

**A Project Topic**

**On**

**“House Hold Smart Share”**

**Submitted by:**

**MD Imran Hosen (18CSE019)**

**Date: May,2022**

**Department of Computer Science & Engineering**

**Bangabandhu Sheikh Mujibur Rahman Science and Technology University**

**House Hold Smart Share**

(This project is submitted in the partial fulfillment of the requirement for the project of "Second Year Second Semester" in Computer Science & Engineering)

By

**MD. Imran Hosen**

**ID: 18CSE019**

Supervised by:

**Mr. Nesarul Hoque**

Assistant Professor

Department of Computer Science & Engineering

Bangabandhu Sheikh Mujibur Rahman Science and Technology University

**Certificate of Approval**

It is hereby declared that the contents of this project original and any part it has not been submitted elsewhere for the award of any degree.

---------------------------------- ---------------------------------

Signature of the Candidate Signature of the Supervisor

Date: Date:

**Acknowledgement**

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them. I owe my deep gratitude to our project supervisor **Mr. Nesarul Hoque**, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system. I am thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of Computer Science and Engineering which helped me in successfully completing my project work.

**Md. Imran Hosen**

May, 2022

**Abstract**

The **House Hold Smart Share** application in Java is an interesting android application designed in the object-oriented programming language –Java, with the use of Android Studio Application. The application is an excellent example of utilization of Java and xml programming language in the field of application development. It demonstrates all the basic commands, syntaxes, functions, structures as well as concepts of file handling in Java and xml. In this application we can upload our old items in this application and those who need these items can request through this application. The main reason for making this application is that our old things do not turn into dirt and are useful for someone else to use. So that people benefit.

**Table of Contents**

[Chapter 1 1](#_Toc104388789)

[Introduction 1](#_Toc104388790)

[**1.1 Introduction** 1](#_Toc104388791)

[**1.2 Motivation** 1](#_Toc104388792)

[**1.3 Features** 1](#_Toc104388793)

[**1.4 Objectives** 2](#_Toc104388794)

[**1.5 Report Organization** 2](#_Toc104388795)

[Chapter 2 3](#_Toc104388796)

[Requirement Analysis 3](#_Toc104388797)

[**2.1 Functional Requirements** 3](#_Toc104388798)

[**2.2 System Requirements** 3](#_Toc104388799)

[Chapter 3 4](#_Toc104388800)

[System Design 4](#_Toc104388801)

[**3.1 Design Approach** 4](#_Toc104388802)

[**3.2 Detailed Design** 4](#_Toc104388803)

[**3.2.1 User Registration:** 4](#_Toc104388804)

[**3.2.2 Search item:** 4](#_Toc104388805)

[**3.2.3 Scan Location:** 5](#_Toc104388806)

[**3.2.4 Checkout and Order:** 5](#_Toc104388807)

[**3.2.5 Invoice Generation and Uploading:** 5](#_Toc104388808)

[**3.3 Application Design:** 5](#_Toc104388809)

[**3.4 User Interface Design:** 5](#_Toc104388810)

[**3.5 Database Design:** 6](#_Toc104388811)

[**3.6 Entity Relationship** 6](#_Toc104388812)

[**3.7 Methodology** 7](#_Toc104388813)

[Chapter 4 8](#_Toc104388814)

[Implementation 8](#_Toc104388815)

[**4.1 Programming Language:** 8](#_Toc104388816)

[**4.1.1 Java:** 8](#_Toc104388817)

[**4.1.2.1 Android Studio:** 9](#_Toc104388818)

[**4.1.2.2 Android Software Development Kit (SDK):** 9](#_Toc104388819)

[**4.1.2.3 Android Debug Bridge (ADB):** 9](#_Toc104388820)

[**4.1.2.4 Gradle Build:** 10](#_Toc104388821)

[**4.1.2.5 Android Device Monitor:** 10](#_Toc104388822)

[**4.1.2.6 SDK Manager:** 10](#_Toc104388823)

[**4.1.2.7 AVD Manager:** 10](#_Toc104388824)

[**4.2 Security and Permissions in Android:** 10](#_Toc104388825)

[**4.3 Application Maintenance:** 11](#_Toc104388826)

