ONLINE NURSERY MANAGEMENT

PROJECT REPORT

Submitted By

P.SOWJANYA :R161165
V.KESAVI :R161521
V.JAYASREE :R161306

Under the supervision of
Mr. N Satyanandaram
Assistant Professor

Department of Computer Science and Engineering

As a part of

Partial fulfilment of the degree of Bachelor of Technology in Computer Science and Engineering

CERTIFICATE

This is to certify that the report entitled Online Nursery Management Application submitted by Vulavala Jayasree,P Sowjanya, Veerla Kesavi in partial fulfilment of the requirements for the award of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out by her under my supervision and guidance.

Mr.N Satyanandaram
Project Guide,
Assistant Professor,
Computer Science and Engineering,
RGUKT,RK Valley

Mrs.Rathnakumari,
Head of Department,
Assistant Professor,
Computer Science Engineering
RGUKT,RK Valley

DECLARATION

We are hereby declare that this report entitled “Online Nursery Management Application” submitted by me under the guidance and supervision of Mr.N Satyanandaram is a bonafide work. we also declare that it has not been submitted previously in part or in full to this University or other University or Institution for the award of any degree or diploma.

Date: 24-01-2022
Place:RK Valley

Vulavala Jayasree(R161306)
P.Sowjanya (R161165)
Veerla Kesavi(R161521)

ACKNOWLEDGMENTS

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 we would not forget to thank them.

We would like to express my sincere gratitude to **N Satyanandaram sir** for valuable suggestions and keen interest throughout the progress of my project.

At the outset, We would like to thank **Rajiv Gandhi University of Knowledge and Technologies, R.K Valley** for providing all the necessary resources for the successful completion of my course work.

With Sincere Regards ,
Vulaval Jayasree
P Sowjanya
Veerla kesavi

Table of Contents

1.ABSTRACT

As getting the information from various farmers in my village including my father and other sources we analysis that many peoples want to buy a plants and they directly concerned to nursery but sometimes people doesn’t know specific information about particular plant items as well Nursery owners are not technically skilled. So, in this case e-nursery is platform where Nursery owners can collaborate with us inorder to sell their palnts online.and Customers can buy the desired plants or seeds from a list pf available menu items.in case if customer desired plants are not available in menu items,Customer can place an order along with their contact details, grading specifications, special services, job codes, and amount of request.After an order is entered, an order confirmation report will be sent to the customer for review based on location of customer and location of nearby nursery and availability of plants requested by customer.So that they can save their Time. Customer service is extremely important. We want each customer to have a pleasant shopping experience,Continue to expand daily sales by adding to the variety of plants we sell.

2.INTRODUCTION

2.1 Android

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets. Android is developed by a consortium of developers known as the Open Handset Alliance and commercially sponsored by Google. It was unveiled in November 2007, with the first commercial Android device, the HTC Dream, being launched in September 2008.

It is free and open-source software; its source code is known as Android Open Source Project (AOSP), which is primarily licensed under the Apache License. However most Android devices ship with additional proprietary software pre-installed,most notably Google Mobile Services (GMS) which includes core apps such as Google Chrome, the digital distribution platform Google Play, and associated Google Play Services development platform.

Over 70 percent of Android smartphones run Google's ecosystem; some with vendor-customized user interface and software suite, such as TouchWiz and later One UI by Samsung, and HTC Sense.Competing Android ecosystems and forks include Fire OS (developed by Amazon) or LineageOS. However the "Android" name and logo are trademarks of Google which imposes standards to restrict the use of Android branding by "uncertified" devices outside their ecosystem.

The source code has been used to develop variants of Android on a range of other electronics, such as game consoles, digital cameras, portable media players, PCs, each with a specialized user interface. Some well known derivatives include Android TV for televisions and Wear OS for wearables, both developed by Google. Software packages on Android, which use the APK format, are generally distributed through proprietary application stores like Google Play Store, Amazon Appstore (including for Windows 11), Samsung Galaxy Store, Huawei AppGallery, Cafe Bazaar, and GetJar, or open source platforms like Aptoide or F-Droid.

Android has been the best-selling OS worldwide on smartphones since 2011 and on tablets since 2013. As of May 2021, it has over three billion monthly active users, the largest installed base of any operating system, and as of January 2021, the Google Play Store features over 3 million apps. Android 12, released on October 4, 2021, is the latest version.

Features of Android

Android is a powerful operating system competing with Apple 4GS and supports great features.They are

1.Beautiful UI

Android OS basic screen provides a beautiful and intuitive user interface.

2.Connectivity

GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.

3.Storage

SQLite, a lightweight relational database, is used for data storage purposes.

4.Media support

H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP.

5.Messaging

SMS and MMS

6.Web browser

Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.

7.Multi-touch

Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.

8.Multi-tasking

User can jump from one task to another and same time various application can run simultaneously.

9.Resizable widgets

Widgets are resizable, so users can expand them to show more content or shrink them to save space.

10.Multi-Language

Supports single direction and bi-directional text.

11.GCM

Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution.

12.Wi-Fi Direct

A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.

13.Android Beam

A popular NFC-based technology that lets users instantly share, just by touching two NFC-enabled phones together.Nursery owners can collaborate with us inorder to sell their palnts online. and Customers can buy the desired plants or seeds from a list

2.2 About Online-Nursery Application

Online-nursery is platform where Nursery owners can collaborate with us inorder to sell their palnts online. and Customers can buy the desired plants or seeds from a list pf available menu items.in case if customer desired plants are not available in menu items,Customer can place an order along with their contact details, grading specifications, special services, job codes, and amount of request.After an order is entered, an order confirmation report will be sent to the customer for review based on location of customer and location of nearby nursery and availability of plants requested by customer.So that they can save their Time.

2.3 Objective

The objective of this project is building an android application where customers can buy plants or seeds online.

2.4 Purpose

The purpose of this application both the customers and sellers can save their time.Here our main motto is to encourage small-scale nurserys and nurserys from rural areas.

