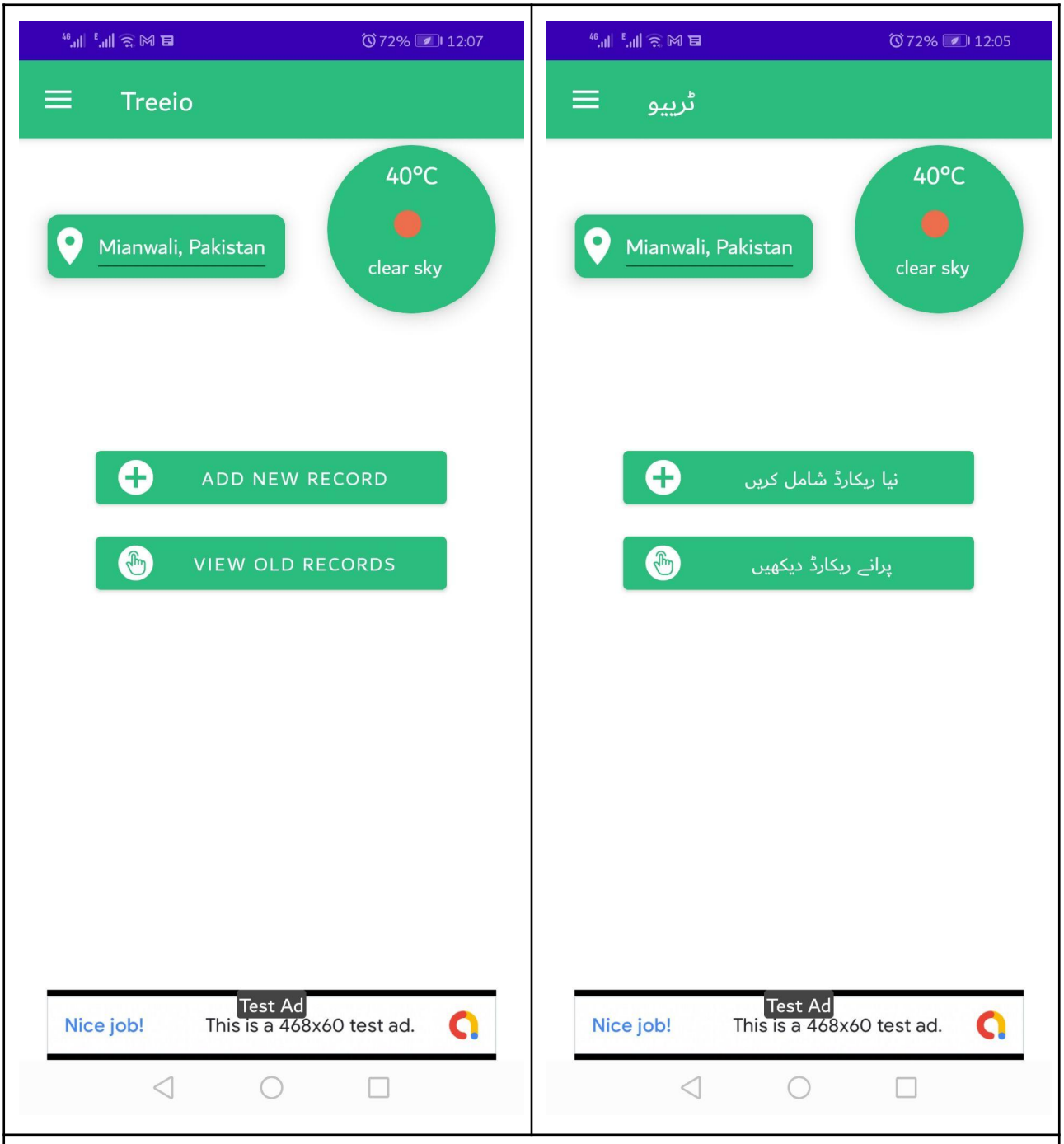


Fig: Main Screen

Note: In the above main screen, the location and weather details are still in English while the locale is “Urdu” because those details are dynamic. Open Weather API and Geocoder use the English language.