[Chapter 5 12](#_Toc104388827)

[Results and Discussions 12](#_Toc104388828)

[**5.1 User Interface Representation:** 12](#_Toc104388829)

[**5.2 Description of Features and the Approach:** 13](#_Toc104388830)

[**5.2.1 User Login/Registration:** 13](#_Toc104388831)

[**5.2.2 Home page:** 14](#_Toc104388832)

[**5.2.3 Navigation:** 14](#_Toc104388833)

[**5.2.4 Profile View:** 15](#_Toc104388834)

[**5.2.5 Search by location:** 15](#_Toc104388835)

[**5.2.6 Upload Product:** 16](#_Toc104388836)

[**5.2.7 My upload:** 16](#_Toc104388837)

[**5.2.8 My order:** 17](#_Toc104388838)

[**5.2.9 Update profile:** 17](#_Toc104388839)

[Chapter 6 18](#_Toc104388840)

[Future Scopes 18](#_Toc104388841)

[Chapter 7 19](#_Toc104388842)

[Conclusions 19](#_Toc104388843)

[References 20](#_Toc104388844)

**List of Figure:**

[**Figure 3.1: Firebase Database. 6**](#_Toc103718838)

[**Figure 5.1:Home Page 13**](file:///D:\Image\ss%20for%20project\Helicopter%20War%20Report(18CSE019).docx#_Toc103718839)

[**Figure 5.2: Login Page 13**](file:///D:\Image\ss%20for%20project\Helicopter%20War%20Report(18CSE019).docx#_Toc103718840)

[**Figure 5.3: Registration Page 13**](file:///D:\Image\ss%20for%20project\Helicopter%20War%20Report(18CSE019).docx#_Toc103718841)

[**Figure 5.4:MainPage 14**](file:///D:\Image\ss%20for%20project\Helicopter%20War%20Report(18CSE019).docx#_Toc103718842)

[**Figure 5.5: Main Page 14**](#_Toc103718843)

[**Figure 5.6:Navigation Bar 14**](#_Toc103718844)

[**Figure 5.7:Profile View 15**](#_Toc103718845)

[**Figure 5.8:Search by Location View 15**](#_Toc103718846)

[**Figure 5.9:Upload Product View 16**](#_Toc103718847)

[**Figure 5.10:Own Upload Product View 16**](#_Toc103718848)

[**Figure 5.11:My Order Request View 17**](#_Toc103718849)

[**Figure 5.12:Update Profile View 17**](#_Toc103718850)

# **Chapter 1**

# **Introduction**

Now we are living in the ocean of science and technology. This is possible for the purposes of programing. The main purpose of this project is to introduce programing everyone and practice them. In this, project design and implement a **House Hold Smart Share** app in Android Studio using java and xml language. In this app, the user can control this application easily and the user interface of the application is easy to use.

## **1.1 Introduction**

We made this application where we can upload our old items in this application and those who need these items can request through this application. The main reason for making this application is that our old things do not turn into dirt and are useful for someone else to use. So that people benefit.

## **1.2 Motivation**

We have so many things around us that we don't use them. I am motivated to see, how these items can be used. And that could be useful for some people.

## **1.3 Features**

Below describe about some features of our project.

* Easy interface
* Application can be use in android phone
* Upload any type of product
* Search product by your region

## **1.4 Objectives**

We made this application or project so that we could give our best in possible ways and show what we learned. The objectives of this project are:

* To use application in mobile.
* To make it user friendly.
* To provide an easy interface.
* To help people in their leisure time.

## **1.5 Report Organization**

For run this application we need some requirement which describe in the **Chapter-2: Requirements Analysis.** In the C**hapter-3: System Design s**how the design of this project and |**Chapter-4: Implementation** describe about the implementation of this project. After the implementation we get results which shown and describe in **Chapter-5: Results and Discussions**. We want to update our project in future which discuss In **Chapter-6: Future Scopes.** Finallyin **Chapter-7: Conclusions** we have added short discuss about our project.

# **Chapter 2**

# **Requirement Analysis**

In this chapter describe about the fundamental requirement for run this application also run android studio. For run this app we need an android phone also need internet connection. More details below describe.