3.SYSTEM REQUIREMENTS

3.1 HardwareRequirements

Processor : Duelcore1.6GHz

HDD : 250GB

RAM : 4GB

Mouse : OpticalMouse

3.2 SoftwareRequirements

OperatingSystem : UBUNU

CodingLanguage : JAVA

FrontEnd : Android UI

IDETools : AndroidSDK

4.LITERATURE SURVEY

ANDROID STUDIO

Android Studio is the official Integrated Development Environment (IDE) for Android app development.On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system

- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code

- Extensive testing tools and frameworks

JAVA:

The android platform allows us to write managed code using Java to manage and control the Android device.

GITHUB

GitHub is a Git repository hosting service.It is a web-based service.It is a file or code-sharing service to collaborate with different people. GitHub is a highly used software that is typically used for version control. It is helpful when more than just one person is working on a project.

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

Example: whenever a developer develops some project (like an app) or something, he/she constantly update it catering to the demands of users, technology, and whatsoever it maybe. Version control systems keep these revisions straight, storing the modifications in a central repository. It allows developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute.

Git is used to storing the source code for a project and track the complete history of all changes to that code, while GitHub is a cloud-based platform

built around the Git tool. So it’s necessary to upload your android project on GitHub.

Features of GitHub

GitHub is a place where programmers and designers work together. They collaborate, contribute, and fix bugs together. It hosts plenty of open

source projects and codes of various programming languages. Some of its significant features are as follows.

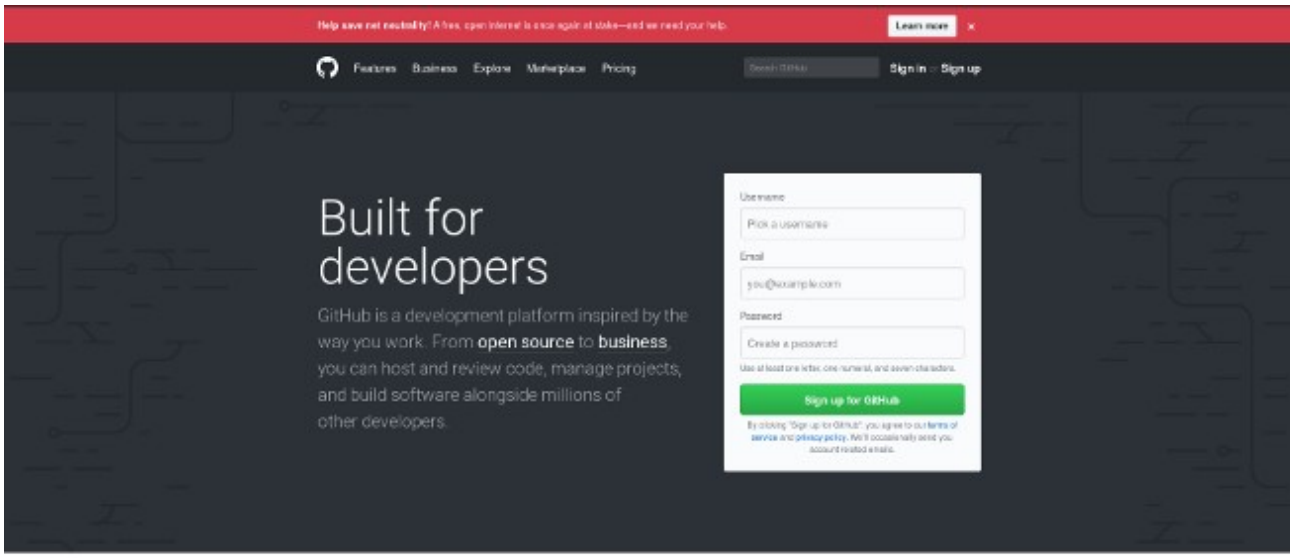
- Collaboration
- Integrated issue and bug tracking
- Graphical representation of branches
- Git repositories hosting
- Project management
- Team management
- Code hosting
- Track and assign tasks
- Conversations
- Wikisc

Benefits of GitHub

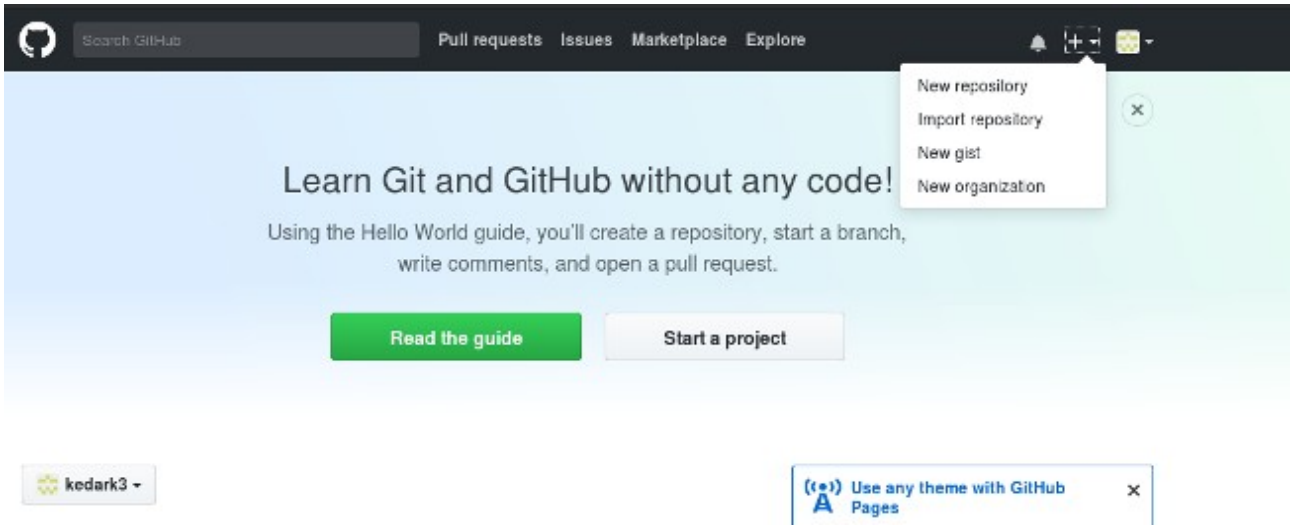
GitHub can be separated as the Git and the Hub. GitHub service includes access controls as well as collaboration features like task management, repository hosting, and team management. The key benefits of GitHub are as follows.

