



IT2010 – Mobile Application Development

BSc (Hons) in Information Technology

2nd Year

Faculty of Computing

SLIIT

2023 – Assessment 02

Description	Student ID	Name
Group Leader	IT21169380	Thuduwage I.M.H.G
Member 2	IT21189944	Madusanka G.K.I
Member 3	IT21186042	Ekanayaka E.M.D.K.L
Member 4	IT21169144	Karunarthne R.Y.D

Application Topic	Reduce Poverty in Rural SriLanka
Group Name	Neighbourly – Y2.S2.IT.04.01

Project Declaration

We, the members of “Neighbourly – Y2.S2.IT.04.01”, hereby declare that our group project is entirely authentic and original. We have conducted thorough research and analysis to ensure that our work is not plagiarized or copied from any other sources.

We have followed all guidelines provided by our LIC and have complied with all ethical and academic standards.

We take full responsibility for the authenticity of our work and understand the implications of academic dishonesty. We have worked collaboratively to produce this project, and each member has contributed to the best of their abilities.

We hereby affirm that our project represents our honest effort and commitment to academic integrity, and we take pride in presenting it as our own.



Group Leader (Signature)
<<Group Leader Name>>

Contents

Introduction	4
Git Repository:	5
Contribution.....	6
1. IT21169380 (Thuduwage I.M.H.G) – User Profile and Money Donations	6
2. IT21189944 (Madusanka G.K.I) – Item Donations	11
3. IT21186042(Ekanayaka E.M.D.K.L) – Family	15
4. IT21169144(Karunarthne R.Y.D) – Article	19
UnitTest For Calculation Part	22

Introduction

Neighbourly is a mobile application built using Kotlin language aimed at reducing poverty in rural Sri Lanka and helping people in economic crises. The app connects volunteers and donors, where volunteers can sign up and provide their details, including their National Identity Card (NIC) for verification. Donors can donate money or items, such as food, school accessories, and medical items. Also, Donors can choose specific categories such as child orphanages, elder houses, low-income families, and low-facility schools for their fund donations. The donor dashboard displays the total fund amount, daily money donations, the donor's donations made in all time, as well as the donation overview for each category. Donors and volunteers can edit or delete their accounts.

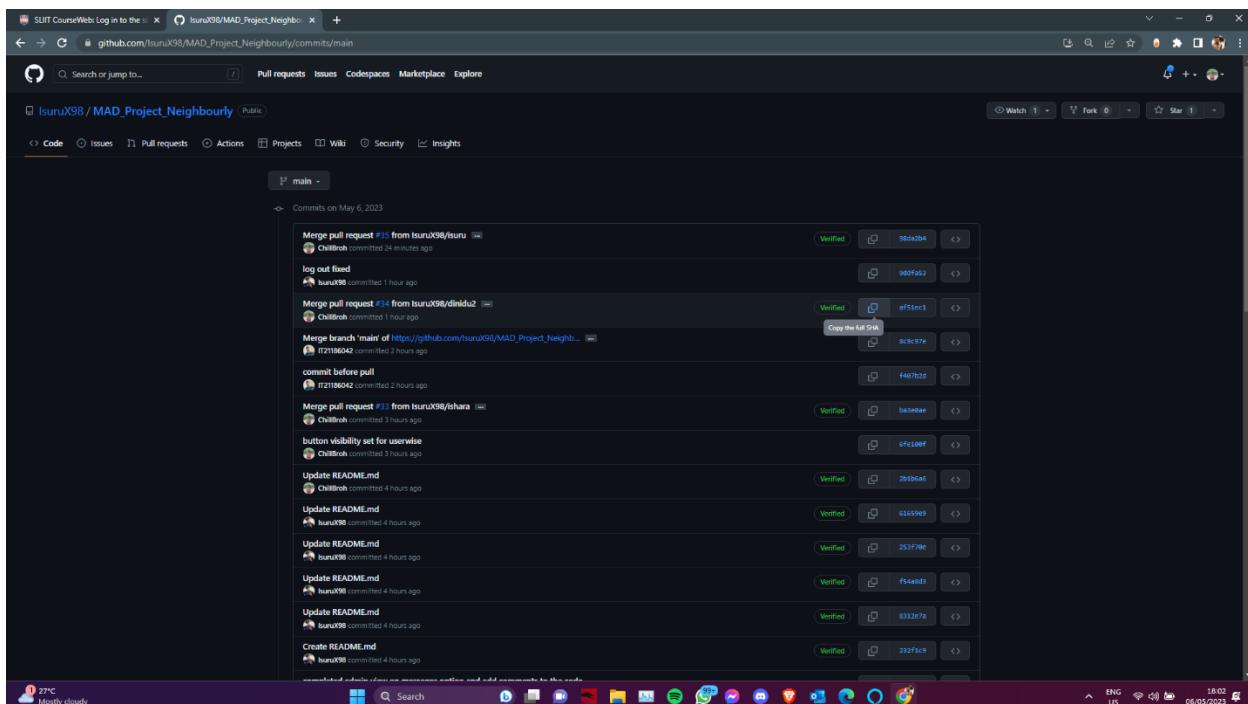
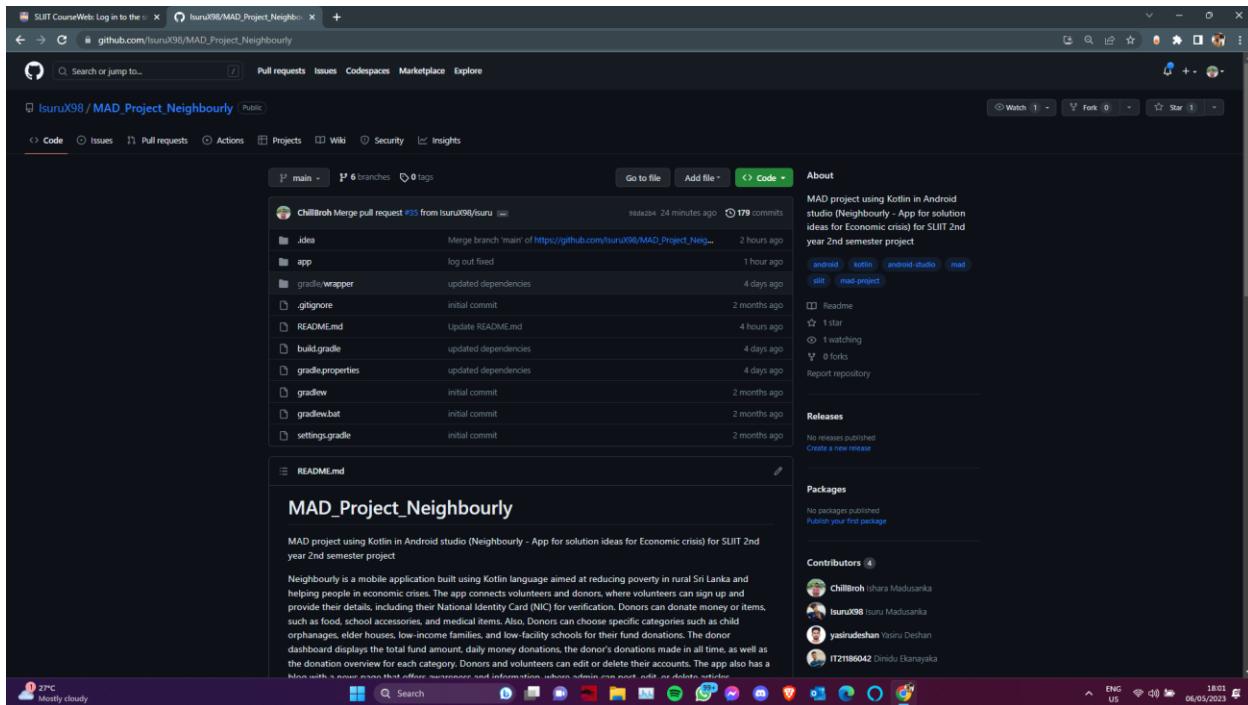
The app also has a blog with a news page that offers awareness and information, where admin can post, edit, or delete articles. Volunteers can add selected low-income families in their area, and anyone can donate their remaining or prepared food for those families, reducing food wastage and hunger simultaneously. The system sends notifications to relevant volunteers to connect the donated food with those in need. The "Contact Us" page allows users to contact the team with any inquiries or concerns, and inquiries are directed to the relevant parties via email. Neighbourly provides a platform to make a positive impact on people's lives and communities in need.

Neighbourly plans to enhance the app's functionality by adding a volunteer delivery system for those who are unable to collect the available items. The app also aims to connect with the government health sector and register local officers such as Public Health Inspectors (PHI) to obtain key info about people in a given area, such as their BMI values, number of children, and monthly income, in order to prioritize the areas that require the most help and support.

Neighbourly is a non-profit app that supports its operations through advertising from item sellers, pharmacies, food cities, restaurants, and shops. With its noble purpose and innovative features, Neighbourly has the potential to make a significant impact on poverty reduction and create a more equitable and inclusive society in Sri Lanka.