## **2.1 Functional Requirements**

Functional requirements are properties that must exist in the final system. For any mobile application, we need to download the application from the play store. The application is free upon the store or merchant. To use the application, the user needs to register and login to the application after installing by providing login information. Once, he or she logins into the application, they can use all the features.

## **2.2 System Requirements**

The application should be installed into a device, system or any machine in such a way that it should have basic requirements like supporting software and hardware of the device, accessing in-built software, say camera for mobile device, internet permissions, and potential security issues such as virus or any malware detection.

# **Chapter 3**

# **System Design**

In this chapter explain about the system design of our project. How look our project every activity. We try our best for make design best. All the modules are designed, implemented and integrated together to make a flawless working application.

## **3.1 Design Approach**

This project is based on the functional design approach, which helps in understanding the design of the project in a simpler way by explaining its flow, use cases, and implementation more like a modular approach. For example, there are different modules in this project which have separate functionality and, other sub functionalities/modules. All the modules are designed, implemented and integrated together to make a flawless working application.

## **3.2 Detailed Design**

The detailed design including modules and sub modules of the application is as follows:

### **3.2.1 User Registration:**

If the user wants to use the House Hold Smart Share, they must download the application from the play store, install and register it by providing login information. Once, they register the registered information is stored on the server and can be validated, checking the valid credentials for the next time he logins with the application, also if user don’t want to registration an account, then the application could be usable by guest mode.

### **3.2.2 Search item:**

The search helps in finding whether the item is available in the store or not. This could help the users save time by searching for the item which is out of stock.

### **3.2.3 Scan Location:**

This feature helps people by allowing them to search location that is available on the item. Once the users search the location, they can see all the information about the item name. Users can scan any number of items they wish and keep adding to their physical add to cart. Later, they can request any item they want by checking the items in the cart.

### **3.2.4 Checkout and Order:**

Checkout is made in an easy step to avoid hassle in this application. The user can just check in with the checkbox from the Order. If the user wishes to request, they can proceed by clicking the “Request” button or they can cancel the purchase at this stage and proceed and with shopping for other items or they can exit the application.

### **3.2.5 Invoice Generation and Uploading:**

Once, the user confirms the purchase, the invoice is generated at the same time and we can share it instantly to cloud (share to drive, send email)

## **3.3 Application Design:**

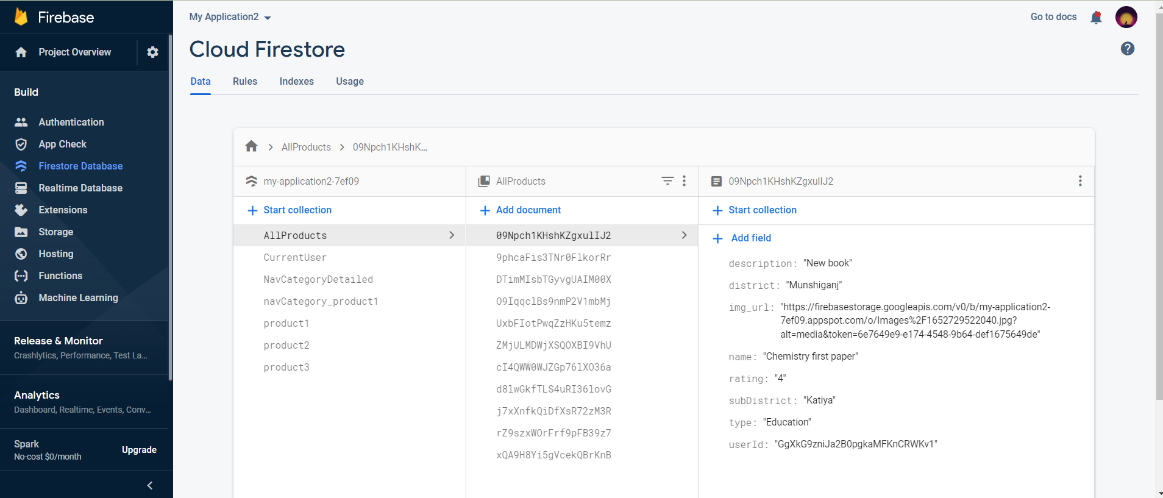
The main aim of the system design is to explain the scenario using use case diagrams. Use case diagrams clarify the flow of the application by deriving the use cases for all the functionalities in form of diagrams for the users.