- It is easy to contribute to open source projects via GitHub.
- It helps to create an excellent document.
- You can attract recruiter by showing off your work. If you have a profile on GitHub, you will have a higher chance of being recruited.
- It allows your work to get out there in front of the public.
- You can track changes in your code across versions.
- GitHub provides you a beautiful visual interface which helps you to track or manage your version controlled projects locally.
- Once you register on GitHub, you can connect with social network and build a strong profile.

Step 1: Create a GitHub account

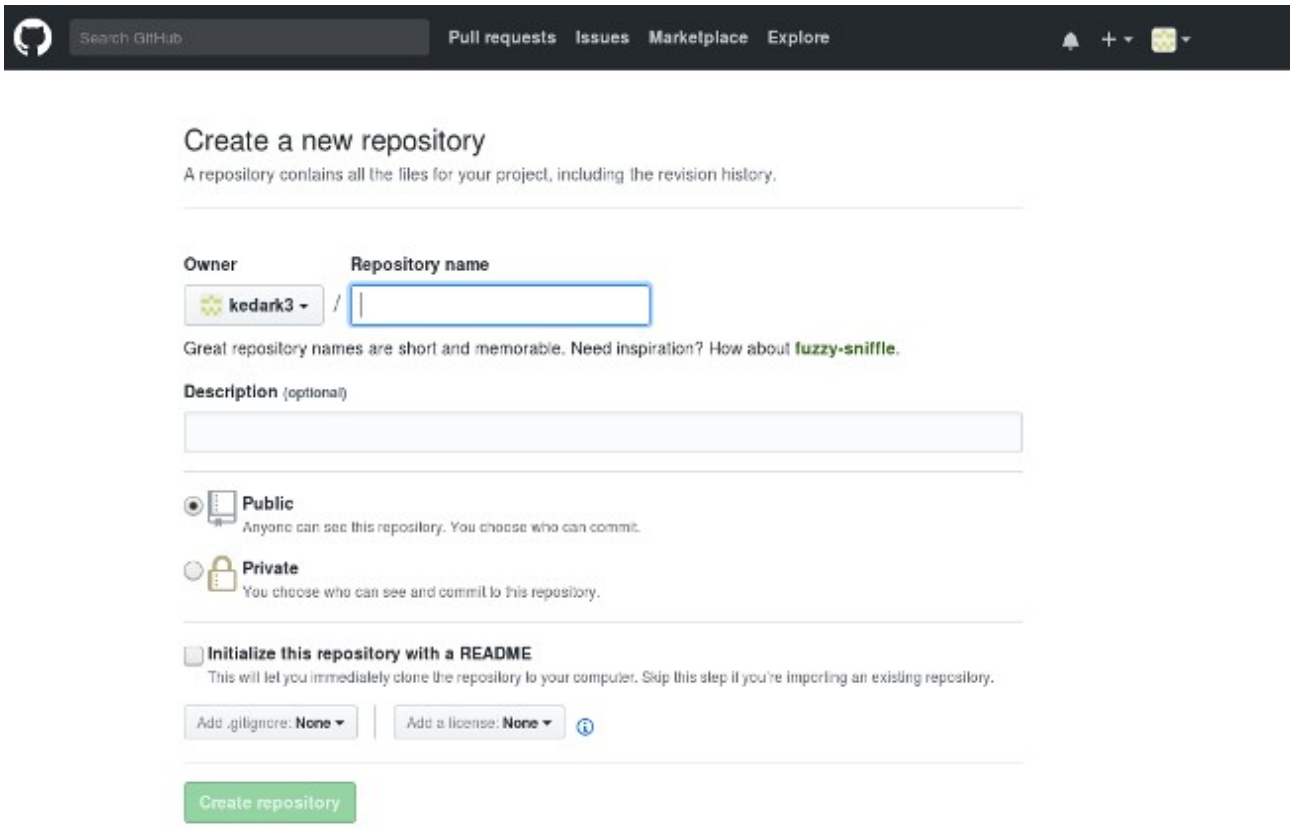


Pick a username (e.g., octocat123), enter your email address and a password, and click Sign up for GitHub . Once you are in, it will look something like this:



Step 2: Create a new repository

A repository is like a place or a container where something is stored; in this case we're creating a Git repository to store code. To create a new repository, select **New Repository** from the + sign dropdown menu (you can see I've selected it in the upper-right corner in the image above.



Enter a name for your repository (e.g, "Demo") and click **Create Repository**. Don't worry about changing any other options on this page.

Congratulations! You have set up your first repo on GitHub.com.

To tell your computer that Demo is a directory managed by the Git program, enter

1. **git init**
2. **git add README.md**
3. **git commit -m "first commit"**

4. **git remote add origin https://github.com/<your_username>/Demo.git**

That's it! You have created your first GitHub repo, connected it to your computer, and pushed (or uploaded) a file from your computer to your repository called *Demo* on GitHub.com. Next time, I will write about Git cloning (downloading your code from GitHub to your computer), adding new files, modifying existing files, and pushing (uploading) files to GitHub.