Git Repository:

<https://github.com/IsuruX98/MAD Project Neighbourly.git>



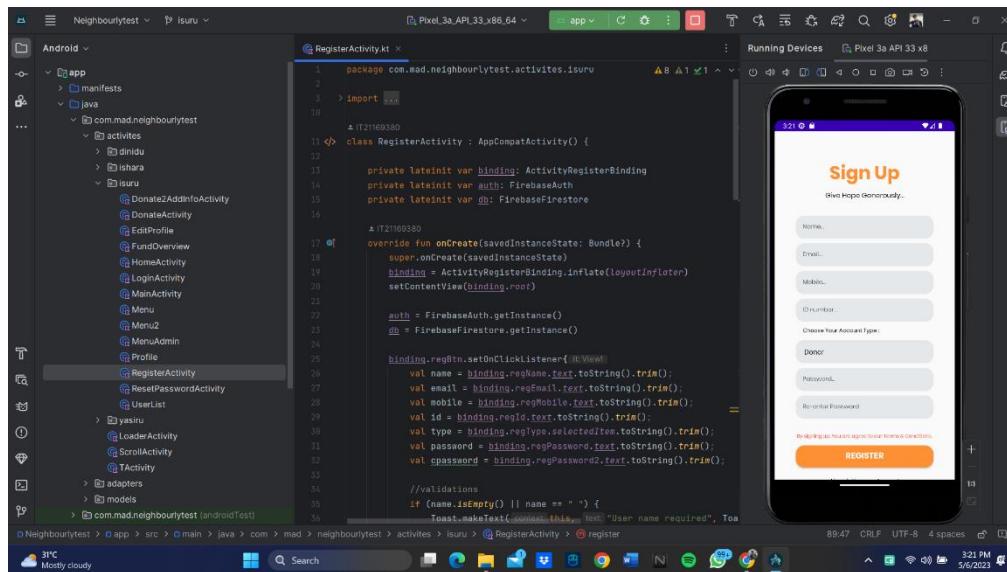
Contribution

1. IT21169380 (Thuduwage I.M.H.G) – User Profile and Money Donations

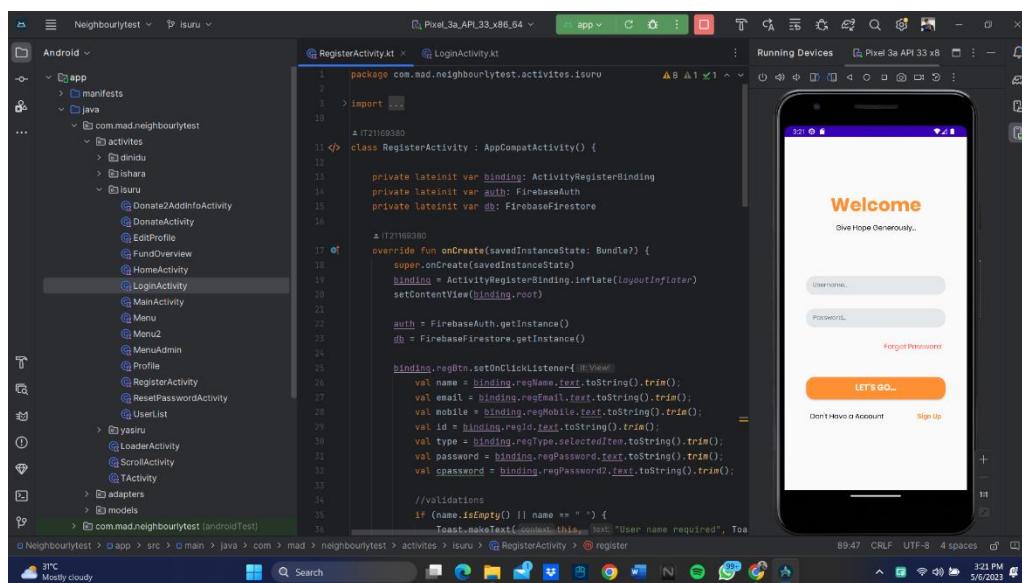
User Management

The user management system allows users to register to the system and store their data in the Firebase Fire store database. When a user registers, their email and password are also stored in the Firebase authentication system as user data. Users can then log in to the system and their credentials will be authenticated. If a user forgets their password, they can reset it in the reset password activity.

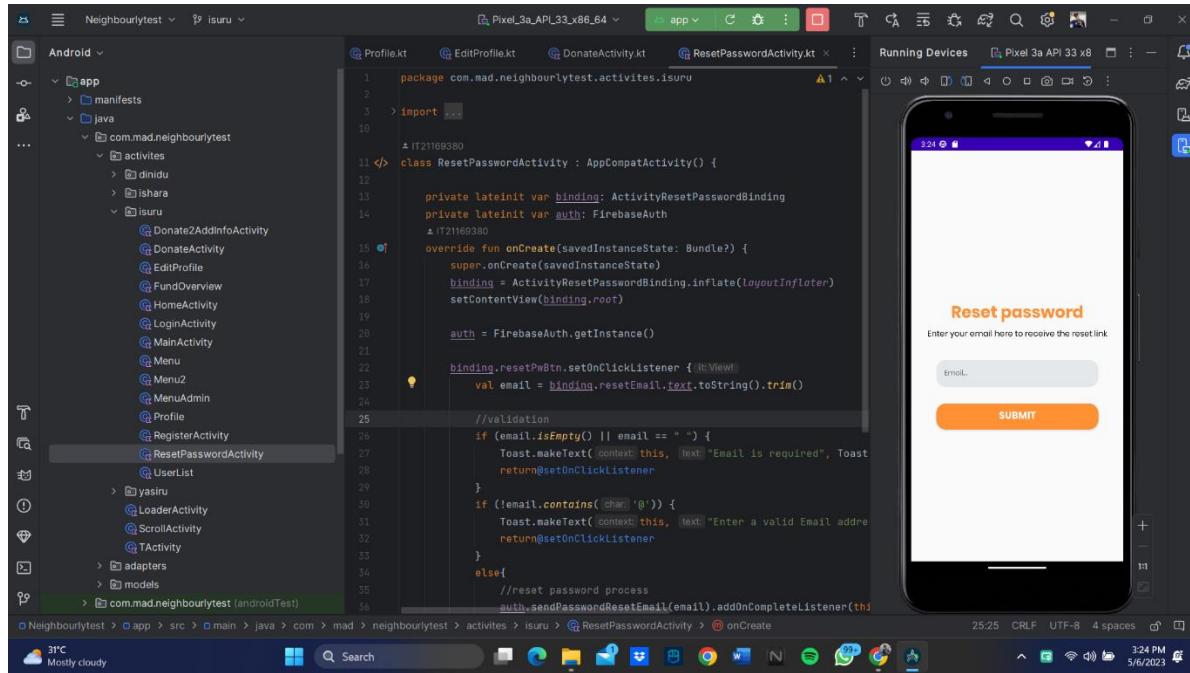
Signup :



Login :



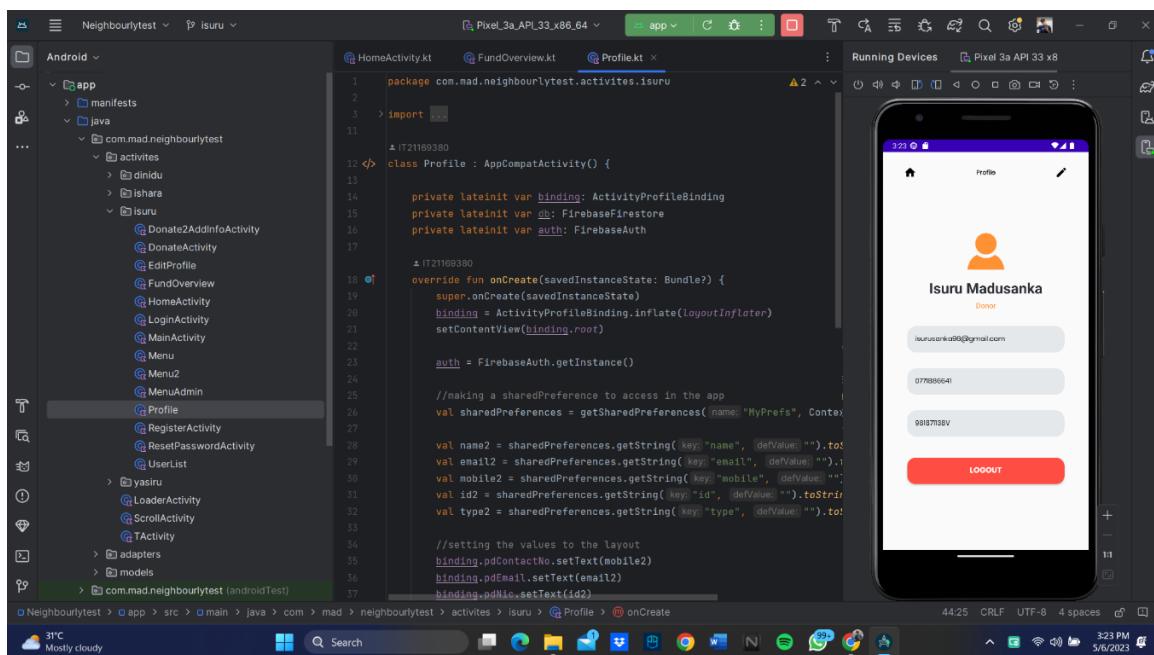
Reset password :



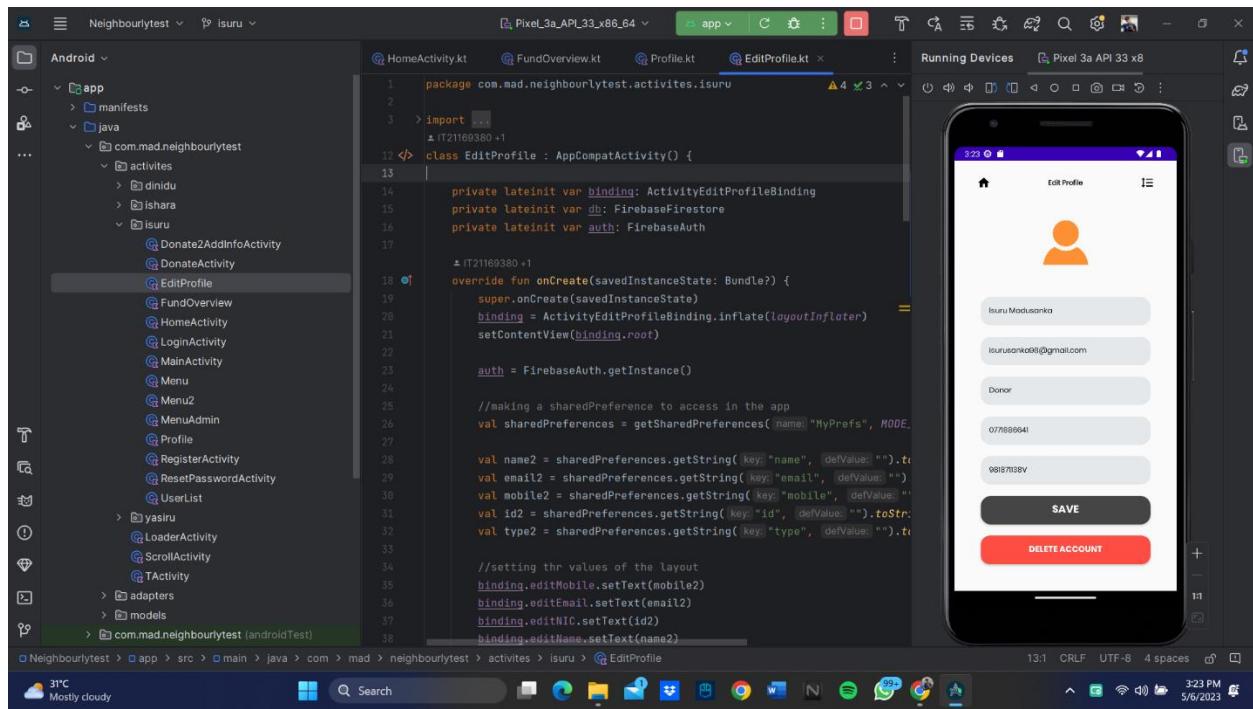
Home Page

Once a user has properly logged in to the system, they will be directed to the home page. The home page is unique to each user and displays their name at the top. There is also a profile button that users can click to view their profile. From this page, users can log out, update their details, and delete their account.