## **3.4 User Interface Design:**

User Interface Design for any application should be very simple. We should have only a few clicks or navigation among the features when using the application to avoid hassle. In this application, there are two main screens, the Login and Home screens. The login page is the first page which appears when the user uses the application. In that page, if he is a new user, he can sign up or if he is an existing user, he can login with the credentials. He next screen is the homepage where the users can select features and use the app. The following image is the home screen. As seen from the image, this screen has all the key features.

## **3.5 Database Design:**

The database should be designed in such a way that it should be easy to access and manipulate [1]. Firebase Database definition and database manipulation operations should be performed accordingly to add, delete, and update values. In this project, we have used a Firebase database which is an open sources database, easy to install and use. The database server could be installed by providing user credentials such as username and password. The created database is hosted remotely and I am accessing the database using google date-base services. Web services act as the interface between the Android application and the database. As the Android application cannot connect directly with the database server, we are using Web Services [2]. The information to be transferred or fetched could be in XML or JSON formats. In this project, when we scan the item, the information is sent via web services in JSON format and displayed to users from Firebase database. The following figure, explains how the process takes place. Below the Figure-3.1 of the Firebase Database.



**Figure 3.1: Firebase Database.**

## **3.6 Entity Relationship**

There are three tables involved for this project under one database (products) which was created in Firebase Database. One table includes information about the items [3]. When the user scans the item, information from this table (item\_details) is fetched through web services. There are seven attributes or fields for this table which includes item name. Secondly, another table (user) has login and registration information about the users who sign up/sign in. It has five attributes including, user\_id, name, e mail, password, and phone number. The third table is for tracking purchases or expenses and it has five attributes namely, id, category, amount, date and notes.

## **3.7 Methodology**

Agile Software Development Methodology is widely used in many projects as it has many advantages. After gathering the project requirements, it is reviewed frequently in the form of small iterations and made into action by executing it. After completing tasks each iteration, it could be reviewed properly and moved to next iteration [4]. The main advantage of this methodology is that we can change the requirements or design even in the middle of the project when the situation arises. Also, code maintenance is easier in contrast to Waterfall Approach. In Waterfall methodology, there is no flexibility in changing the requirements when we develop the project because we must understand the working flow of the project at least 80% even before the start of the project and work according to that. Only if the design process is done, we can move to construction, testing, and support. Though this is not a team project, we have approached this project using agile methodology by applying its principles. To illustrate, for example: In one of the modules, “Scan an Item”, before start developing it, we have analyzed the design requirements initially and planned to execute them in the form of short iterations which are called “Sprints”. We tried to execute the planned tasks of the iterations accordingly, but We could not able to finish some of the tasks on that particular iteration and they are carried to next sprint and are called “Product Backlogs”. Also, as mentioned, we have changed some of the requirements for the convenience of the user interface while developing the project which is the main advantage of this methodology. Hence, for these reasons, we had decided to follow the principles of agile methodology.

# **Chapter 4**

# **Implementation**

In this chapter discuss about the implementation of our project. What programming language and application we use for build this android application.

## **4.1 Programming Language:**

Introduction to Programming Languages, IDE’S, Tools and Technologies used for this Implementation.

### **4.1.1 Java:**

As the project is developing an Android Application, the default programming language is Java. All Android applications are built using Java in Android Studio or Eclipse or both. Java is a popular and widely used language throughout the world. As mentioned in, Java is one of the powerful programming languages like C, C++. developed by Sun Microsystems which has many powerful features as described below. After the development of C, C++, Java has come into evolution by addressing their drawbacks. It is one of the open sources projects that could be easily installed in our machine. The language is also easy to learn, understand and implement. Java is used in various kinds of applications like Web, Desktop, Mobile, and Big Data [5]. Many powerful features are supported by Java including various libraries, application services, graphics library for 2D/3D applications. The language is flexible enough to maintain code complexity, test, implementation, integration and support. Apart from these, there are other key features which make Java more special. It is object oriented programming language, one of the important hierarchies in the programming languages which is used to implement real time applications, it provides for code reusability, it has a platform independence feature including any virtual machines(Write Once Read Everywhere), as in no need to write the 20 code for different OS as the Java Compliers convert the java source files to bytecode and this could be interpreted by any machine and the actual code is compiled irrespective of any machine, OS. It is more secured as the compilers are designed efficiently to figure out any kind of errors.