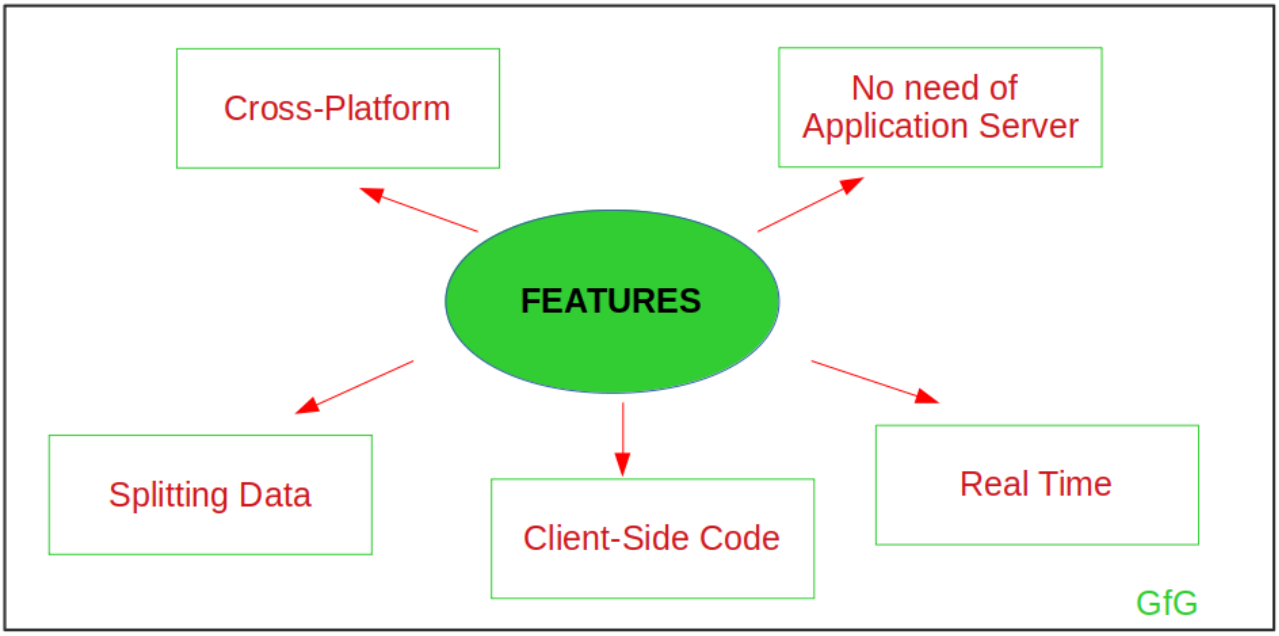
Latest Article on Github

https://github.com/ReScience/articles

FIREBASE

It is a real time database lets you to build collaborative applications Firebase is a backend platform for building Web, Android and IOS applications. It offers real time database, different APIs, multiple authentication types and hosting platform. This is an introductory tutorial, which covers the basics of the Firebase platform and explains how to deal with its various components and sub-components.

Firebase Features

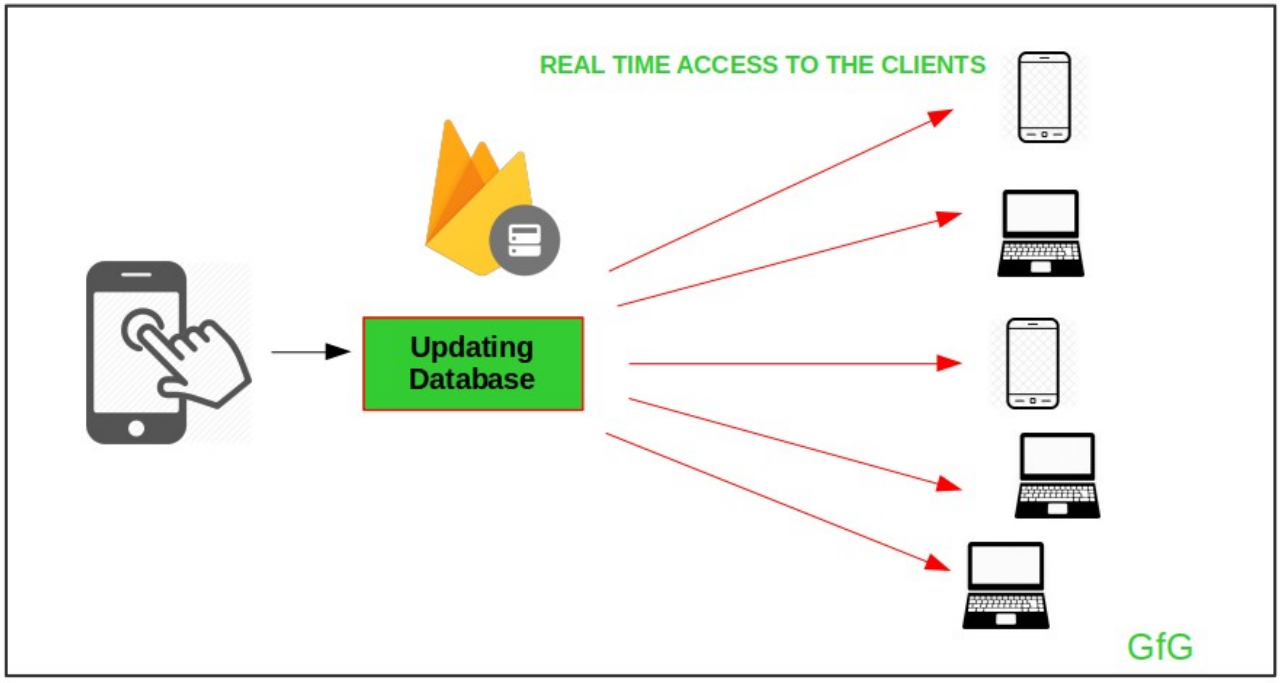


• **Real-time Database** − Firebase supports JSON data and all users connected to it receive live updates after every change.

• **Authentication** − We can use anonymous, password or different social authentications.

• **Hosting** − The applications can be deployed over secured connection to Firebase servers.

Firebase Realtime Database :-



The Firebase Realtime Databse is a cloud-hosted database.
Data is stored as Json and ssynchronized in realtime to every connected client.

➢ When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

➢ It is to store and sync data with our NoSQL cloud database. Data is synced across all clients in realtime, and remains available when

your app goes offline.

➢ **Realtime** : It is Instead of typical HTTP requests, the Firebase Realtime Database uses data synchronization—every time data changes, any connected device receives that update within milliseconds. Provide collaborative and immersive experiences without thinking about networking code.

➢ **Accesible from Client devices** : In our Application Firebase Realtime Database can be accessed directly from a mobile device. there's no need for an application server. Security and data validation are available through the Firebase Realtime Database Security Rules, expression-based rules that are executed when data is read or written.

➢ The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code

➢ Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud Platform. To learn more about the differences between database options, see Choose a database Cloud Firestore or Realtime.

➢ Firebase Remote Config stores developer specified key-value pairs to change the behavior and appearance of your app without requiring users to download an update.