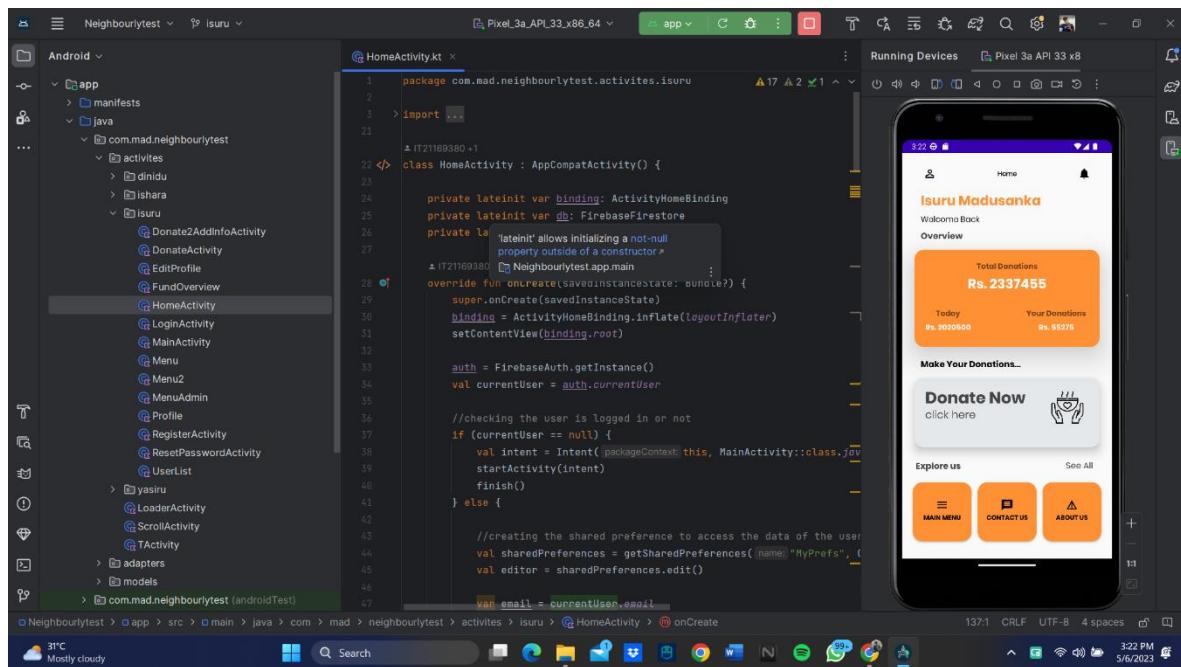
Profile :



Edit profile :



Home Page :



Fund Overview

The Fund Overview card in the home displays the following information:

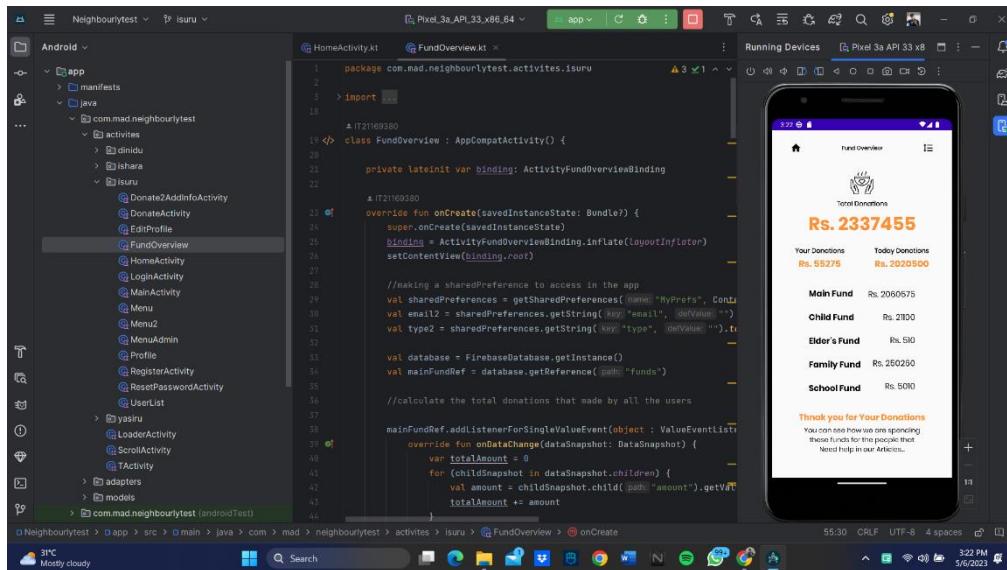
- Total donations: The total amount of donations made by all the donors in the system.
- Donations by logged-in user: The total amount of donations made by the logged-in user if they are a donor.
- User count: The total number of users in the system, including admins, donors, and volunteers if the logged-in user is an Admin.
- Today's donations: The total amount of donations made by all the donors on the current day.

Clicking on the Fund Overview card takes the user to a detailed page that displays the following information:

- Total funds: The total amount of donations made to each fund, including the main fund, child fund, elder fund, low-income family fund, and low facility school fund. This feature will display the sum of all donations made to each fund separately, so that users can view how much has been donated to each fund.
- Total donations: The total amount of donations made by all the donors in the system.
- Today's donations: The total amount of donations made by all the donors on the current day.
- Donations by logged-in user: The total amount of donations made by the logged-in user if they are a donor.
- User count: The total number of users in the system, including admins, donors, and volunteers if the logged-in user is an Admin.

When an admin is logged in, the Fund Overview page displays the total user count of the system, instead of the donation sum of the logged-in user. This is because an admin is not a donor, and hence, the donation sum is not applicable to them.

In addition to the Fund Overview page, the home page also displays a main menu with different options for admins, donors, and volunteers. This menu allows users to access different parts of the system based on their user type.

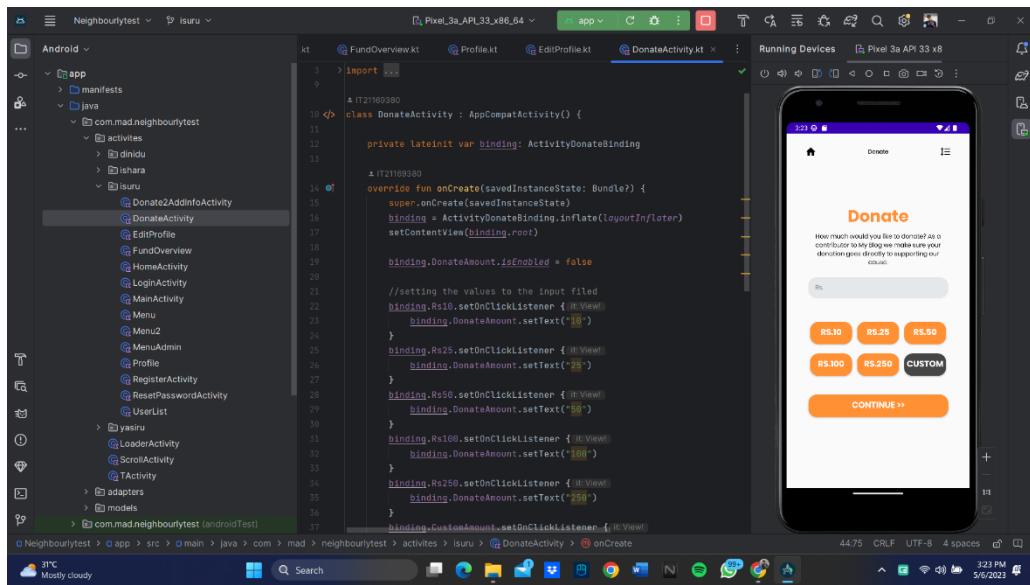


Money Donations

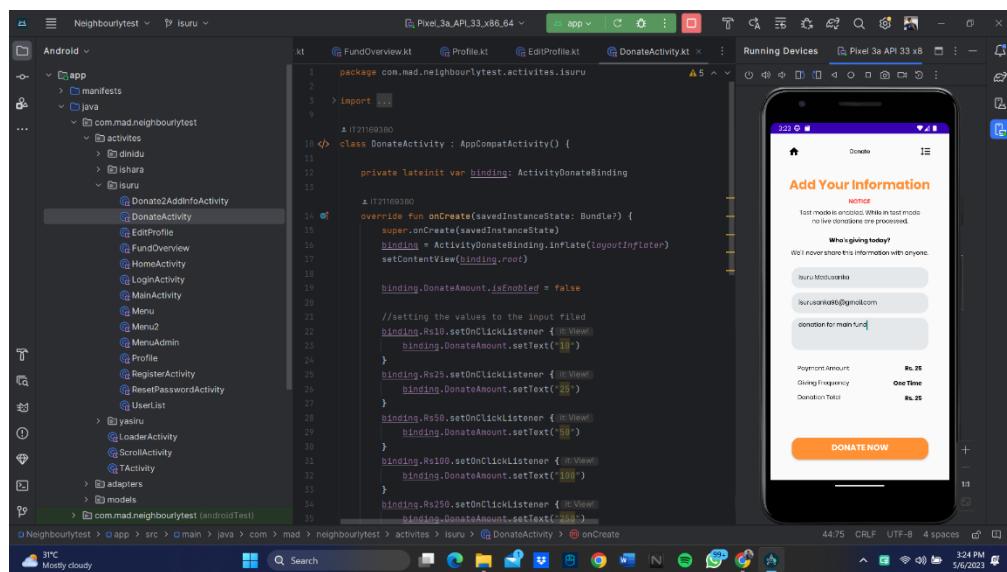
Users can donate to several funds such as main, child, elder, low-income family, and low facility school funds. Users can choose to donate to the main fund and directly donate the money to the main fund. They can also donate a system-provided donation amount or a custom amount as the donation amount. After that, users will need to provide some information about the donation. The system will automatically fill in some details from the user profile details.