**4.1.2 IDE’s, Tools and Technologies:**

Below describe about the android studio, SDK also other tools and technologies we use in our project.

### **4.1.2.1 Android Studio:**

Android Studio is exclusively designed for developing Android applications. It consists of all Android SDK tools to design, develop, maintain, test, debug and publish our app [6]. The IDE is designed very efficiently which makes the developer’s job easy. It also supports the IntelliJ IDE, the main idea behind this IDE is that it automatically senses the variables, methods, classes, built-in functions or it could be anything else when we press the first letter of it. Say, suppose we declared few variables or methods that starts with an ‘S’, it automatically senses everything that starts with an ‘S’ and makes suggestions. It also supports Git as a version control system to maintain the app changes and push them into GitHub. All java files, layout files (for design) are integrated into a single project easily. After the completion of project, the whole application could be put as an .APK (Android Package) file, in which we can run that APK file in any device and use the application. Other main tools include Android SDK, ADB, and Gradle Build.

### **4.1.2.2 Android Software Development Kit (SDK):**

One of the main tools used in developing android applications, as it packages many core features into one SDK and it can be used in the application easily [7]. This helps us to avoid writing lot of code, and building applications faster.

### **4.1.2.3 Android Debug Bridge (ADB):**

Android SDK uses ADB tool as a connection device which allows us to connect the Android Devices or Emulator with the machine via USB [7]. After developing or while developing applications, we can connect with the device to check how the application runs. Later, we can debug and run the applications.

### **4.1.2.4 Gradle Build:**

Gradle Scripts are the recent feature that is added to Android Studio. It is basically an automated build system which is used to automate the various phases involved in designing an application that includes design, development, test, debug, and publish [7]. We need to configure the project and modules by mentioning all the supported jar files, SDK’s, version name, level, compiled SDK version, build tools version. to ensure that the developed app is compatible with the testing device/emulator. Gradle is also similar to Ant and Maven which helps in maintaining java projects (repositories).

### **4.1.2.5 Android Device Monitor:**

If we want to access all the hidden files that are generated when we run the application, we can use the monitor. We can select any project and explore the files that are related to that project. But, as they are hidden files, we need root permissions to access them. Suppose, if we run the app in device, we need to root the device and run commands in adb shell to get permissions.

### **4.1.2.6 SDK Manager:**

It is one of the main tools to maintain the updates of all the installed components required to run the project [7]. It also notifies us when the project is not compatible with device or any other compatibility issues and to download any component that is required.

### **4.1.2.7 AVD Manager:**

It is used to create virtual devices of any desired API level to support higher level SDK incase our device does not support. Using emulators to test the application is difficult as it might be little slower when compared to real device.

## **4.2 Security and Permissions in Android:**

Security notions in Android are quite high. Whenever a new Android Application is created, a unique user and group ID. This makes the maintenance of the application in an easier way to avoid any security or privacy issues [8]. As the application is created uniquely, it becomes private and no one can access other’s applications. Permissions are another important concept which is included in AndroidManifest.XML configuration file. This is required if the application wants to access the external features. For ex, if the application wants to access the Internet, Camera or it could be any feature, it requires permissions. It is included within the tags as it is an XML file. Permissions are automatically created for the basic applications at the time when we create the application. If the app uses higher level API or SDK. We must explicitly mention the permissions inside uses-permissions tag to access the features or components.

## **4.3 Application Maintenance:**

Apart from designing and developing the application, maintaining the application is one of the important characteristics. The developer/owner of the application should be concerned about the maintenance of the application by fixing the issues [9]. The solution for fixing such type of issues when the application crashes by using any of the “Crash Reporting Service” that are available for mobile applications. I have used “Crashlytics” reporting service in this application to track the issues when the user faces. Crashlytics is a lightweight reporting service for mobile applications including Android, iOS by providing respective SDKs. It also enables by analyzing the data in the form of reports which includes all the details about the number of issues faced, affected users because of the issues and finally indicating about the root cause of the crash by including the exact line in the source code of the application (error code) due to which the application might get crashed. The report analytic engine that is used by Crashlytics to perform real time crash reporting analysis is using “Fabric”.

