

LOCATION BASED STUDENT PRESENCE TRACKING APPLICATION

A MINI PROJECT REPORT

Submitted by

JEBAS WINSTON E

201906017

VIKNEYSH RAJ G G A

201906055

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

MEPCO SCHLENK ENGINEERING COLLEGE

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

BONAFIDE CERTIFICATE

Certified that this mini project report “**Location Based Student Presence Tracking Application**” is the bonafide work of **JEBAS WINSTON E (201906017)** and **VIKNEYSH RAJ G G A(201906055)** who carried out the project work under my supervision.

HEAD OF THE DEPARTMENT

Dr. T. Revathi, M.E., Ph.D.,
Senior Professor & Head,
Department of Information Technology,
Mepco Schlenk Engineering College,
Sivakasi-626005.
Virudhunagar Dt.,
Tamil Nadu.

FACULTY INCHARGE

Mrs. N. Malathy, M.E., (Ph. D.),
Assistant Professor (Sr. G),
Department of Information Technology,
Mepco Schlenk Engineering College,
Sivakasi-626005.
Virudhunagar Dt.,
Tamil Nadu.

TABLE OF CONTENTS

Chapter	Topic	Page No.
	ABSTRACT	i
	LIST OF FIGURES	ii
1	INTRODUCTION	1
	1.1. Aim	1
	1.2. Objective	1
	1.3. Modules Used	1
2	SOFTWARE DESCRIPTION	2
	2.1. Android Studio IDE	2
	2.2. Firebase	2
3	SOFTWARE REQUIREMENTS	3
	3.1. Functional Requirements	3
	3.2. Non-functional Requirements	4
4	WORK FLOW	5
	4.1. Flow Diagram	5
5	CODING	6
6	OUTPUT SCREENSHOTS	56
7	CONCLUSION AND FUTURE WORK	59
8	REFERENCES	59

ABSTRACT

It is undeniable that expert system application is gaining popularity. Advancement in computer technology has influenced the development in many other fields including environment security. Over the years the process of manual attendance has been carried out which is not only time consuming but also provides erroneous result. Taking attendance in a class can often be a time consuming and a manual process, which is prone to human error(s) and hence recording incorrect data. Also, querying the data per student can be the tedious process since it involves some sort of manual tracking/counting of days attended. With advent of smartphones and tablets which are very handy to use, this process can be made completely automated and error free by using the right technologies. Automated time and attendance monitoring system provides many benefits to organizations. This reduces the need of pen and paper based manual attendance tracking system. Following this thought, we have proposed an attendance system based on location using global positioning(GPS) technology which is implemented on mobile application on smartphone. The location of a smartphone which can be determined by the GPS. This location is defined as a key of time and attendance tracking on our paper and the attendance is taken using Geo-Fence technique which detects whether the location of the student is within the Geo-Fence area or not.

LIST OF FIGURES

Figure	Title	Page No.
4.1.	Flow Diagram	5
6.1.	User Authentication	56
6.2.	Account Details and Attendance	56
6.3.	Location Monitoring	57
6.4.	Notification and SMS	57
6.5.	Admin Screens	58
6.6.	Hamburger menus	58

1. INTRODUCTION

1.1. Aim

In this project, we aim to build an android application that can mark attendance for students by monitoring their location. The app focuses on monitoring the real-time location of the users and tracking their attendance automatically and updating the same to Firebase real-time database.

1.2. Objective

- To Build a Mobile application to Track and Monitor students movement in campus and hostel premises and give attendance on the basis of location.
- The Mobile application should be useful to track or monitor the in or out movement of students. It should capture the in-time & out-time of student movement in the hostel premises.
- Parent's mobile number should be linked with the application and the movement details of the students may be communicated to the parents' mobile number through SMS.
- The Attendance details of the students in classes (Theory & Practical) is also to be communicated to parents.