Finally, the system is currently in the testing phase, and live donations are not being processed. However, the details of the donations made by donors are being stored in the Firebase Realtime Database for testing purposes.

Donation amount :



Add donation information :

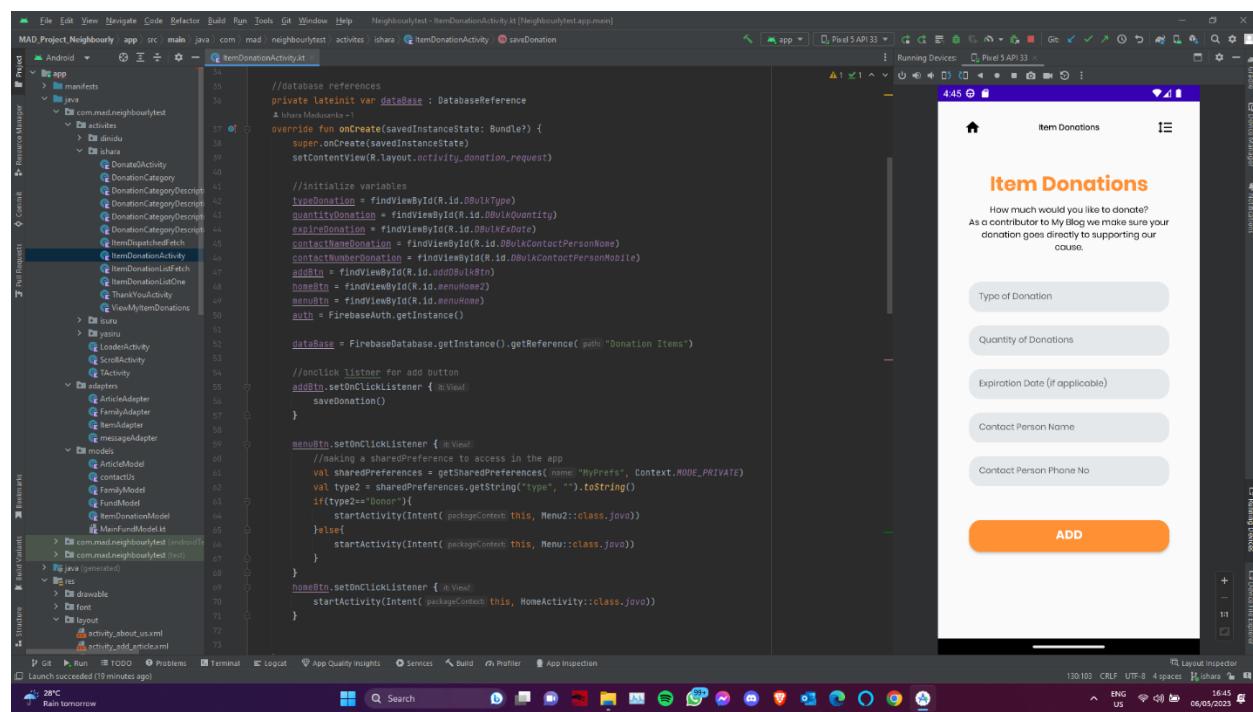


2. IT21189944 (Madusanka G.K.I) – Item Donations

Add Items

Donors can donate items such as food, school supplies, and medical supplies after logging into the system. The donation process includes providing the item's quantity, expiry date (if applicable), the donor's name, and contact information.

To ensure the accuracy and integrity of the information on donated items, all input fields and activates in the item donations system validated and displays alerts using sweet alerts. Following a successful donation, the system will save the logged-in user's email address and other provided information to the Firebase database. After the donation donors will be directed to the thank you page. Users can then return home by clicking the home button.



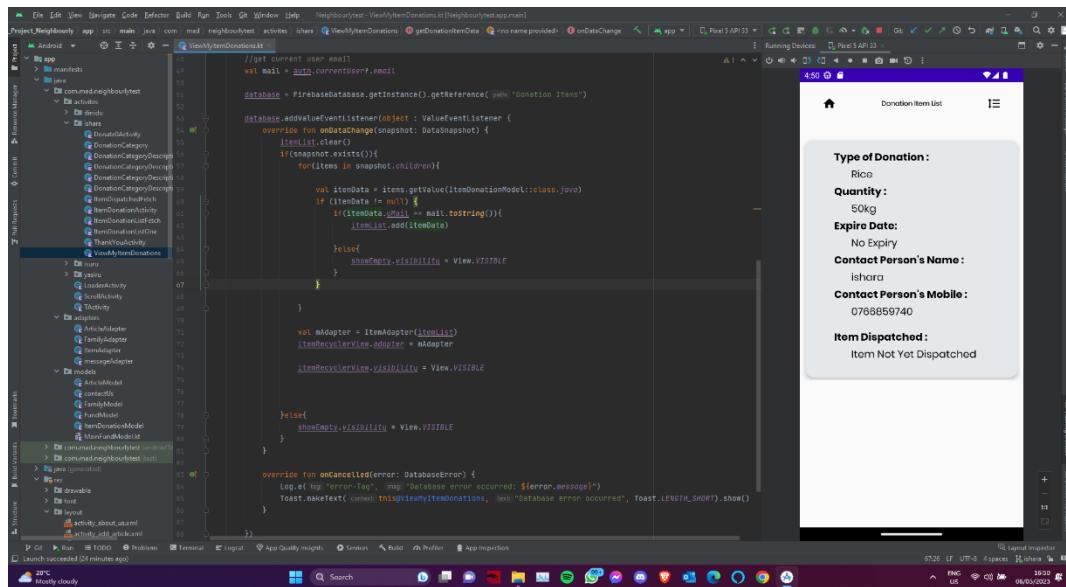
View Items

Users can also view their donated items in the "view my donations" section. This feature will show the user the items they donated.

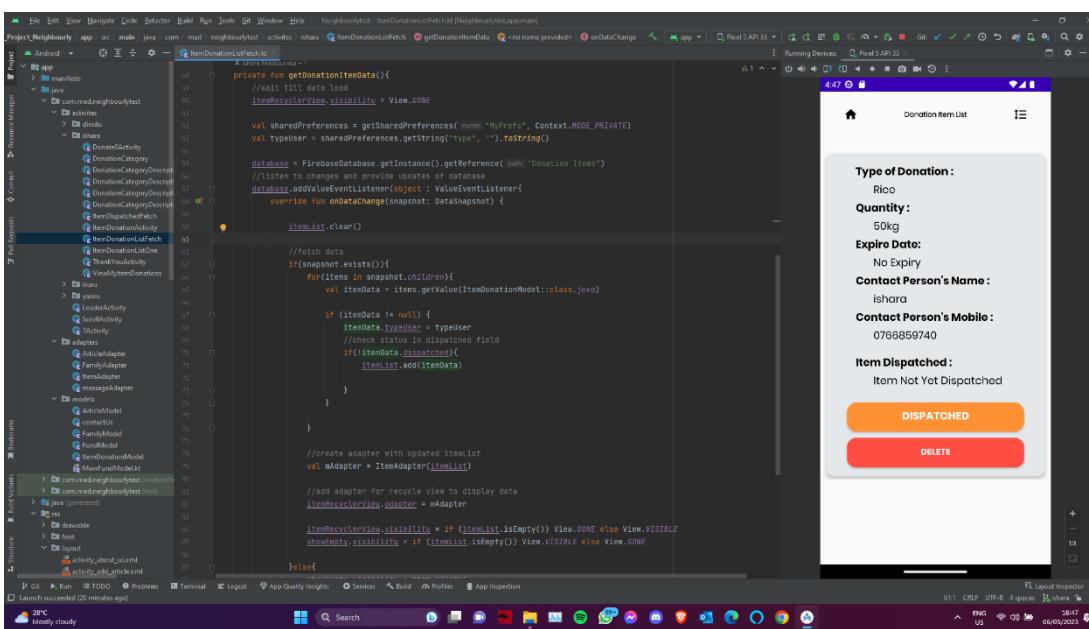
When retrieving data from the database, the system will check the expiry and dispatch status of each item. According to this system will update the status of item's expiration date and dispatch.

Volunteers can view all the items donated by users and perform actions such as deleting or dispatching donations.

Donor :

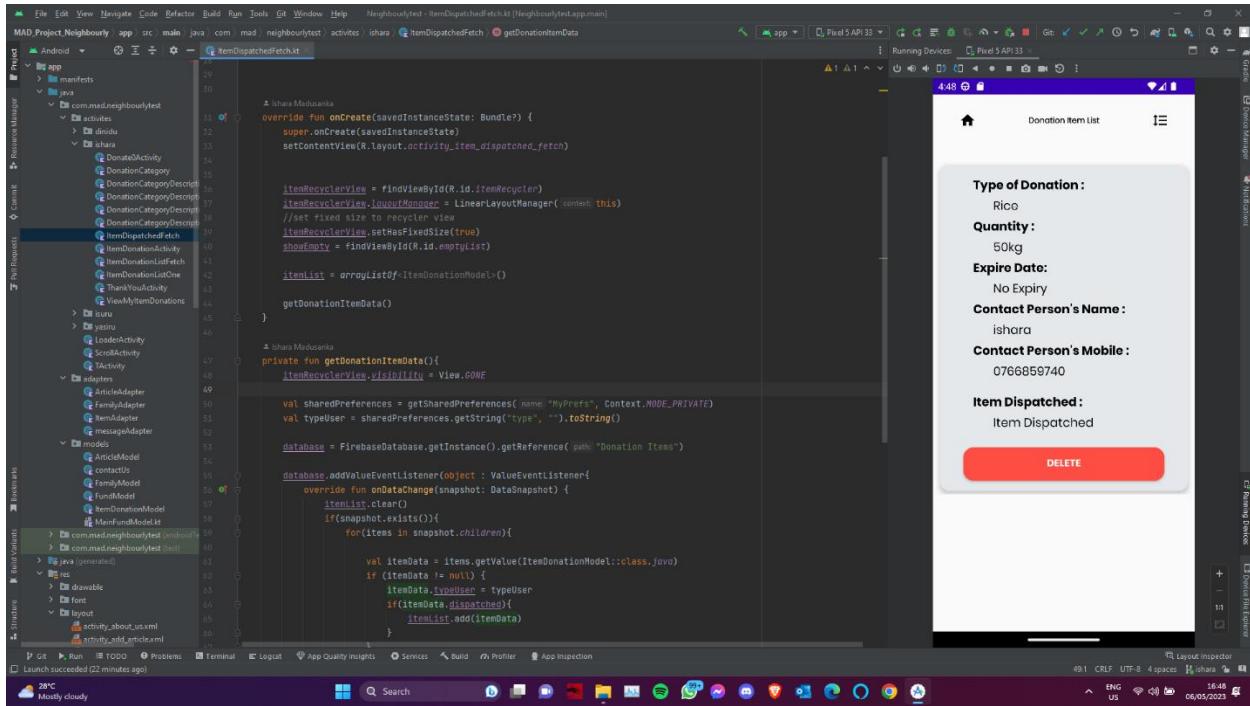


Volunteer:



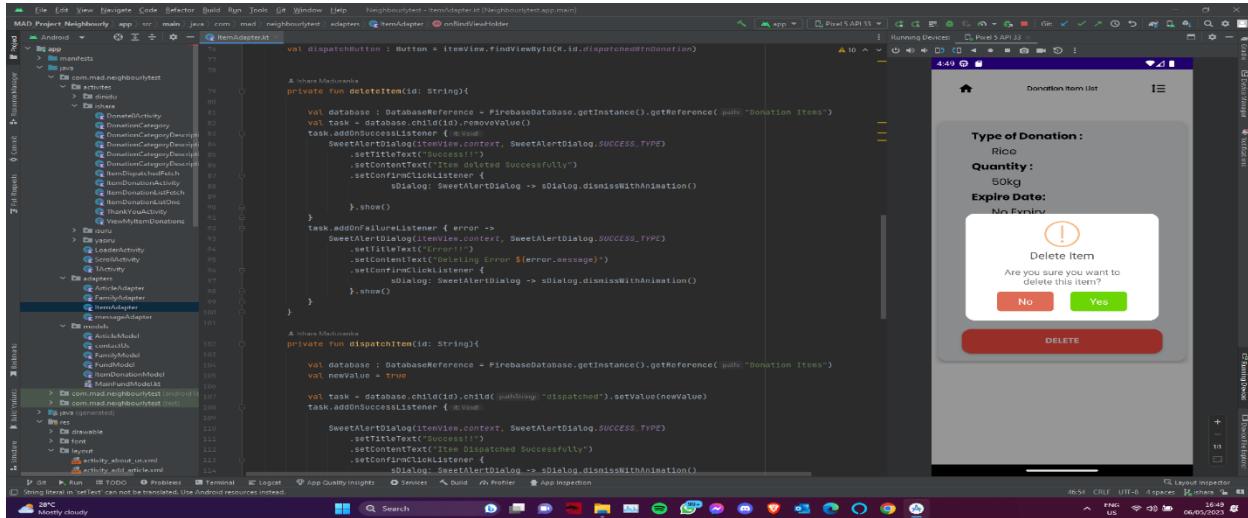
Update Items

A volunteer clicks the dispatch button, the item's dispatch status is updated in the database. When an item is dispatched, it is removed from the item list and appears as dispatched items in the dispatched items list.

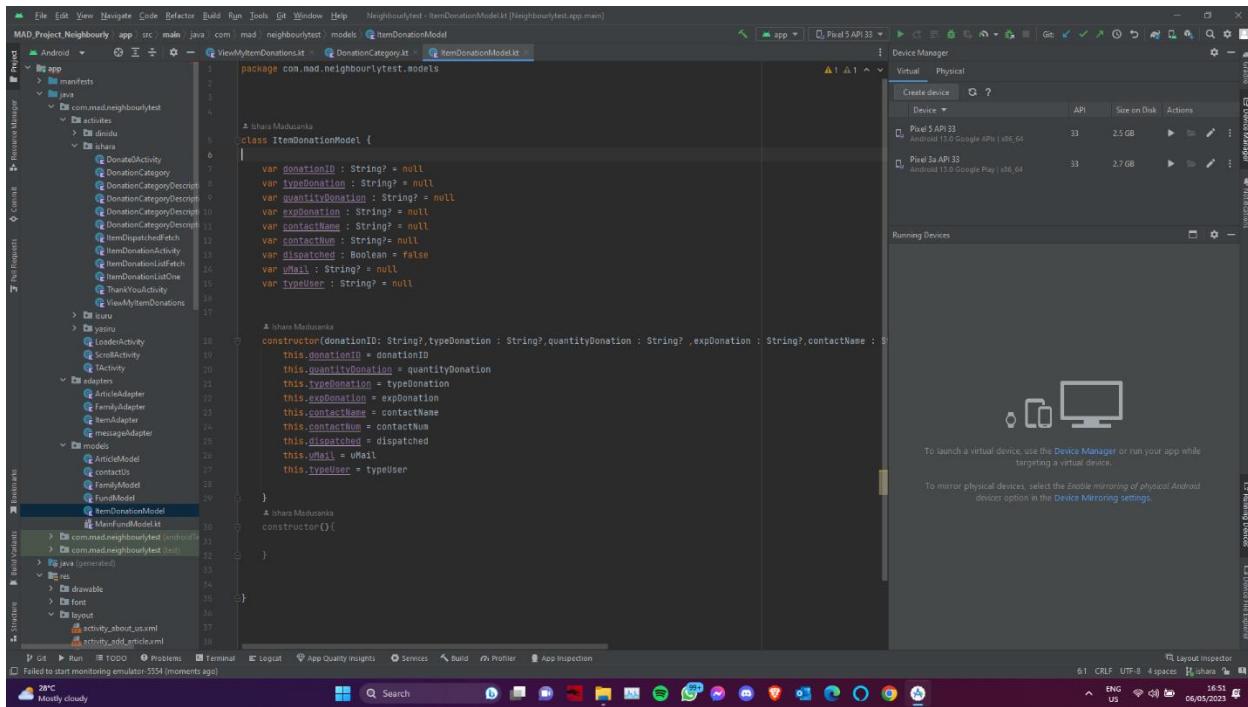


Delete Items

Volunteers can do delete item by clicking on the item's delete button that they want to delete in the "Item List" section and selecting the "Delete" option. This will remove the item's information from Firebase.



item Donation Model :



3. IT21186042(Ekanayaka E.M.D.K.L) – Family

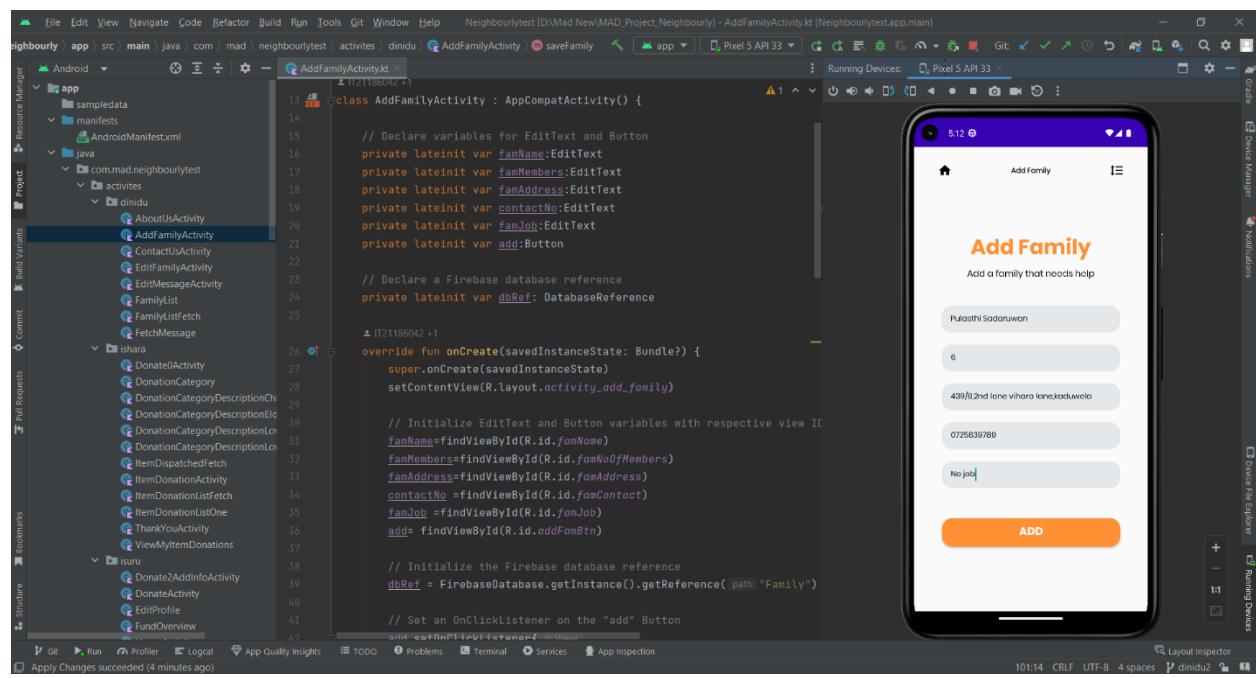
Family

Family section is providing a simple and intuitive way for volunteers to add, edit, and delete family information for those who need help. With user-friendly interface, volunteers can quickly input key details about families in need, such as their names, contact numbers, and specific needs. And with our firebase technology, this information is stored securely and made readily accessible to other volunteers.