# **Chapter 5**

# **Results and Discussions**

In this chapter discuss about the result of our project, the results and discussion may involve an evaluation of the design or method used. In a feasibility or case study, the results and discussion section would involve measuring the feasibility or evaluating the success of one or more solutions.

## **5.1 User Interface Representation:**

To make the application interactive, different controls have been used and designed using the layout file [10]. Following are the important controls that are designed and used in this application:

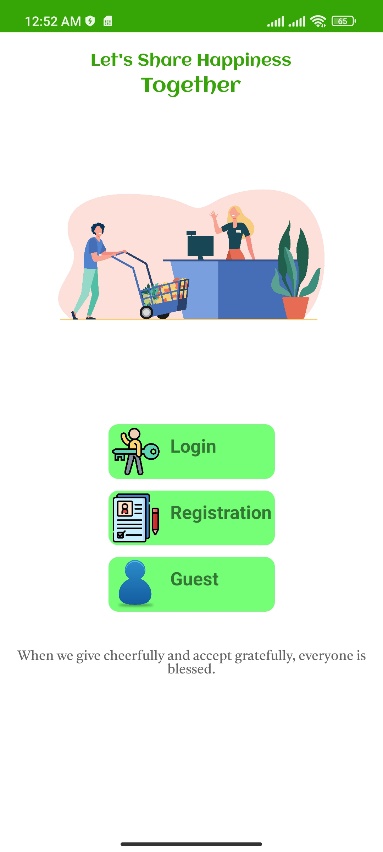
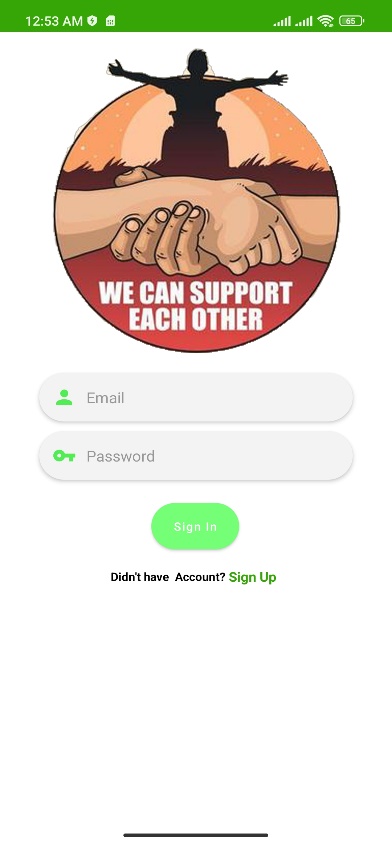
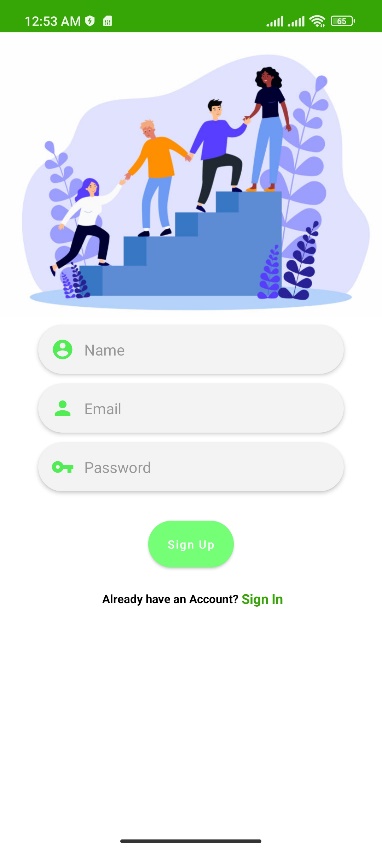
* Text View: The text view component belongs to the view group as a part of GUI. It displays the text or content view of any activity to the user and allows them to edit.
* Edit Text: This allows itself to be editable in the text box.
* Button: One of the important components in which the application needs. It is mainly associated with action when the user clicks it. We can represent the button using any text which holds the action class on it.
* Image Button: Suppose, if we want to have an image for the button which we have designed, we can include using this control by adding the source or path of the image file within the tags in the layout file
* List View: This is a key component under the view group which helps in displaying the information about anything when we click the action button. It also allows us to scroll through the screen and have a look about the information displayed. Using the list adapter, the content is pulled from the database.
* Checkbox: It is the control component which allows us to use or make use of the function by just clicking on the check box button. When we include check box widget in the application, we can see a small box in the screen on which we can check it and it will be selected.