12.0.5. Navigation Drawer:

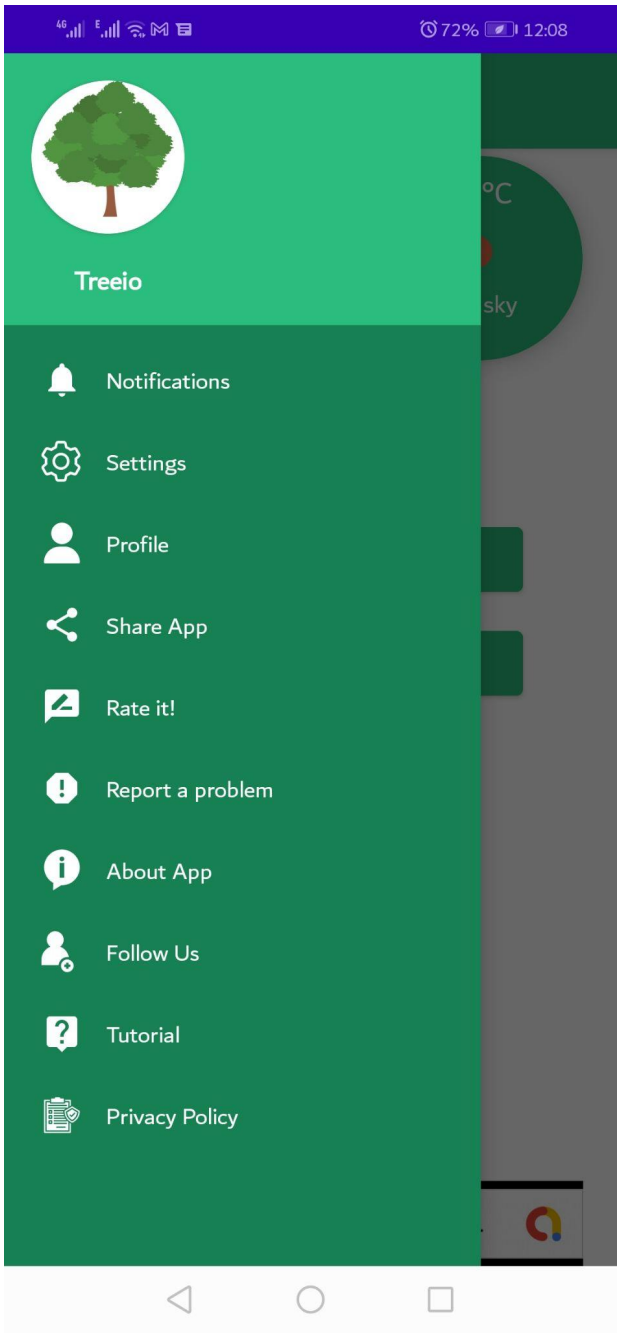


Fig: Navigation Drawer

12.0.6. View Old Records on Map Screen:

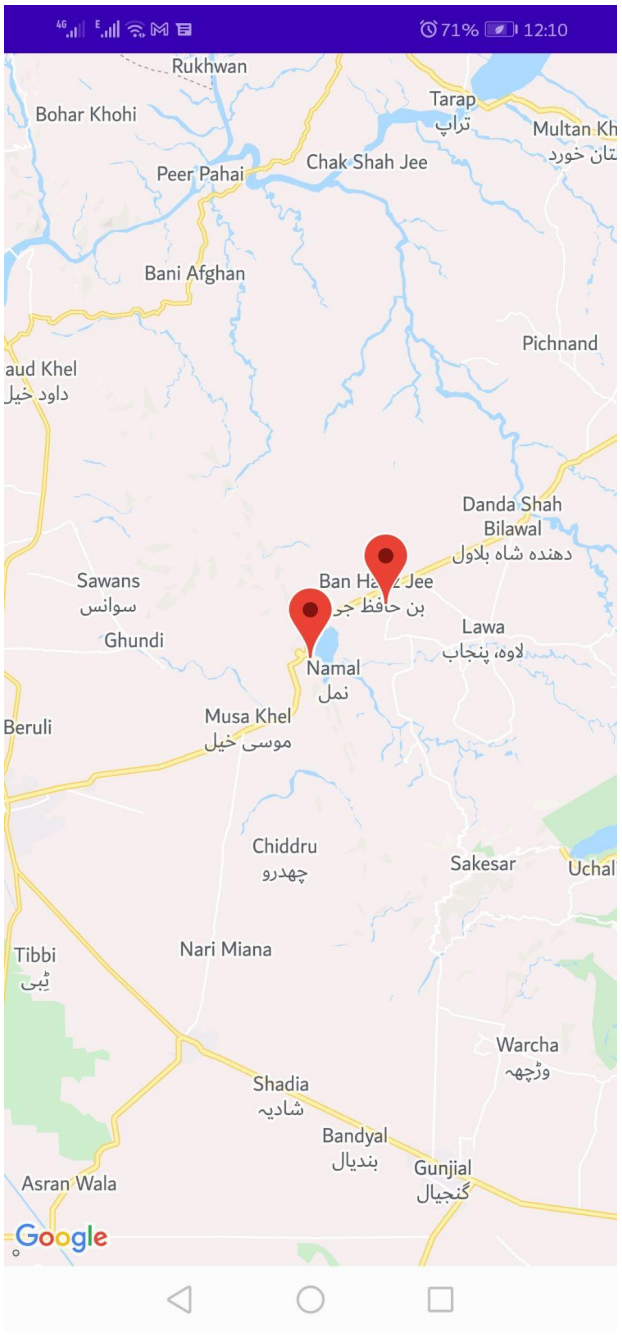


Fig: View Old Records on Campus

12.0.7. View Individual Records Screen:

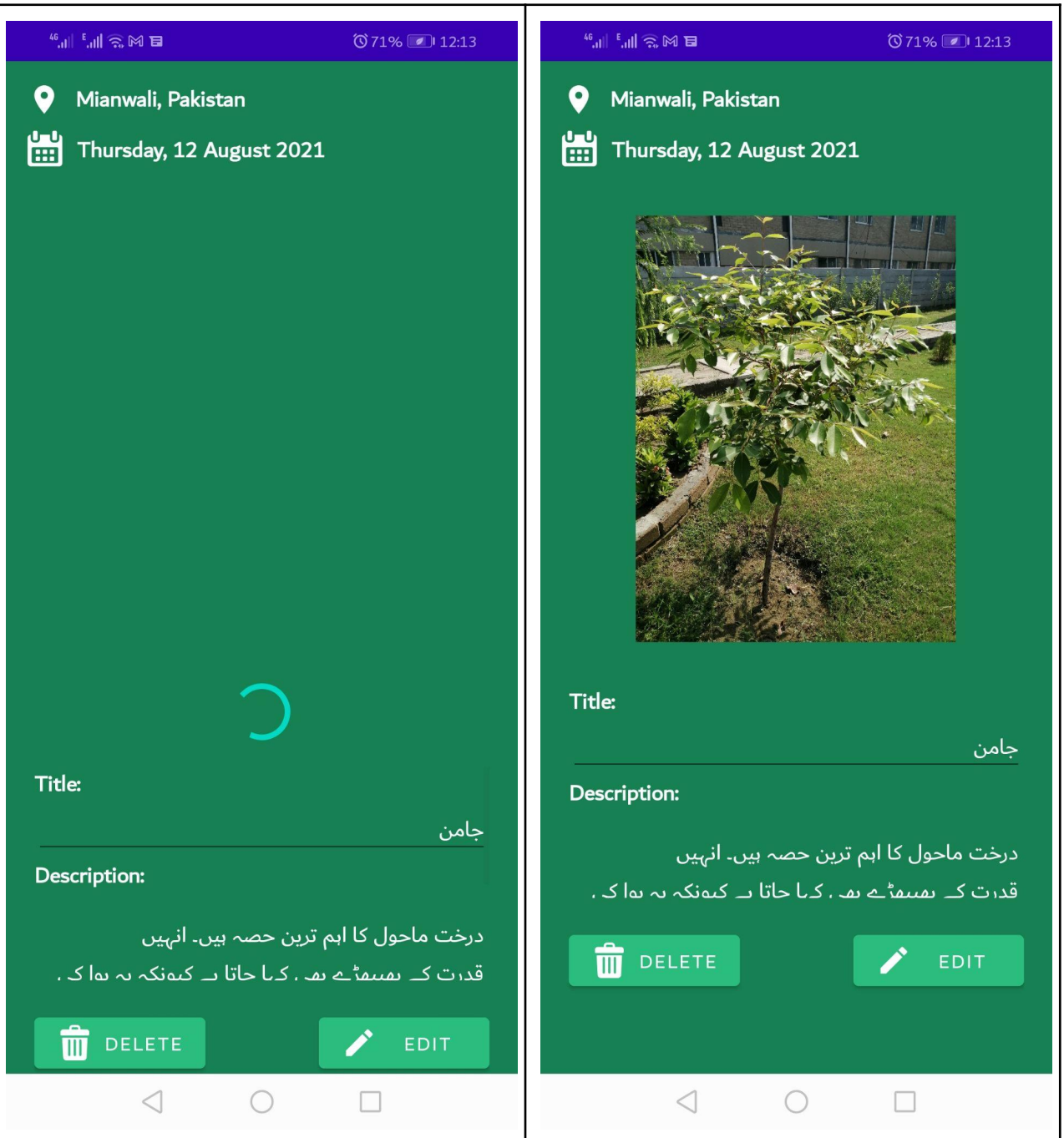


Fig: View Individual Records Screen

12.0.8. Update Records:

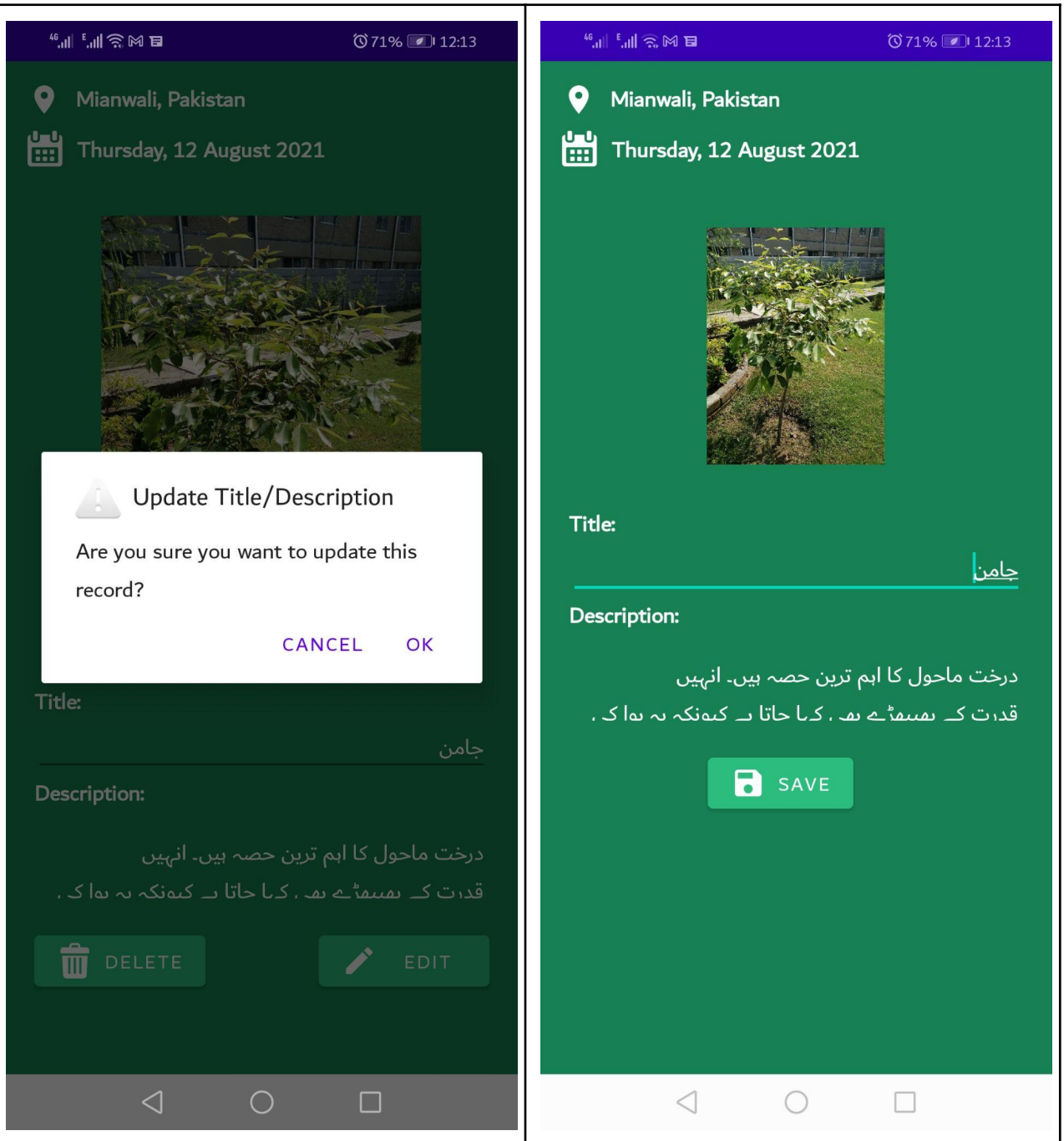