1.3. Technology & Concepts Used

The application is built using Android studio IDE 2021.1.1 in Android SDK 32. The minimum SDK is 19 with a target SDK of version 32. The application is debug and run in an Android Virtual Emulator Pixel 4 API 24. The datastore being made use of is Firebase real-time database. The application makes use of the following concepts in Android.

- Views
- ViewGroups
- Menus
- Intents
- Database Connectivity
- Firebase Authentication
- Location Based Services
- Geofencing

2. SOFTWARE DESCRIPTION

2.1. Android Studio IDE

Android Studio is the official Integrated Development Environment (IDE) for android application development. Android Studio provides more features that enhance our productivity while building Android apps. Android Studio was announced on 16th May 2013 at the Google I/O conference as an official IDE for Android app development. It started its early access preview from version 0.1 in May 2013. The first stable built version was released in December 2014. Features of the Android Studio are

- o It has a flexible Gradle-based build system.
- o It has a fast and feature-rich emulator for app testing.
- o Android Studio has a consolidated environment where we can develop for all Android devices.
- o Apply changes to the resource code of our running app without restarting the app.
- o Android Studio provides extensive testing tools and frameworks.
- o It supports C++ and NDK.
- o It provides build-in supports for Google Cloud Platform. It makes it easy to integrate Google Cloud Messaging and App Engine.

The Android Studio project contains one or more modules with resource files and source code files. These include different types of modules-

- o Android app modules
- o Library modules
- o Google App Engine modules

By default, Android Studio displays our project files in the Android project view, as shown in the above image. This view is formed by modules to provide quick access to our project's key source files.

These build files are visible to the top-level under Gradle Scripts. And the app module contains the following folders:

- o manifests: It contains the AndroidManifest.xml file.
- o java: It contains the source code of Java files, including the JUnit test code.
- o res: It contains all non-code resources, UI strings, XML layouts, and bitmap images.

2.2. Firebase

Google Firebase is a Google-backed application development software that enables

developers to develop iOS, Android and Web apps. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment. Firebase is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google's infrastructure. Key features of the Firebase are

1. Authentication

It supports authentication using passwords, phone numbers, Google, Facebook, Twitter, and more. The Firebase Authentication (SDK) can be used to manually integrate one or more sign-in methods into an app.

2. Realtime database

Data is synced across all clients in realtime and remains available even when an app goes offline.

3. Hosting

Firebase Hosting provides fast hosting for a web app; content is cached into content delivery networks worldwide.

4. Test lab

The application is tested on virtual and physical devices located in Google's data centers.

5. Notifications

Notifications can be sent with firebase with no additional coding.

3. SOFTWARE REQUIREMENTS

3.1. Functional Requirements

Login

After registration, one can log in to the system as the system's operator on the user's behalf. The user will get only those privileges that are given to the user for which one has registered. For example, if a user has registered as a student, then the user only has the privilege to view the data and cannot make any changes to the data that is shown.

User

The user can be anyone either a student or an admin. If the user is an admin, then the interface shown is different from the one shown to the students. Like admin can view attendance and can make any changes if any mistakes have been made but students can only view their attendance. Attendance of the students if he/she is

genuinely within the marked range a detailed message and a notification will be sent to the students. Students can upload their images for their profiles.

Attendance

This module is only present in the students' interface but not in the admins' interface. Student can mark their attendance by uploading their currently taken picture and their location coordinates such as latitude and longitude information will be considered for making sure their presence to make attendance. Admins can rectify a student's attendance if it is marked incorrect by chance of any student. Students can only view their attendance if they are not detained for any reason. The marked attendance can only be rectified within a week, after that it cannot be rectified.

3.2. Non-functional Requirements

Security

The user should provide a password to log on to the system. He/she should be able to see the attendance of him/her and can mark their attendance.

Reliability

Due to wireless connectivity, reliability cannot be guaranteed. Anyhow, the system should function with the maximum possible degree of reliability.

Availability

The system should be available during working hours. At any time, the system should be functional and be able to track the attendance of the students.

Maintainability

There should be a facility to add or delete or update users. The admin should have the authority over the process of addition or deletion of users.