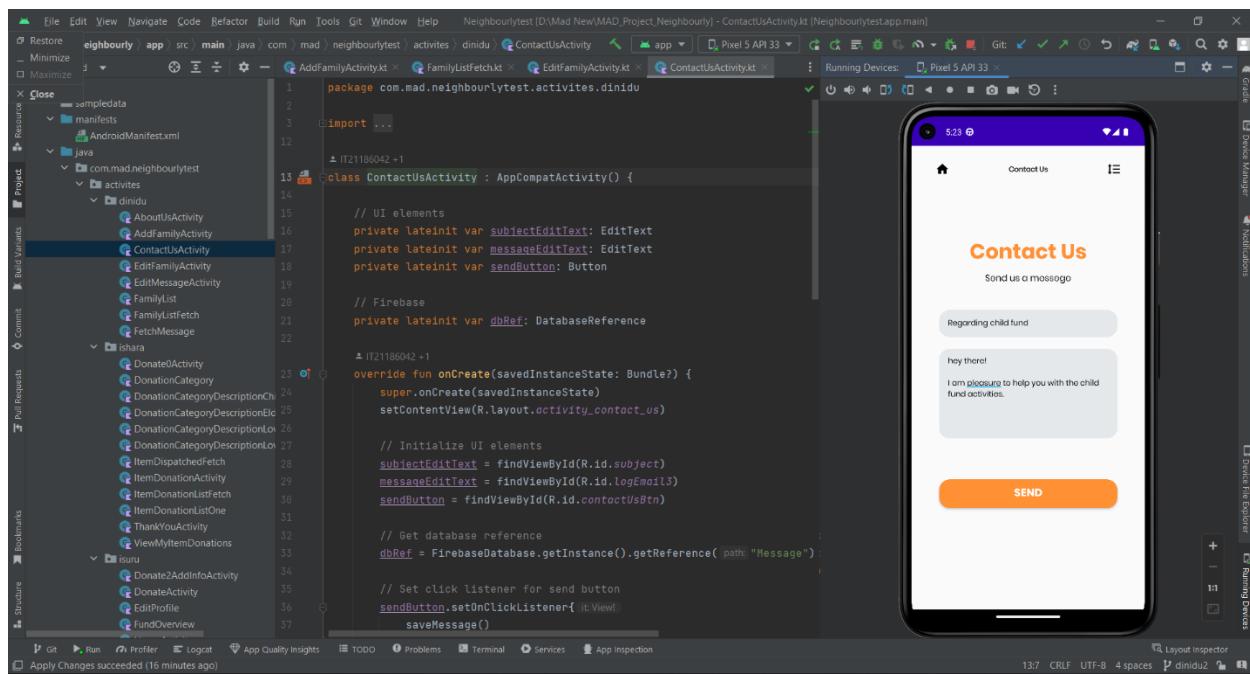
Adding family

Volunteers can add a new family to the Neighbourly Mobile App by navigating to the "Menu" section and clicking on the "Add Family" button. From there, they will be directed to a user-friendly form where they can input essential details about the family in need. This information includes the names and number of family members, their address, contact number, their job if they have. Once the form is complete, the information is securely stored in Firebase technology. This means that other volunteers and donors using the app can access the family's information and offer their support.

add Family:



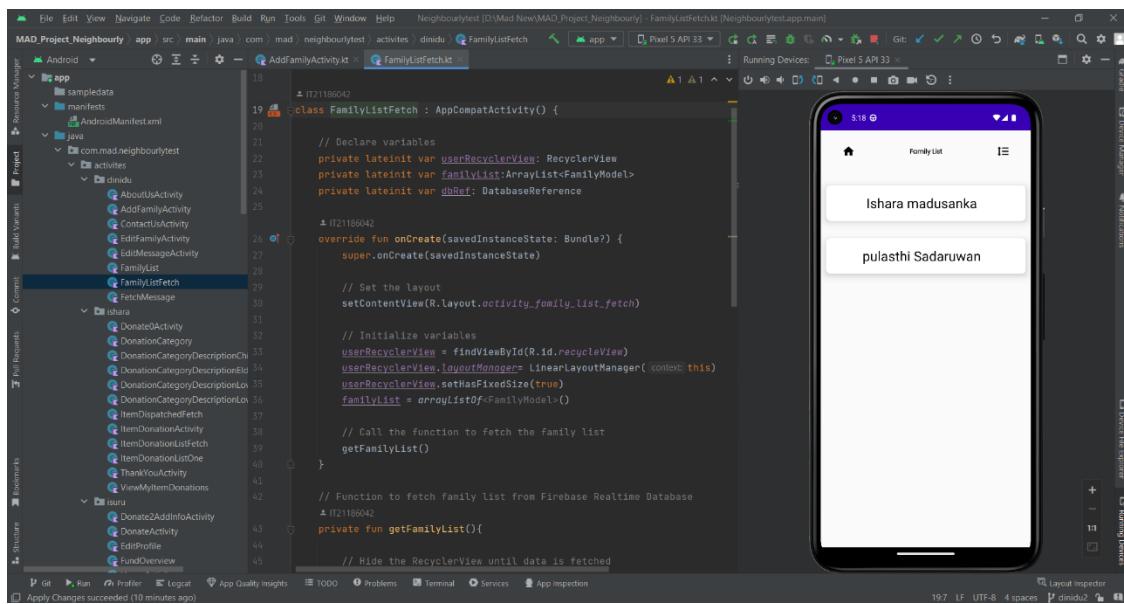
contact Us :



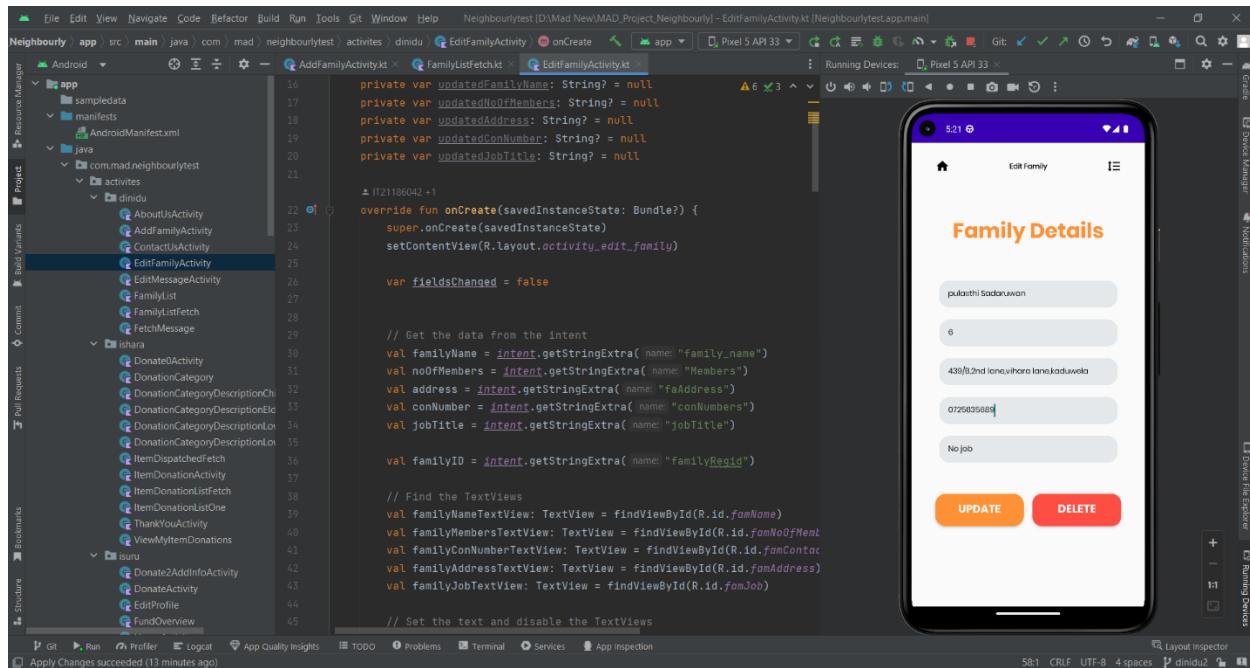
View and Update family details

Volunteers can edit an existing family's information by clicking on that family's name in the "Family List" section. This will bring up a detailed view of the family's profile, where the volunteer can make any necessary edits to the family's information (such as adding or removing family members, updating their family members, or changing their contact information). Once the edits are made, the updated information is stored in Firebase and can be accessed by other volunteers and donors who are using the app.

View Family :



Edit family :



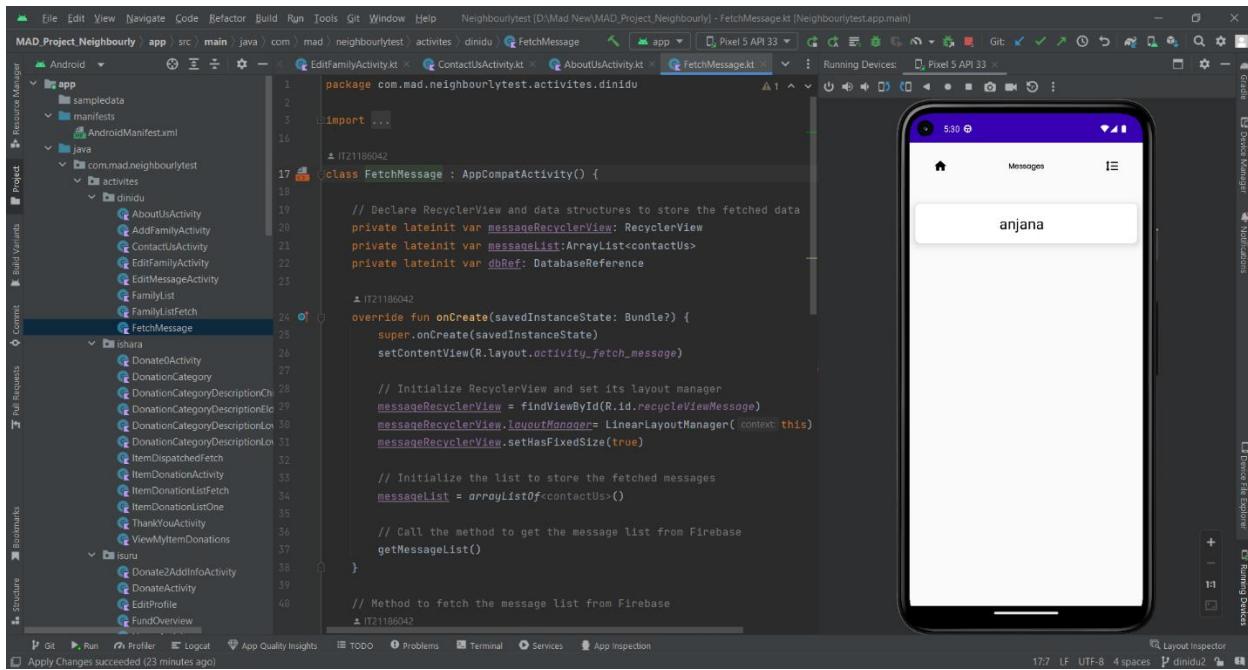
Delete family

In addition to adding and editing family information, the Neighbourly Mobile App Family Information function also allows volunteers to delete families from the app. Volunteers can do this by clicking on the family they want to delete in the "Family List" section and selecting the "Delete" option. This will remove the family's information from Firebase and make it inaccessible to other volunteers and donors using the app.