## **5.2 Description of Features and the Approach:**

Below discuss about the features of this application. Also add some snapshot for better understanding features of this applications.

### **5.2.1 User Login/Registration:**

If the user wants to use the House Hold Smart Share, we must download the application from the play store, install and register it by providing login information. As shown in the Figure-5.1, the login information includes user name and password show in Figure-5.2. For the new user, the user must sign up by providing Full Name, Email Id and Password show in Figure-5.3. Once, he registers, the registered information is stored in the server and can be validated, checking for the valid credentials for the next time he logins with the application.



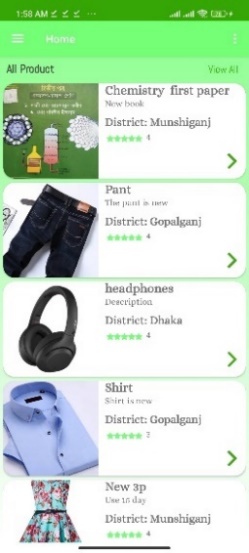
**Figure 5.3: Registration Page**

**Figure 5.2: Login Page**

**Figure 5.1: Home Page**

### **5.2.2 Home page:**

Below in Figure-5.4 and Figure-5.5 show the image of home page. In home page upper has a search option where we can search product, also have Popular item Card view, All Type Also food type Card view.

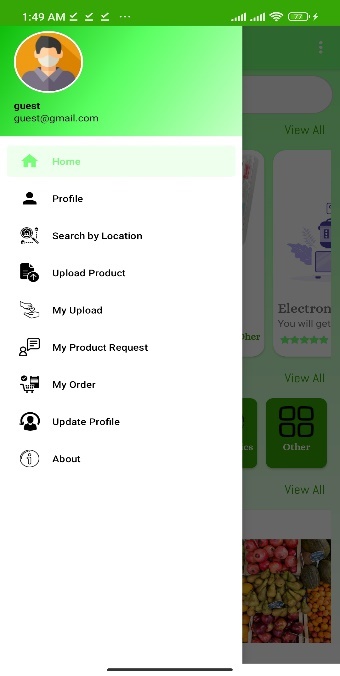


**Figure 5.4: Main Page**

**Figure 5.5: Main Page**

### **5.2.3 Navigation:**

This app also has navigation view. Below the image of navigation view in Figure-5.6.



**Figure 5.6: Navigation Bar**

### **5.2.4 Profile View:**

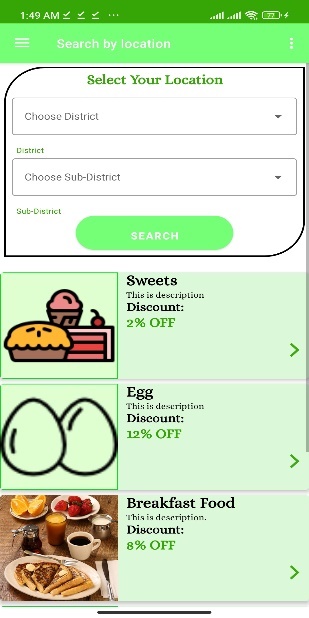
In the profile view we will show our profile details. Below the profile view fragment image in Figure-5.7.