Fig: Update Records

12.0.9. Delete Records:

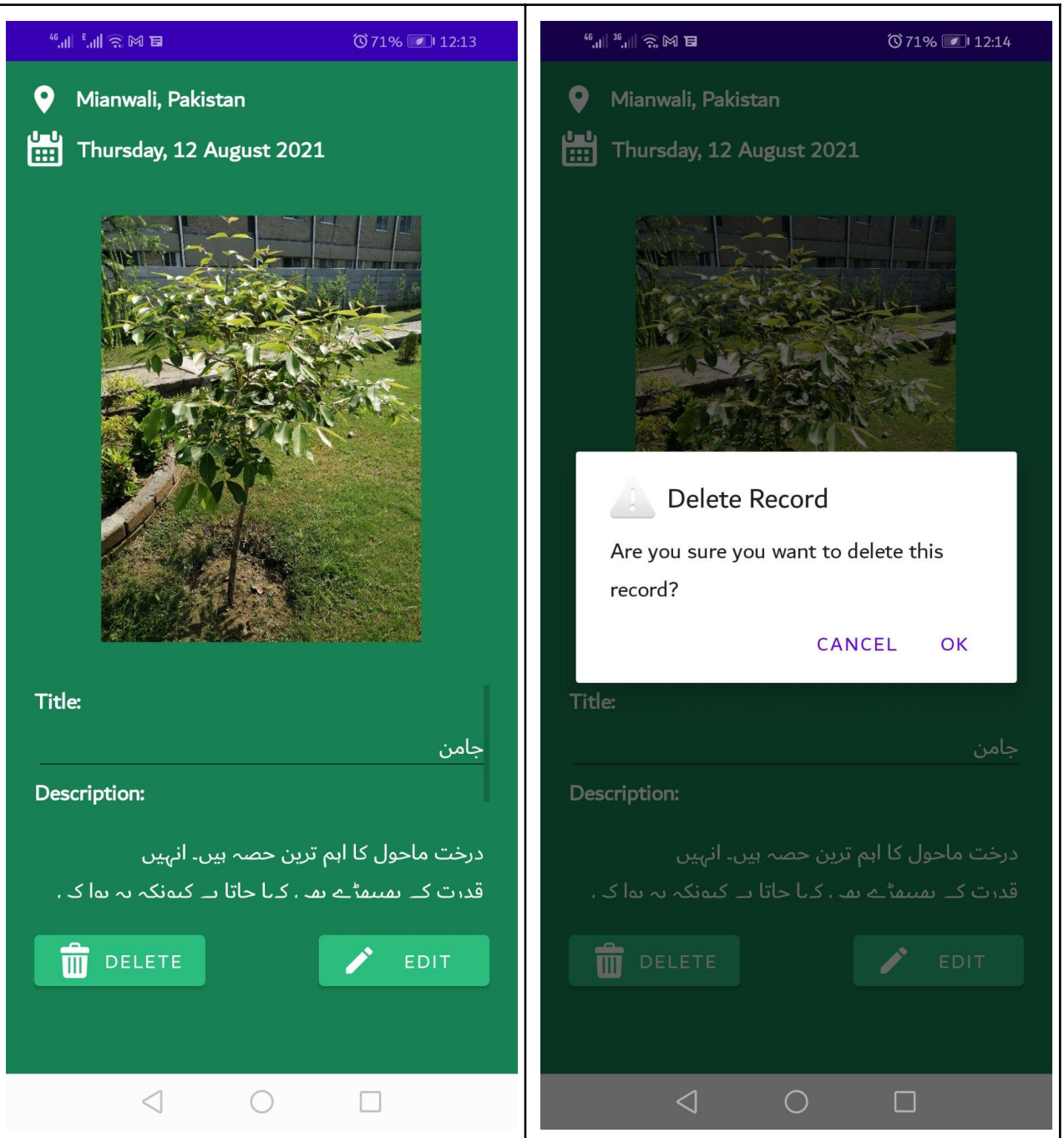


Fig: Delete Records

12.0.10. Logout Screen:

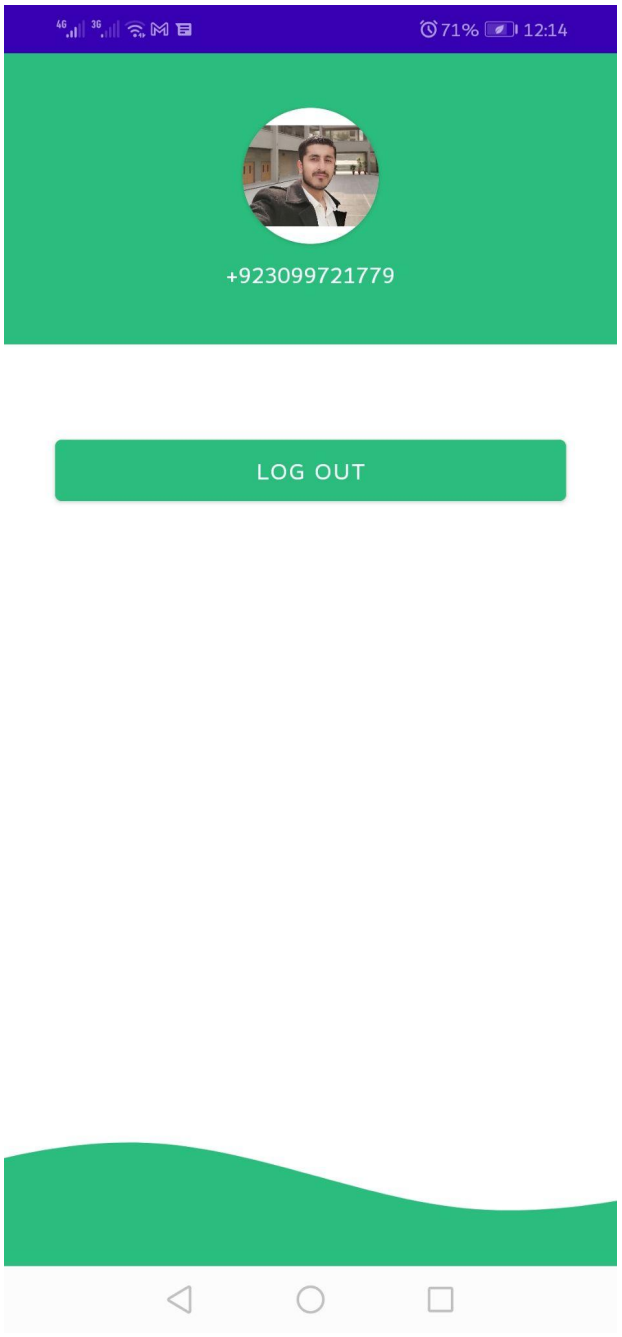


Fig: Logout Screen

13. Dependencies (Mobile App):

Following are the dependencies of my application:

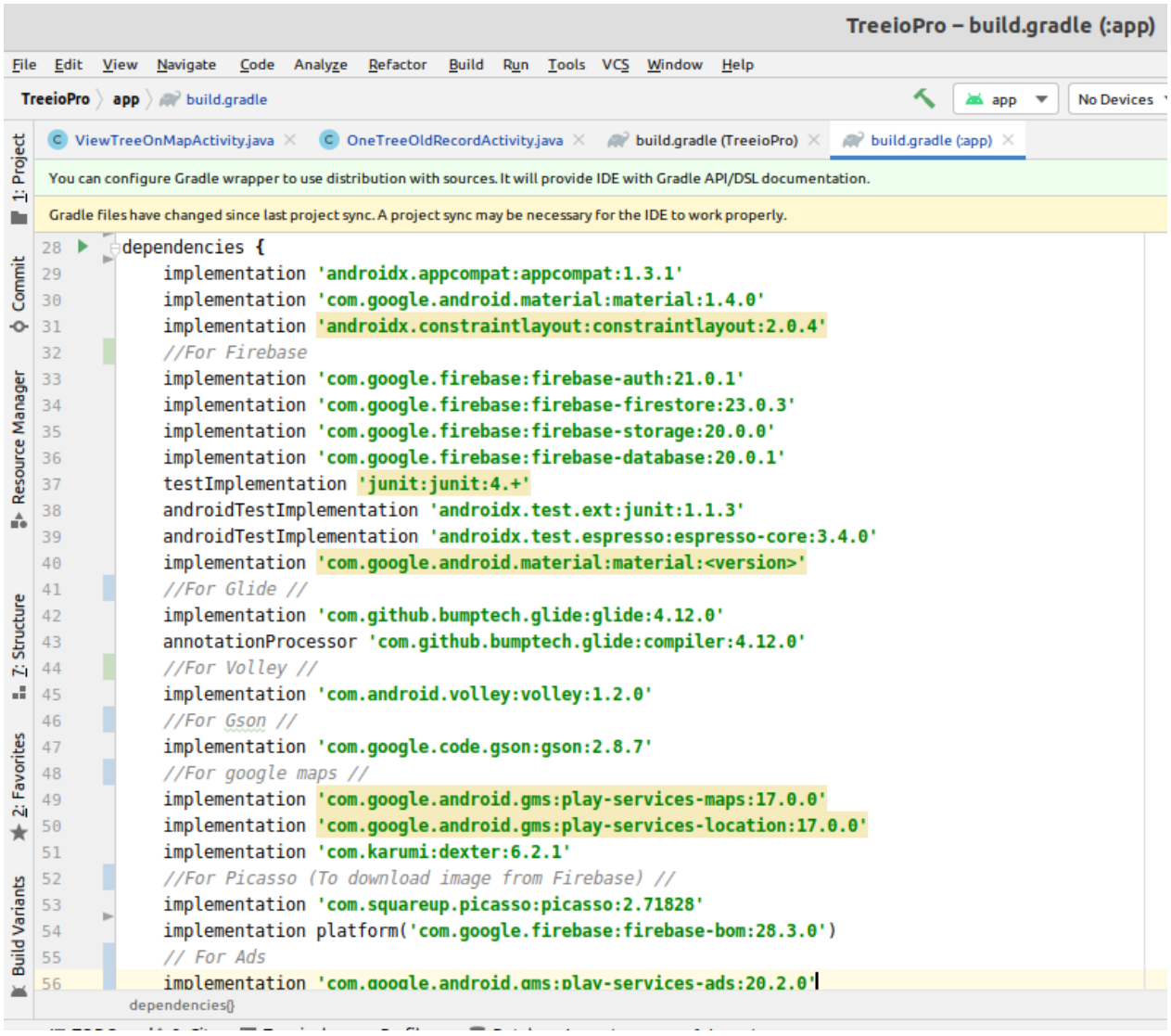


Fig: View of the dependencies file

14. Bibliography:

---------->For Web Interface<----------

1. *Pyrebase*. *github*, https://github.com/thisbejim/Pyrebase. Accessed 2021.
2. *Flask-Firebase Authentication - simple, easy webapp*. Virtual Monk, 2019. *Youtube*, https://www.youtube.com/watch?v=7NjyOx8fACA&t=310s. Accessed 2021.
3. *Google Maps Api Documentation*. https://developers.google.com/maps/gmp-get-started#enable-api-sdk. Accessed 2021.
4. *Template login page*. *codepen*, https://codepen.io/xmas1224/pen/MWJqbao. Accessed 2021.
5. *How to enable Google Maps Platform APIs and SDKs*. Google Maps Platform, https://www.youtube.com/watch?v=n1UorU1PALk&t=48s. Accessed 2021.

---------->For Android Mobile App<----------

6. App Studio, Game. *How to create the multi-language app in Android Studio | Game App Studio*. Game App Studio, 2020, https://www.youtube.com/watch?v=wXNyTunz0B0.
7. Demos, Coding. *How to Take a Photo Using The Camera And Display it in Android Imageview*. Coding Demos, 2020. *YouTube*, https://www.youtube.com/watch?v=YLUmfyGFjnU.
8. “For Errors etc.” *stackoverflow*, 15 sep 2008, https://stackoverflow.com/. Accessed 2021.