➢ Firebase Hosting hosts the HTML, CSS, and JavaScript for your website as well as other developer-provided assets like graphics, fonts, and icons.

➢ Cloud storage stores files such as images, videos, and audio as well as other user-generated content.

Firebase Advantages

• It is simple and user friendly. No need for complicated configuration.

• The data is real-time, which means that every change will automatically update connected clients.

• Firebase offers simple control dashboard.

• There are a number of useful services to choose.

Firebase Connection to your application

• Click **Tools > Firebase** to open the **Assistant** window.

• Click to expand one of the listed features (for example, Analytics), then click the provided tutorial link (for example, Log an Analytics event).

• Click the **Connect to Firebase** button to connect to Firebase and add the necessary code to your app.

To create a Firebase project:

1. Go to the **FirebaseConsole.**

2. Click Add project, then select or enter a **Project name.**

    • If you have an existing Google project associated with your app, select the project from the **Project name** dropdown menu.
    • If you don't have an existing Google project, enter a new **Project name.**

3. (*Optional*) Edit the **Project ID.**

    Firebase automatically assigns a unique ID to your Firebase project. This identifier displays in publicly visible Firebase services forexample:

    • Default database URL — *your-project-id.firebaseio.com*
    • Default hosting subdomain — *your-project-id.firebaseapp.com*

Now that you have a project, you can add your Android app to it:

1. Click **Add Firebase to your Android app** and follow the setup steps. If you're importing an existing Google project, this may happen automatically and you can just download the config file.

2. When prompted, enter your app's package name. It's important toenter the package name your app is using; this can only be set when you add an app to your Firebase project.

3. During the process, you'll download a google-services.json file.

4. After you add the initialization code, run your app to send verification to the Firebase console that you've successfully installed Firebase.

Add the SDK

Add rules to your root-level build.gradle file, to include the google-services plugin and the Google's Maven repository:

Then, in your module Gradle file (usually the app/build.gradle), add the apply plugin line at the bottom of the file to enable the Gradle plugin:

```
buildscript {
    // ...
    dependencies {
        // ...
        classpath 'com.google.gms:google-services:4.2.0' // google-services plugin
    }
}

allprojects {
    // ...
    repositories {
        // ...
        google() // Google's Maven repository
    }
}
```

```
apply plugin: 'com.android.application'

android {
  // ...
}

dependencies {
  // ...
  implementation 'com.google.firebase:firebase-core:16.0.5'

  // Getting a "Could not find" error? Make sure you have
  // added the Google maven respository to your root build.gradle
}

// ADD THIS AT THE BOTTOM
apply plugin: 'com.google.gms.google-services'
```

You should also add the dependencies for the Firebase SDKs you want to use. We recommend starting with com.google.firebase:firebase-core, which provides Google Analytics for Firebase functionality...

AMAZON WEB SERVICES

AWS offers low, pay-as-you-go pricing with no up-front expenses or long-term commitments. We are able to build and manage a global infrastructure at scale, and pass the cost saving benefits onto you in the form of lower prices. With the efficiencies of our scale and expertise, we have been able to lower our prices on 15 different occasions over the past four years.

AWS provides a massive global cloud infrastructure that allows you to quickly innovate,experiment and iterate. Instead of waiting weeks or

months for hardware, you can instantly deploy new applications, instantly scale up as your workload grows, and instantly scale down based on demand. Whether you need one virtual server or thousands, whether you need themfor a few hours or 24/7, you still only pay for what you use.
AWS is a language and operating system agnostic platform. You choose the development platform or programming model that makes the most sense for your business. You can choose which services you use, one or several, and choose how you use them. This flexibility allows you to focus on innovation, not infrastructure.

5.FEASIBILITY REPORT

The feasibility of the project is analyzed in this phase and a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out.For feasibility analysis, some understanding of themajor requirements fort he system is essential.They are
1.ECONOMICALFEASIBILITY
2.TECHNICALFEASIBILITY
3.SOCIALFEASIBILITY

5.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the systemThe expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available.

5.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility,that is ,the technical requirements of the system. Any system developed must not have a high demand on the available technical resources.This will lead to high demands on the available technical resources.The developed system must have a modest requirement,as only minimal or null changes are required for implementing this system.

5.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user.

6.SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts,diagnosing problems and using the information to recommend improvements on the system. System analysis is a problem solving activity that requires intensive communication between the system users and system developers.System analysis or study is an important phase of any system development process. The system is viewed as a whole, the inputs are identified and the system is subjected to close study to identify the problem areas. The solutions are given as a proposal.The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

6.1 EXISTING SYSTEM

In any Online Nursery Management Application it is hard to provide fresh and pure natural plants to customers and it is hard to maintain the records of each and every plant.In Existing System they won't focus more on local nurseries near to villages so it may delay to supply plants on correct time and also they won't provide direct collabaration option for Nursery Owners and they focus more on gardening plants rather than agricultural crops

Disadvantages:

- More focus on gardening plants instead agricultural crops.
- Plants lose their freshness due to delay in supply

6.2 PROPOSED SYSTEM

The Proposed System mainly focuing on local farmers and local nurseries. Our Online Nursery Management system provides direct collabaration option to local nursery owners to expand their business i.e to sell their

plants in online.we are mainly focusing on agricultural crops instead of gardening plants.We take each order and supply order to the customers as soon as possible because we use local nurseries so,crops dont lose their freshness

Advantages:

- Direct collabaration of Nursery Owners to expand business
- Focus more on local nurseries and local farmers to make easy to farmers
- We supply fresh and natural plants to customers

6.3 FUNCTIONALITIES:

This application comprises of three modules, platform owners,product owners and delivery personnel nothing but customers.

A) Admin module:-

1. Admin first register and log in himself in to the Application.
2. He can view the order which can be added to card from the customer side.
3. He can send order confirmation message to the customer based on location of customer and location of nearby nursery and availability of plants requested by customer.
4.He can send confirmation message to the Nursery owners based on location of Nursery and availability of plants in the Nursery.
5. He can update the information by adding plants to menu.