Reusability

There's a facility to add or delete or update users each year. The system should be able to adapt to changes over time.

Scalability

The system should be scalable in future.

4. WORK FLOW

4.1. Flow Diagram

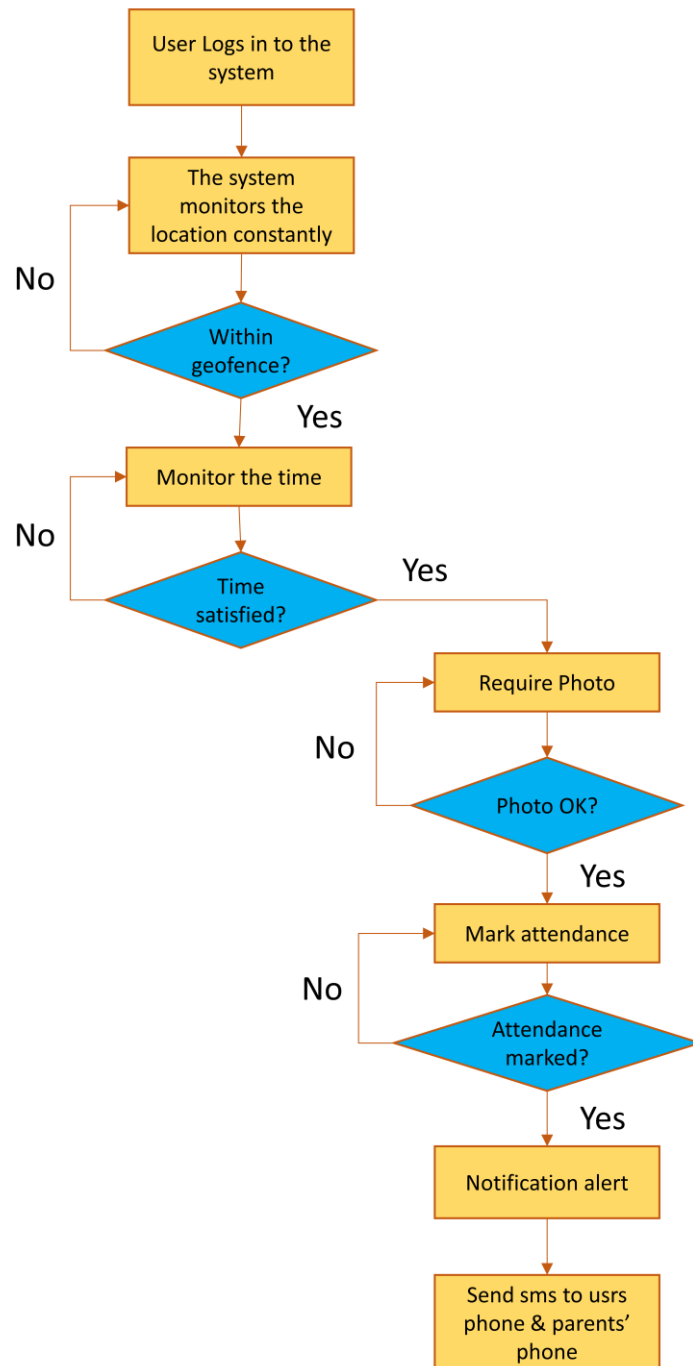


Fig. 4.1. Process Flow

5. CODING

MainActivity.java

```
package com.example.geotendance;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

public class MainActivity extends AppCompatActivity {

    boolean valid;
    Button loginButton;
    EditText emailIDET;
    EditText passwordET;
    FirebaseAuth fAuth;
    FirebaseFirestore fStore;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        emailIDET = findViewById(R.id.emailID);
        passwordET = findViewById(R.id.userPassword);
```

```

loginButton = findViewById(R.id.loginButton);
fAuth = FirebaseAuth.getInstance();
fStore = FirebaseFirestore.getInstance();
loginButton.setOnClickListener(view -> {
    checkFields(emailIDET);
    checkFields(passwordET);
    fAuth.signInWithEmailAndPassword(emailIDET.getText().toString(),
passwordET.getText().toString())
        .addOnSuccessListener(authResult -> {
            Toast.makeText(this, "LoggedIn Successfully!",
Toast.LENGTH_LONG).show();
            checkUserAccessLevel(authResult.getUser().getUid());
        })
        .addOnFailureListener(exception -> {
            Toast.makeText(this, "Try again, your credentials are not valid!",
Toast.LENGTH_LONG).show();
        });
});
}

private void checkUserAccessLevel(String uid) {
    DocumentReference dRef = fStore.collection("Users").document(uid);
    dRef.get().addOnSuccessListener(documentSnapshot -> {
        Log.d("TAG", "onSuccess " + documentSnapshot.getData());
        if(documentSnapshot.getString("isAdmin") != null) {
            startActivity(new Intent(getApplicationContext(),
AdminHome.class).putExtra("Uid", uid));
            finish();
        } else {
            startActivity(new Intent(getApplicationContext(),
Home.class).putExtra("Uid", uid));
            finish();
        }
    }).addOnFailureListener(exception -> {

```

```

        Toast.makeText(this, exception.getMessage(), Toast.LENGTH_LONG).show();
    });

}

public boolean checkFields(EditText textField) {
    if(textField.getText().toString().isEmpty()) {
        valid = false;
        textField.setError("Enter a value!");
    } else {
        valid = true;
    }
    return valid;
}

@Override
protected void onStart() {
    super.onStart();
    if(FirebaseAuth.getInstance().getCurrentUser() != null) {
        startActivity(new Intent(getApplicationContext(), Home.class));
        finish();
    }
}
}

```

Home.java

```

package com.example.geotendance;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

```

```
import androidx.core.app.NotificationCompat;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import android.Manifest;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.app.AlertDialog;