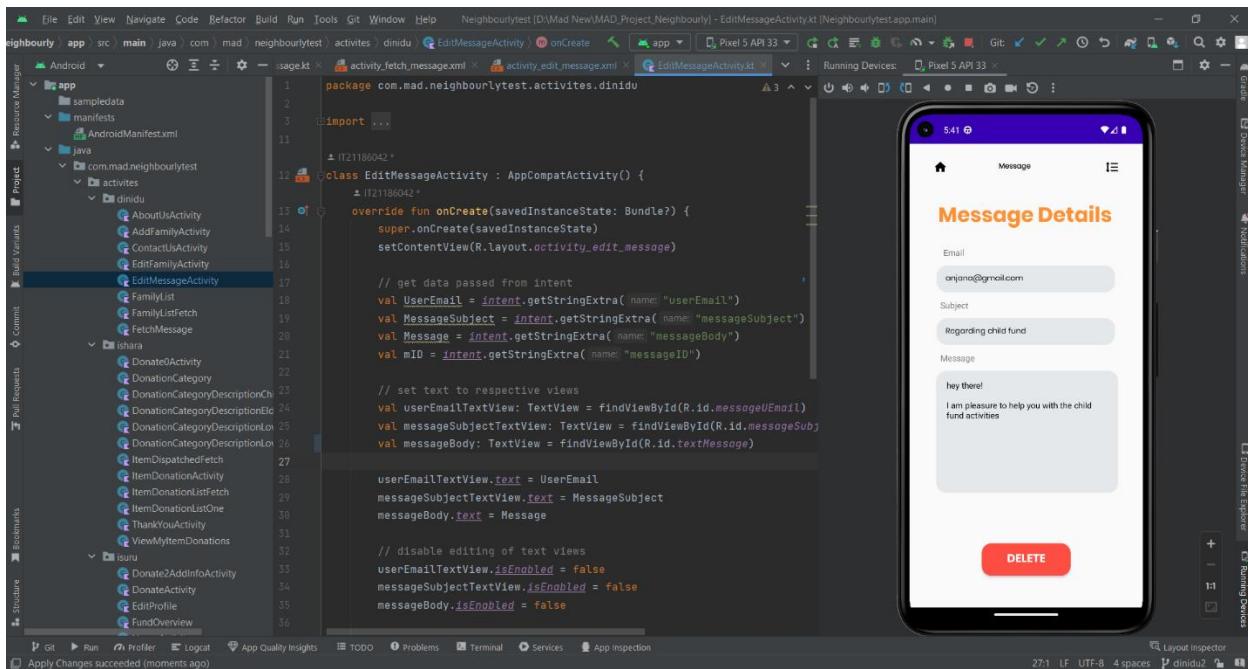
Contact Us :

The "Contact Us" feature is a valuable tool in the Neighborly Mobile App that enables donors and volunteers to send messages to admin team. The messages are stored in the "Messages" section, where the sender's name is displayed. By clicking on the sender's name, admin team can view the sender's email address, the subject line of the message, and the contents of the message. This function allows users to provide feedback, ask questions, or report any issues they may be experiencing with the app.

View Messages list :



Message details and delete :

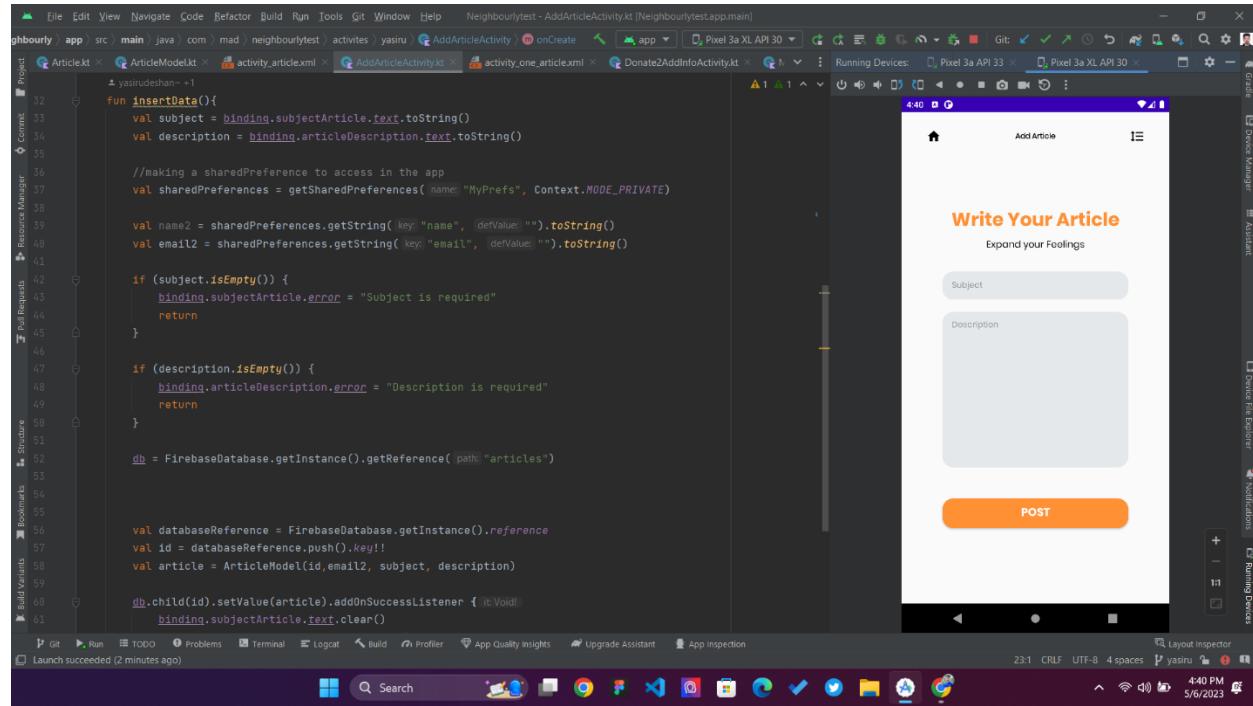


4. IT21169144(Karunarthne R.Y.D) – Article

The Neighbourly Mobile App's article management function provides a straightforward and efficient way for volunteers to add, retrieve, update and delete articles. The user-friendly interface enables volunteers to easily input key details about the articles in need, including the article title and description. The app uses Firebase technology to securely store and make the information readily accessible to other users.

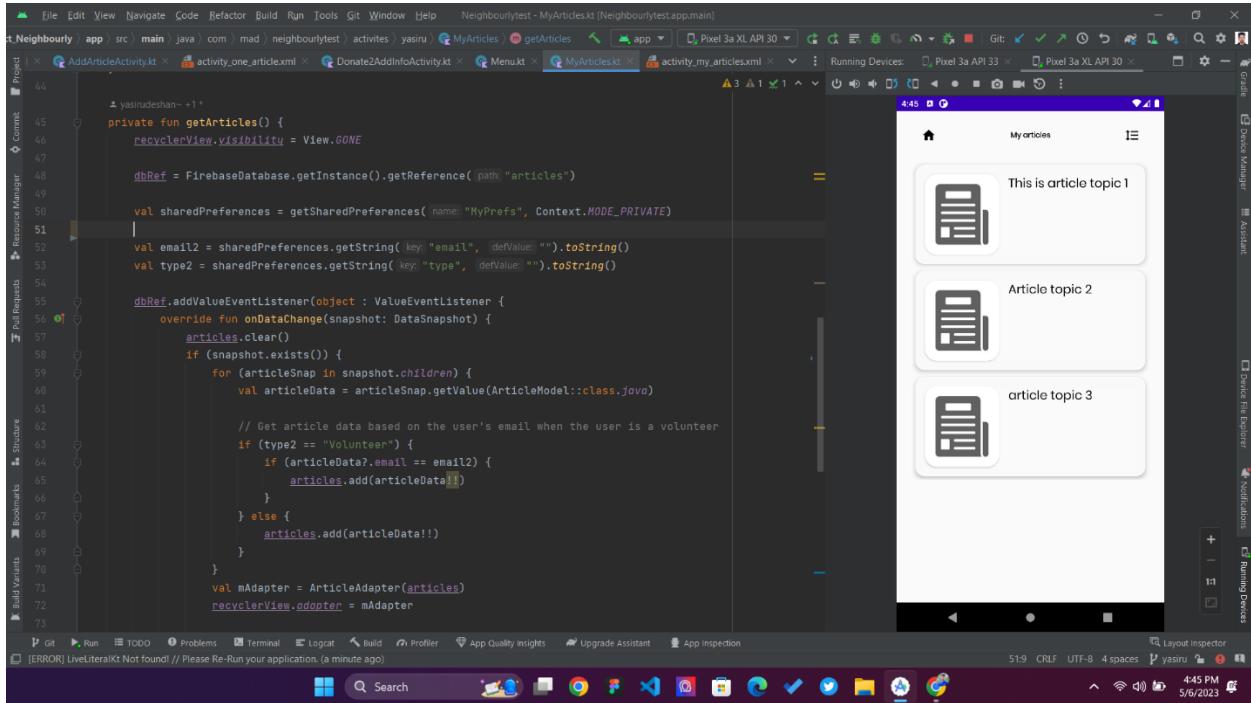
Add article :

To add an article, volunteers can navigate to the "Main Menu" section and click on the "Add Article" button. This directs them to a simple form where they can input essential details about the article. After submitting the form, the information is stored in Firebase, and donors and admins of the app can access the article's and read them.