B) Customer module:-

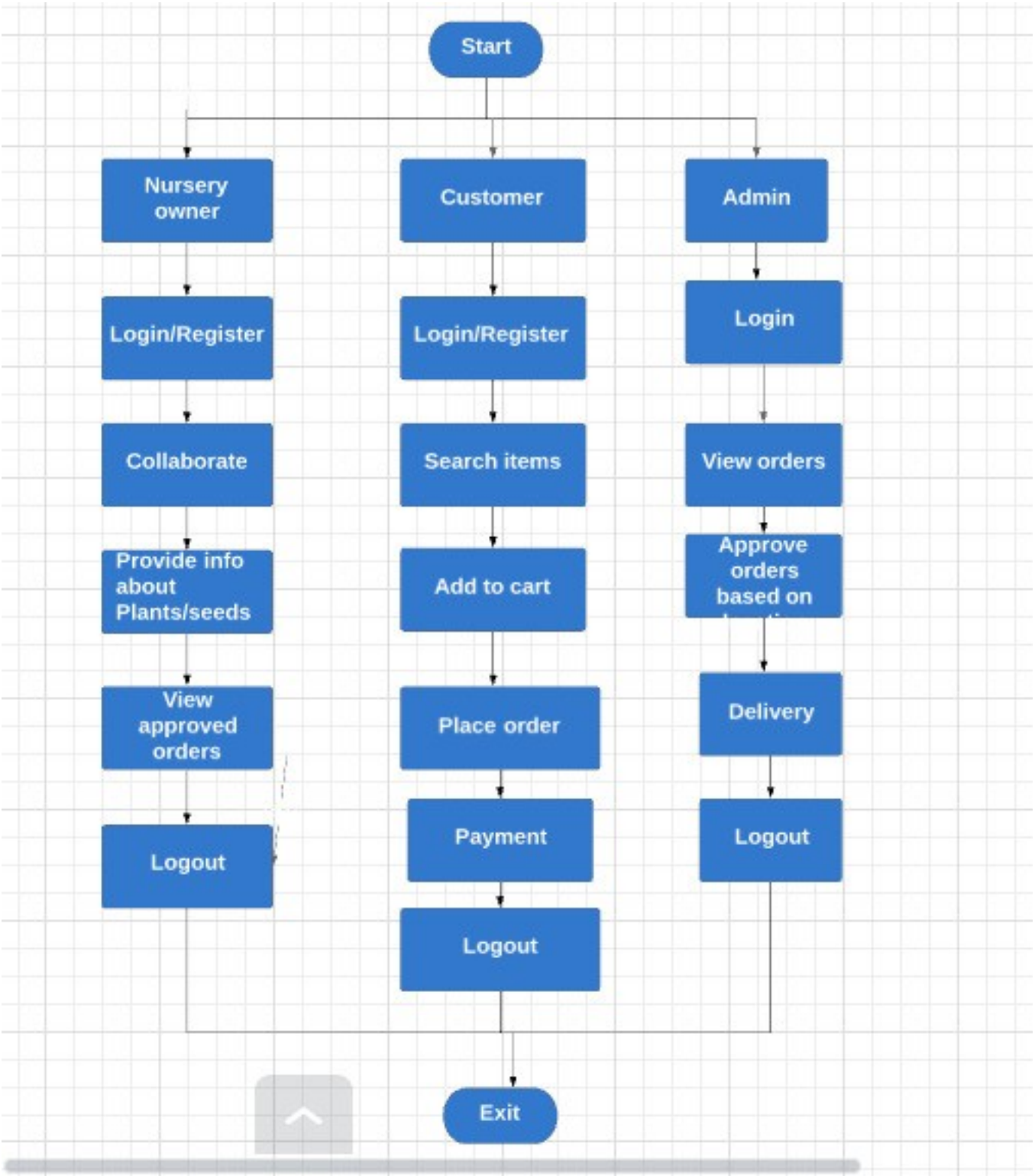
1. Customer registers and log in himself in to the Application.
2. He can select the plant item and purchase the plant by comparing prices with different shop keeper.
3.He can place an Order along with their contact details,Plant Specification and amount of request
4. Purchased plant details will be added to the card.

5. If any complaint about the product then he can give the feedback

C) Nursery owner module:

1.Nursery owner registers and log in himself in to the Application.
2.He can able to collaborate with us by using collaborate option.
3. He should provide the information about the availability of plants, seeds and nursery location.

6.4.REPRESENTATION OF FRAMEWORK



9.SAMPLE CODE

Activity_Main.xml

```
<androidx.drawerlayout.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    tools:openDrawer="start">
    <include
        layout="@layout/app_bar_main"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
        <FrameLayout
            android:id="@+id/fragment_container"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="?android:actionBarSize"
            app:layout_constraintLeft_toLeftOf="parent"
            app:layout_constraintTop_toTopOf="parent" />
        <nl.joery.animatedbottombar.AnimatedBottomBar
            android:id="@+id/navigation"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@drawable/bg_half_rounded"
            android:elevation="5dp"
            app:abb_animationDuration="500"

app:abb_animationInterpolator="@android:anim/overshoot_interpolator"
```

```
        app:abb_indicatorAppearance="round"
        app:abb_indicatorColor="@color/white"
        app:abb_indicatorHeight="5dp"
        app:abb_indicatorMargin="30dp"
        app:abb_selectedIndex="1"
        app:abb_selectedTabType="text"
        app:abb_tabAnimation="slide"
        app:abb_tabColor="@color/white"
        app:abb_tabColorSelected="@color/white"
        app:abb_tabs="@menu/navigation"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        tools:ignore="MissingConstraints" />
</androidx.constraintlayout.widget.ConstraintLayout>
<com.google.android.material.navigation.NavigationView
    android:id="@+id/nav_view"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_gravity="start"
    android:background="@color/teal"
    android:fitsSystemWindows="true"
    android:paddingEnd="15dp"
    app:headerLayout="@layout/nav_header_main"
    app:itemBackground="@drawable/drawer_selected_item"
    app:itemIconTint="@color/drawer_item_color"
    app:itemTextColor="@color/drawer_item_color"
    app:labelVisibilityMode="labeled"
    app:menu="@menu/menu_navigation_drawer"
    tools:ignore="RtlSymmetry">
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_marginBottom="10dp"
            android:gravity="center_horizontal"
            android:text="@string/version_app"
            android:textColor="@color/white"
            android:textSize="14sp" />
```

```
        </RelativeLayout>
    </com.google.android.material.navigation.NavigationView>
</androidx.drawerlayout.widget.DrawerLayout>
```