import android.app.Application;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationManager;
import android.net.Uri;
import android.os.Bundle;
import android.os.Looper;
import android.provider.Settings;
import android.view.Gravity;
import android.view.MenuItem;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import android.widget.Toolbar;

import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationCallback;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationResult;
import com.google.android.gms.location.LocationServices;
import com.google.android.material.navigation.NavigationView;
import com.google.firebase.auth.FirebaseAuth;
```

```

public class Home extends AppCompatActivity {

    // View Attributes
    Button logoutBtn, locationBtn, checkAttendanceBtn;
    TextView latitudeTV, longitudeTV;
    String latitude, longitude;

    // Location Attributes
    private static final int REQUEST_LOCATION = 1;
    int PERMISSION_ID = 44;
    final double DESTINATION_LATITUDE = 9.5246967;
    final double DESTINATION_LONGITUDE = 77.8552909;
    FusedLocationProviderClient mFusedLocationClient;

    // Hamburger menu Attributes
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle actionBarDrawerToggle;

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        if(actionBarDrawerToggle.onOptionsItemSelected(item)) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);

        // Hamburger menu

```

```

drawerLayout = findViewById(R.id.drawerLayout);
navigationView = findViewById(R.id.navigationView);
actionBarDrawerToggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.menu_open, R.string.menu_close);
drawerLayout.addDrawerListener(actionBarDrawerToggle);
actionBarDrawerToggle.syncState();
getSupportActionBar().setDisplayHomeAsUpEnabled(true);
navigationView.setNavigationItemSelectedListener(item -> {
    switch (item.getItemId()){
        case R.id.nav_home:
            startActivity(new Intent(getApplicationContext(), Home.class)
                .putExtra("latitude", latitude)
                .putExtra("longitude", longitude)
                .putExtra("Uid", getIntent().getStringExtra("Uid")));
            drawerLayout.closeDrawer(GravityCompat.START);
            break;

        case R.id.nav_account:
            startActivity(new Intent(getApplicationContext(), Accounts.class)
                .putExtra("latitude", latitude)
                .putExtra("longitude", longitude)
                .putExtra("Uid", getIntent().getStringExtra("Uid")));
            drawerLayout.closeDrawer(GravityCompat.START);
            break;

        case R.id.nav_attendance:
            startActivity(new Intent(getApplicationContext(), AttendanceDetails.class)
                .putExtra("latitude", latitude)
                .putExtra("longitude", longitude)
                .putExtra("Uid", getIntent().getStringExtra("Uid")));
            drawerLayout.closeDrawer(GravityCompat.START);
            break;

        case R.id.nav_location:

```



```

        startActivity(new Intent(Intent.ACTION_VIEW,
Uri.parse("https://www.google.com/maps/search/?api=1&query="+latitude+", "+longitude)));
        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_logout:
        FirebaseAuth.getInstance().signOut();
        startActivity(new Intent(getApplicationContext(), MainActivity.class));
        finish();
        drawerLayout.closeDrawer(GravityCompat.START);
        break;
    }
    return false;
});

// Location Control
ActivityCompat.requestPermissions( this,
    new String[] {Manifest.permission.ACCESS_FINE_LOCATION},
REQUEST_LOCATION);

mFusedLocationClient = LocationServices.getFusedLocationProviderClient(this);
getLastLocation();

checkAttendanceBtn = findViewById(R.id.checkAttendance);
checkAttendanceBtn.setOnClickListener(view -> {
    // number of km per degree = ~111km (111.32 in google maps, but range varies
between 110.567km at the equator and 111.699km at the poles)
    double kiloMeters = 0.1;
    double coef = kiloMeters/111.32;
    double oldLatitude = Double.parseDouble(latitude);
    double newLatitude = DESTINATION_LATITUDE + coef;
    // pi / 180 ~= 0.01745
    double oldLongitude = Double.parseDouble(longitude);
    double newLongitude = DESTINATION_LONGITUDE + coef /

```

```

Math.cos(DESTINATION_LATITUDE * 0.01745);
    if((oldLatitude <= newLatitude) && (oldLatitude >=
DESTINATION_LATITUDE)
        && (oldLongitude <= newLongitude) && (oldLongitude >=
DESTINATION_LONGITUDE)) {
        sendNotification();
    } else {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setMessage("Sorry! You're not within the range.");
        builder.setCancelable(true);
        builder.setPositiveButton("Okay", (dialog, which) -> {
            finish();
        });
        builder.setNegativeButton("Cancel", (dialog, which) -> {
            // If user click no then dialog box is canceled.
            dialog.cancel();
        });
        AlertDialog alertDialog = builder.create();
        alertDialog.show();
    }
});

// Logout Functionality
logoutBtn = findViewById(R.id.logout);
logoutBtn.setOnClickListener(view -> {
    FirebaseAuth.getInstance().signOut();
    startActivity(new Intent(this, MainActivity.class));
    finish();
});

// Locating the user
locationBtn = findViewById(R.id.loctionBT);
locationBtn.setOnClickListener(view -> {
    startActivity(new Intent(Intent.ACTION_VIEW,

```

```

Uri.parse("https://www.google.com/maps/search/?api=1&query="+latitude+", "+longitude)));
    });
}

private void sendNotification() {
    NotificationCompat.Builder builder =
        new NotificationCompat.Builder(this)
            .setSmallIcon(R.drawable.ic_baseline_message_24) //set icon for
notification
            .setContentTitle("GeoTendance") //set title of notification
            .setContentText("Your entry is registered successfully!")//this is
notification message
            .setAutoCancel(true) // makes auto cancel of notification
            .setPriority(NotificationCompat.PRIORITY_DEFAULT); //set priority of
notification

    Intent notificationIntent = new Intent(this, NotificationView.class);
    notificationIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
    //notification message will get at NotificationView
    notificationIntent.putExtra("message", "This is a notification message");

    PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, notificationIntent,
        PendingIntent.FLAG_UPDATE_CURRENT);
    builder.setContentIntent(pendingIntent);

    // Add as notification
    NotificationManager manager = (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);
    manager.notify(0, builder.build());
}

@SuppressWarnings("MissingPermission")
private void getLastLocation() {
    if (checkPermissions()) {