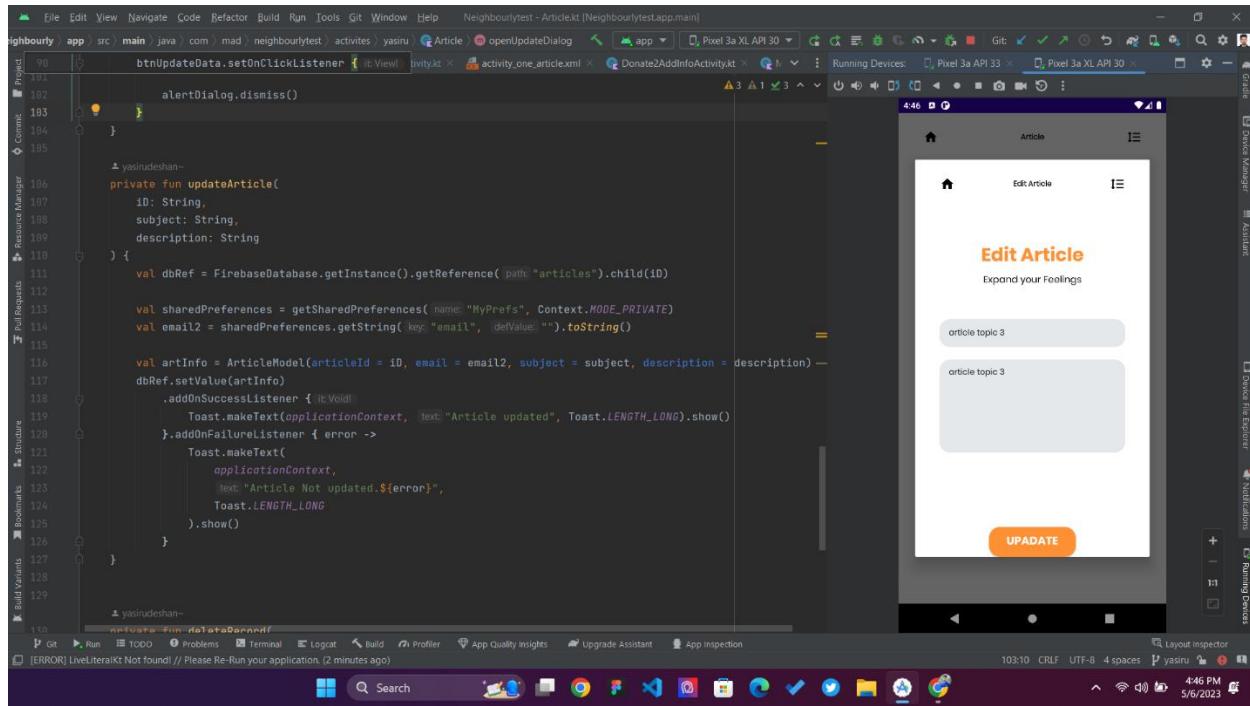
View Articles :

Volunteers can retrieve and view all articles using the recycler view. They can also edit an existing article's information to their added articles by clicking on the article's title in the "Articles List" section.



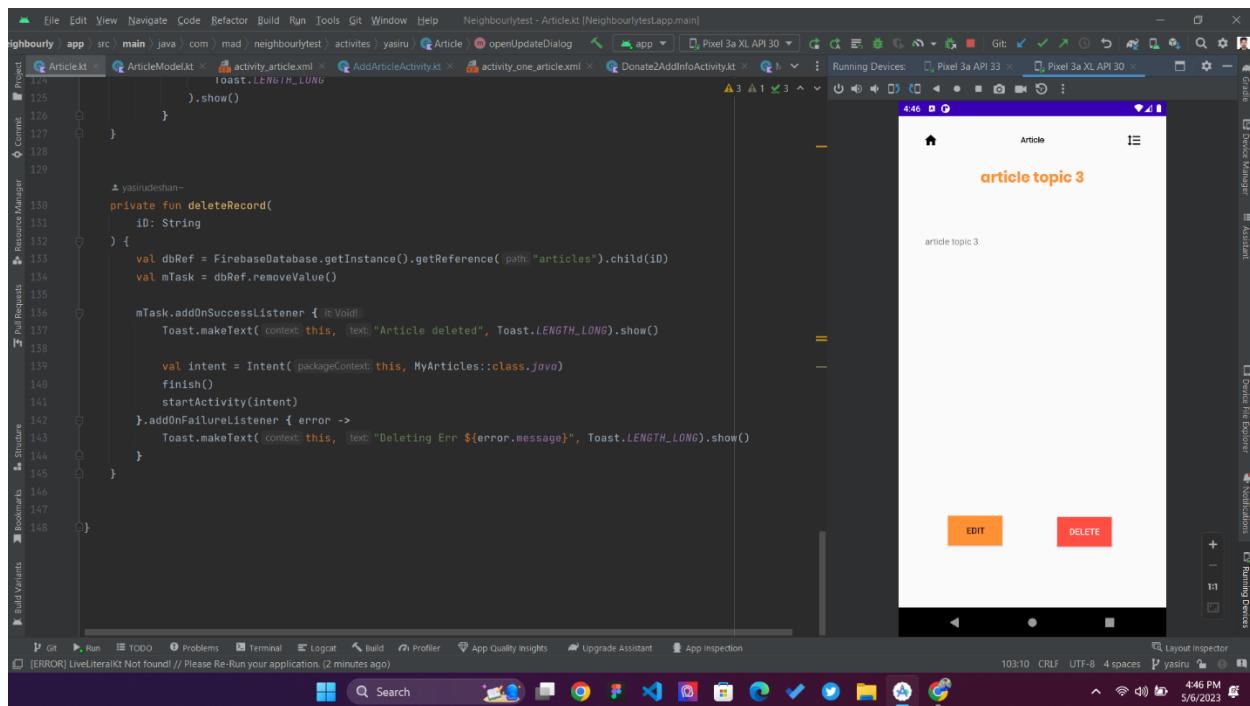
Edit Article :

Volunteers can make any necessary updates, such as modifying the article title and description. Once the edits are complete, the updated information is stored in Firebase and is accessible to donors and admins of the app.



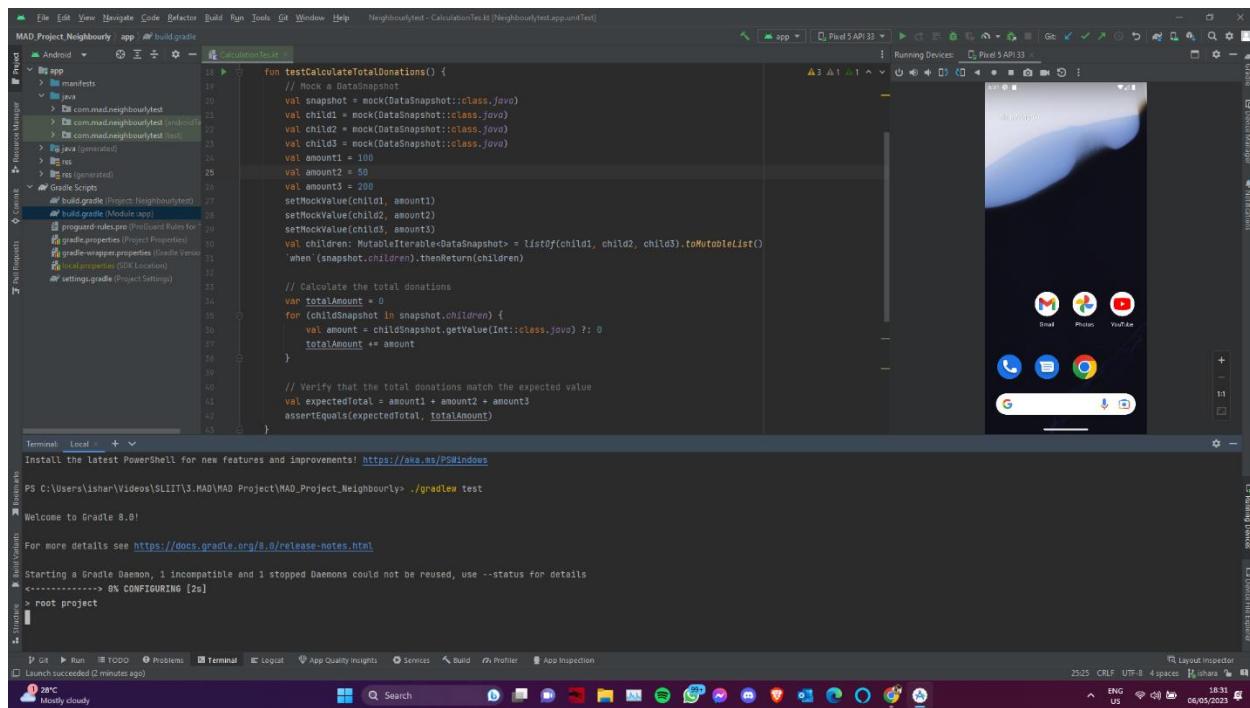
Delete article :

If volunteers want to delete an article, they can do so by clicking on the article they want to remove in the "Articles List" section and selecting the "Delete" option. This will remove the article's information from Firebase and make it inaccessible to other users of the app. With these features, the Neighbourly Mobile App's article management function provides a comprehensive solution for volunteers to add, retrieve, update, and delete articles, making it an invaluable resource for those in need.



UnitTest For Calculation Part

The code for testing is written for the testCalculateTotalDonations class and tests the function, which computes the total amount of donations made to a fund. The Mockito library is used in the test to mock a DataSnapshot object, which is a class from the Firebase Realtime Database SDK. The DataSnapshot object is typically used to read data from a database, but it is being mocked in this case to simulate the data that would be retrieved from the database. Beside this also used Log.d() function to check the values in the real code.



A screenshot of the Android Studio interface. The top navigation bar shows "File", "Edit", "View", "Navigate", "Code", "Refactor", "Build", "Run", "Tools", "Git", "Window", "Help". The title bar says "MAD_Project_Neighbourly - app [build.gradle]". The main area shows the code for "CalculationTest.kt":

```
fun testCalculateTotalDonations() {
    // Mock a DataSnapshot
    val snapshot = mock(DataSnapshot::class.java)
    val child1 = mock(DataSnapshot::class.java)
    val child2 = mock(DataSnapshot::class.java)
    val child3 = mock(DataSnapshot::class.java)
    val amount1 = 100
    val amount2 = 50
    val amount3 = 200
    setMockValue(child1, amount1)
    setMockValue(child2, amount2)
    setMockValue(child3, amount3)
    val children: MutableIterable<DataSnapshot> = listOf(child1, child2, child3).toMutableList()
    whenever(snapshot.children).thenReturn(children)

    // Calculate the total donations
    var totalAmount = 0
    for (childSnapshot in snapshot.children) {
        val amount = childSnapshot.getValue(Int::class.java) ?: 0
        totalAmount += amount
    }

    // Verify that the total donations match the expected value
    val expectedTotal = amount1 + amount2 + amount3
    assertEquals(expectedTotal, totalAmount)
}
```

The bottom terminal window shows the command: "PS C:\Users\lshan\Videos\SLIIT\3.MAD\Project\Neighbourly> ./gradlew test". The output includes "Welcome to Gradle 8.0!" and "Starting a Gradle Daemon, 1 incompatible and 1 stopped Daemons could not be reused, use --status for details". The status bar at the bottom right shows "25:25 CRLF UTF-8 4 spaces ENG US 18:31 06/05/2023".