MainActivity.java

```
package com.project.plantapp;
import android.annotation.SuppressLint;
import android.os.Bundle;
import android.util.Log;
import android.view.MenuItem;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentManager;
import com.google.android.material.navigation.NavigationView;
import com.project.plantapp.menu.favorite.FavoriteFragment;
import com.project.plantapp.menu.home.HomeFragment;
import com.project.plantapp.menu.profile.ProfileFragment;
import java.util.Objects;
import nl.joery.animatedbottombar.AnimatedBottomBar;
public class MainActivity extends AppCompatActivity {
    private static final String TAG = MainActivity.class.getSimpleName();
    private Toolbar toolbar;
    AnimatedBottomBar animatedBottomBar;
    FragmentManager fragmentManager;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        setToolbar();
        initViews(savedInstanceState);
        initComponentsNavHeader();
    }
    private void setToolbar() {
        toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        Objects.requireNonNull(getSupportActionBar()).setTitle(0);
```

```
    }
    @SuppressLint("NonConstantResourceId")
    private void initViews(Bundle savedInstanceState) {
        /**
         * Menu Bottom Navigation Drawer
         * */
        animatedBottomBar = findViewById(R.id.navigation);
        if (savedInstanceState == null) {
            animatedBottomBar.selectTabById(R.id.nav_menu_home, true);
            fragmentManager = getSupportFragmentManager();
            HomeFragment homeFragment = new HomeFragment();
            fragmentManager.beginTransaction().replace(R.id.fragment_container,
homeFragment)
                    .commit();
        }
        animatedBottomBar.setOnTabSelectListener((lastIndex, lastTab,
newIndex, newTab) -> {
            Fragment fragment = null;
            switch (newTab.getId()) {
                case R.id.nav_menu_home:
                    fragment = new HomeFragment();
                    break;
                case R.id.nav_menu_wishlist:
                    fragment = new FavoriteFragment();
                    break;
                case R.id.nav_menu_signin:
                    fragment = new ProfileFragment();
                    break;
            }
            if (fragment != null) {
              fragmentManager = getSupportFragmentManager();

fragmentManager.beginTransaction().replace(R.id.fragment_container,
fragment)
                        .commit();
            } else {
                Log.e(TAG, "Error in creating Fragment");
            }
        });
        /**
         * Menu Navigation Drawer
         **/
```

```
        DrawerLayout drawer = findViewById(R.id.drawer_layout);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
                this, drawer, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
        drawer.addDrawerListener(toggle);
        toggle.setDrawerIndicatorEnabled(false);
        toggle.setToolbarNavigationClickListener(view ->
drawer.openDrawer(GravityCompat.START));
        toggle.setHomeAsUpIndicator(R.drawable.ic_drawer);
        toggle.syncState();
    }
    private void initComponentsNavHeader(){
        NavigationView navigationView = findViewById(R.id.nav_view);
//        navigationView.setItemIconTintList(null); //disable tint on each icon to
use color icon svg
        navigationView.setNavigationItemSelectedListener(new
NavigationView.OnNavigationItemSelectedListener() {
            @SuppressLint("NonConstantResourceId")
            @Override
            public boolean onNavigationItemSelected(@NonNull MenuItem item) {
                switch (item.getItemId()) {
                    case R.id.nav_weeds:
                        Pesan("Menu Weeds");
                        break;
                    case R.id.nav_insects:
                        Pesan("Menu Insects");
                        break;
                    case R.id.nav_diseases:
                        Pesan("Menu Diseases");
                        break;
                    case R.id.nav_products:
                        Pesan("Menu Products");
                        break;
                    case R.id.nav_help:
                        Pesan("Menu Help");
                        break;
                }
                DrawerLayout drawer = findViewById(R.id.drawer_layout);
                drawer.closeDrawer(GravityCompat.START);
                return true;
            }
            private void Pesan(String pesan) {
```

```
            Toast.makeText(MainActivity.this, pesan,
Toast.LENGTH_SHORT).show();
            }
        });
    }
    @Override
    public void onBackPressed() {
        DrawerLayout drawer = findViewById(R.id.drawer_layout);
        if (drawer.isDrawerOpen(GravityCompat.START)) {
            drawer.closeDrawer(GravityCompat.START);
        } else {
            super.onBackPressed();
        }
    }
}
```

Firebase Authentication:

```
package com.google.firebase.quickstart.auth;

import android.app.Activity;

import android.os.Bundle;

import android.util.Log;

import android.widget.Toast;


import androidx.annotation.NonNull;


import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.Task;

import com.google.firebase.auth.AuthResult;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.FirebaseUser;
```

```
public class EmailPasswordActivity extends Activity {

    private static final String TAG = "EmailPassword";

    private FirebaseAuth mAuth;


    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        mAuth = FirebaseAuth.getInstance();

    }


    @Override
    public void onStart() {
        super.onStart();

        FirebaseUser currentUser = mAuth.getCurrentUser();
        if(currentUser != null){
```