**Figure 5.7: Profile View**

### **5.2.5 Search by location:**

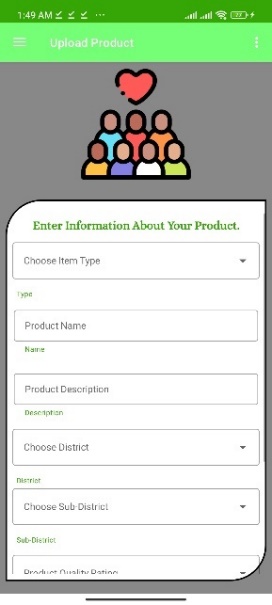
Using search by location all products sort by input location. Below the image of Search by location Figure-5.8.



**Figure 5.8: Search by Location View**

### **5.2.6 Upload Product:**

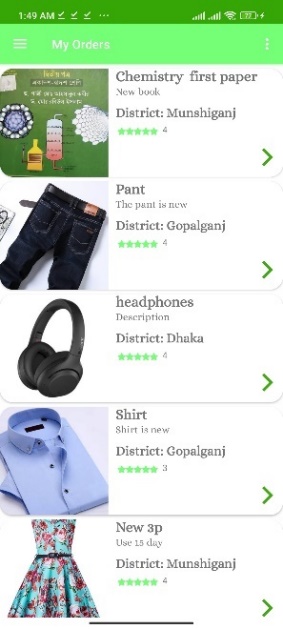
In upload product we can upload out product. Below the image of Upload product shown in Figure-5.9.



**Figure 5.9: Upload Product View**

### **5.2.7 My upload:**

In upload option we can see which product we already uploaded. Below the picture of my upload view show in Figure-5.10.



**Figure 5.10: Own Upload Product View**

### **5.2.8 My order:**

In order fragment will which product we requested. Below the picture of my order view. We can also cancel our order using the delete red button in Figure-5.11.



**Figure 5.11: My Order Request View**

### **5.2.9 Update profile:**

In update profile we can update our profile, like name, image, address, phone number etc. Below the picture of the update profile fragment in Figure-5.12.



**Figure 5.12: Update Profile View**

# **Chapter 6**

# **Future Scopes**

There are several features of the project that would be quite challenging to be added. The desirable features that House Hold Smart Share could have.

* Including visual tour (location of items) inside the store to find items quickly using Google Map and Indoor Map API. It will help us in saving time and being tired too on searching for the items.
* Another feature is allowing the application to read RFID tags which are attached along with the barcode tags in many clothing stores. The RFID tags could be read using NFC (Near Field Communication) which allows the users to exchange data between two devices and read tags. Also, this could be used for making payments by turning on the NFC feature in our devices and make reading the credit/debit cards. To use this feature, our device should support NFC feature and the store owners should also agree to take the RFID tags along with the items after we scan and buy the item as many of the tags are quite expensive. For cheaper tags, if they agree, we can implement this feature.
* Moreover, implementing this application in iOS platform as an iPhone App is one of the important considerations as we have many iPhone users.
* Add more security.
* Add chat system.

# **Chapter 7**

# **Conclusions**

We have learned a lot from this project on how to develop Android Application and publishing it in real time, use Web Services using Firebase, reporting using chart libraries, other libraries for scanning, payment, SDKs, requirement gathering, xml design, firebase database, data formats. As mentioned, other existing applications does not help in avoiding people to stand in a long checkout line, instantly searching about availability of products, tracking purchases (all together in a single application). If people use House Hold Smart Share in the future, they have several advantages which includes, easy checkout, sharing invoices instantly as it is quite easy to misplace organizing them in easily, particularly helpful for elderly people by avoiding them to wait for a long time in the checkout line, providing quick information about items available, upload products which helps users to budget their expenses wisely and finally used and unused products.

# **References**

[1] https://en.wikipedia.org/wiki/Database\_design

[2] https://firebase.google.com/docs/database/web/structure-data

[3] https://en.wikipedia.org/wiki/Relational\_database

[4] https://ecomputernotes.com/database-system/rdbms/phases-of-design-methodology

[5] https://en.wikipedia.org/wiki/Java\_(programming\_language)

[6] https://developer.android.com/studio/intro

[7] https://docs.expo.dev/guides/using-firebase/

[8] https://developer.android.com/guide/topics/permissions/overview

[9] https://developer.android.com/guide/components/services

[10] https://developer.android.com/training/basics/firstapp/building-ui