9. “For help and understanding.” *developers*, https://developer.android.com/. Accessed 2021.
10. “For Icons.” *Flaticons*, https://www.flaticon.com/. Accessed 2021.
11. Hedlund, David. *How do I display an alert dialog on Android?* David Hedlund, 2019. *stackoverflow*, https://stackoverflow.com/questions/2115758/how-do-i-display-an-alert-dialog-on-android.
12. I/O, Google. “Help.” *Firebase*, Google, 2011, https://firebase.google.com/. Accessed - July 2021.
13. Jamal, MD. *Current location in google map Android Studio | Current location on Google map*. MD Jamal, 2020. *YouTube*, https://www.youtube.com/watch?v=kRAyXxgwOhQ.

14. Ramani, Ketan. *How to convert Date to a particular format in android?* Ketan Ramani, 2019. *stackoverflow*, https://stackoverflow.com/questions/35939337/how-to-convert-date-to-a-particular-format-in-android.

15. Skillz, Technical. *Android Firebase - 10 - How Update Data in Firebase Realtime Database.* Technical Skillz, 2019. *YouTube*, https://www.youtube.com/watch?v=0HLyJNuyhSo.

16. SmallAcademy, SmallAcademy. *Pick captured Image From Gallery For ImageView | Android App Development Tutorials | Part 2*. SmallAcademy, 2020. *YouTube*, https://www.youtube.com/watch?v=KaDwSvOpU5E.