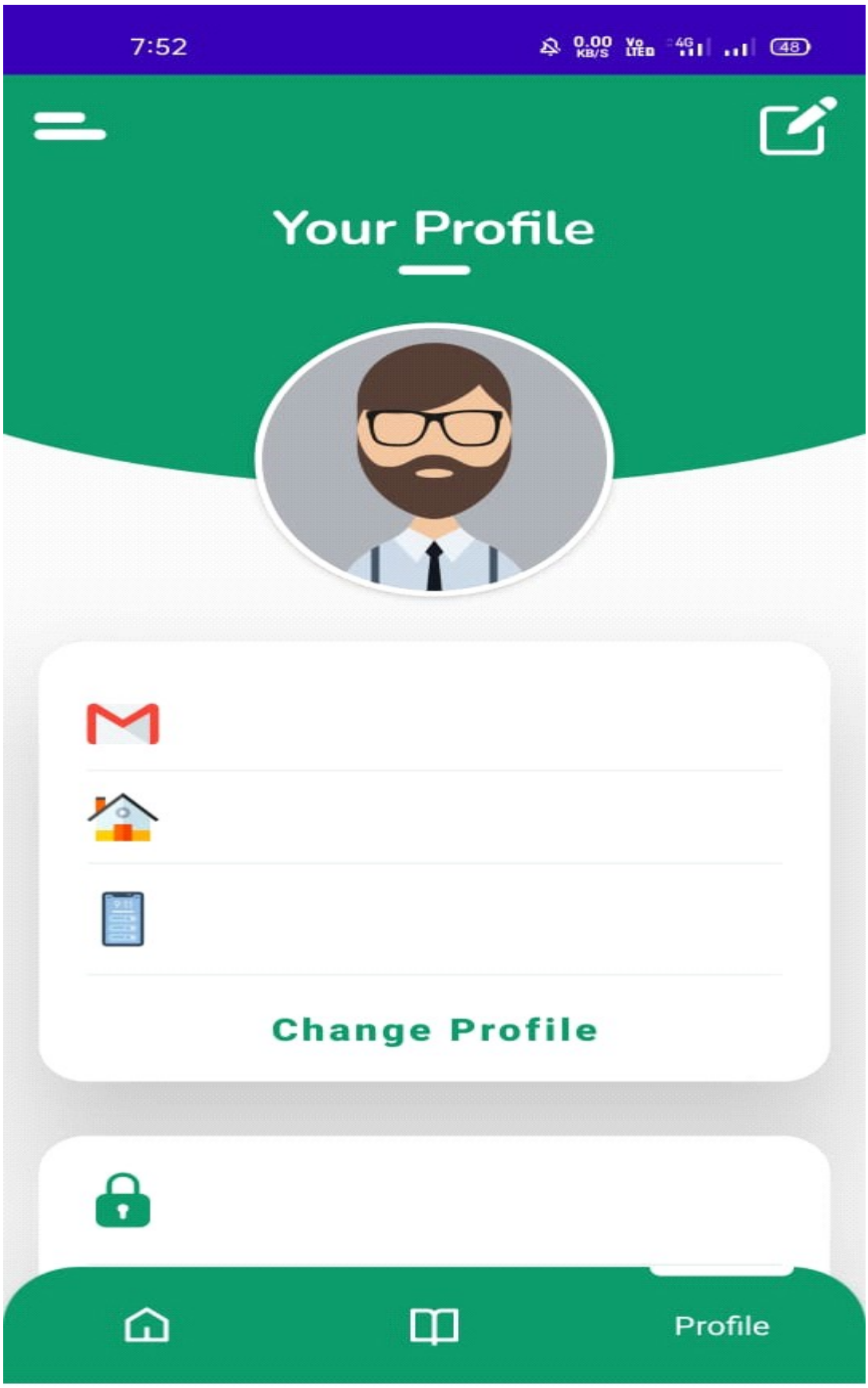
```
            reload();
        }
    }


    private void createAccount(String email, String password) {


        mAuth.createUserWithEmailAndPassword(email, password)
                .addOnCompleteListener(this, new
OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()) {


                            Log.d(TAG, "createUserWithEmail:success");
                            FirebaseUser user = mAuth.getCurrentUser();
                            updateUI(user);
                        } else {


                            Log.w(TAG, "createUserWithEmail:failure",
task.getException());
                            Toast.makeText(EmailPasswordActivity.this, "Authentication
failed.",
                                    Toast.LENGTH_SHORT).show();
                            updateUI(null);
```

```
                }
            }
        });

    }

    private void signIn(String email, String password) {

        mAuth.signInWithEmailAndPassword(email, password)
            .addOnCompleteListener(this, new
OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if (task.isSuccessful()) {

                        Log.d(TAG, "signInWithEmail:success");
                        FirebaseUser user = mAuth.getCurrentUser();
                        updateUI(user);
                    } else {

                        Log.w(TAG, "signInWithEmail:failure", task.getException());
                        Toast.makeText(EmailPasswordActivity.this, "Authentication
failed.",
                                Toast.LENGTH_SHORT).show();
```

```
                    updateUI(null);
                }
            }
        });

}

private void sendEmailVerification() {

    final FirebaseUser user = mAuth.getCurrentUser();
    user.sendEmailVerification()
        .addOnCompleteListener(this, new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {

            }
        });

}

private void reload() { }
private void updateUI(FirebaseUser user) {
```

```
    }
}
```

10.OUTPUT SCREENS


7:52
Your Profile
Change Profile
Profile

7:51

Your Favorite

Angelica Archangelica Flower

Russia

Buy Now

Armeria Maritima Flower

Italy

Buy Now

Favorite

NON-FUNCTIONAL REQUIREMENTS:

Reliability:

User should get appropriate information about his complaint

Usability:

This tool should has user friendly GUI. User can use it effectively. Availability: Using should get information 24x7. User can access at any time with this tool

Accessibility:

This tool support multi user accessing. Any user can access the system from different places to use the tool.

Performance:

User should have fast access to get the information from the help center. User should retrieve the information from help center database very quickly.

Security:

As it is a web based application it should be more secure in order to save help centers confidential data from hackers.

Platform Compatibility:

This tool has to work on any kind of operating system without modifying it.

CONCLUSION:

The proposed application can guarantee to keep the records are safe and privacy which is stored in the database.

1. In this dissertation, we have developed an approach to allow customers to buy plants without even visiting shop.

2. Being able to buy anytime, anywhere, any place.

3. App enables them to browse before they shop, and to Order/Research the product so they have more confidence in what they are buying.

4. Online shopping becomes more enjoyable and easier than real- world shopping.

5. It provides online payment system.

6. Customer can track their order detail and give the feedback if any problem occur during shipment.

FUTURE SCOPE:

1. Number of Product owners can register to the Application for increase their sale.

2. This application can be used by any user to purchase the online plants and get appropriate information by viewing short summery about the plants items through videos.

3. If any changes to make customer can purchase the plants through different payment schemes like debit card, credit card, pat, phone pay, cash on delivery etc.