```

```

        if (isLocationEnabled()) {
            mFusedLocationClient.getLastLocation().addOnCompleteListener(task -> {
                Location location = task.getResult();
                if (location == null) {
                    requestNewLocationData();
                } else {
                    latitude = String.valueOf(location.getLatitude());
                    longitude = String.valueOf(location.getLongitude());
                }
            });
        } else {
            Toast.makeText(this, "Please turn on" + " your location...",
Toast.LENGTH_LONG).show();

            Intent intent = new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
            startActivity(intent);
        }
    } else {
        requestPermissions();
    }
}

private void requestPermissions() {
    ActivityCompat.requestPermissions(this, new String[]{
        Manifest.permission.ACCESS_COARSE_LOCATION,
        Manifest.permission.ACCESS_FINE_LOCATION}, PERMISSION_ID);
}

private LocationCallback mLocationCallback = new LocationCallback() {

    @Override
    public void onLocationResult(LocationResult locationResult) {
        Location mLastLocation = locationResult.getLastLocation();
        latitude = String.valueOf(mLastLocation.getLatitude());
    }
}

```

```

        longitude = String.valueOf(mLastLocation.getLongitude());
//        latitudeTV.setText("Latitude: " + mLastLocation.getLatitude() + "");
//        longitudeTV.setText("Longitude: " + mLastLocation.getLongitude() + "");
    }
};

@Override
public void
onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
@NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);

    if (requestCode == PERMISSION_ID) {
        if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            getLastLocation();
        }
    }
}

@SuppressWarnings("MissingPermission")
private void requestNewLocationData() {
    LocationRequest mLocationRequest = new LocationRequest();
    mLocationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    mLocationRequest.setInterval(5);
    mLocationRequest.setFastestInterval(0);
    mLocationRequest.setNumUpdates(1);
    mFusedLocationClient = LocationServices.getFusedLocationProviderClient(this);
    mFusedLocationClient.requestLocationUpdates(mLocationRequest,
mLocationCallback, Looper.myLooper());
}

@Override
public void onResume() {

```

```

        super.onResume();
        if (checkPermissions()) {
            getLastLocation();
        }
    }

    private boolean checkPermissions() {
        return ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) ==
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED;
    }

    private boolean isLocationEnabled() {
        LocationManager locationManager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
        return locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER) ||
locationManager.isProviderEnabled(LocationManager.NETWORK_PROVIDER);
    }
}

```

Accounts.java

```

package com.example.geotendance;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import android.content.Intent;

```

```

import android.net.Uri;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.TextView;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

import java.util.Map;

public class Accounts extends AppCompatActivity {

    // View Attributes
    TextView nameValTV, emailValTV, classValTV, phoneValTV;

    // Hamburger menu Attributes
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle actionBarDrawerToggle;

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        if(actionBarDrawerToggle.onOptionsItemSelected(item)) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_accounts);
    }

```

```

// Hamburger menu
drawerLayout = findViewById(R.id.drawerLayout);
navigationView = findViewById(R.id.navigationView);
actionBarDrawerToggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.menu_open, R.string.menu_close);
drawerLayout.addDrawerListener(actionBarDrawerToggle);
actionBarDrawerToggle.syncState();
getSupportActionBar().setDisplayHomeAsUpEnabled(true);
navigationView.setNavigationItemSelectedListener(item -> {
    switch (item.getItemId()){
        case R.id.nav_home:
            startActivity(new Intent(getApplicationContext(),
Home.class).putExtra("Uid", getIntent().getStringExtra("Uid")));
            drawerLayout.closeDrawer(GravityCompat.START);
            break;

        case R.id.nav_account:
            startActivity(new Intent(getApplicationContext(), Accounts.class)
                .putExtra("latitude", getIntent().getStringExtra("latitude"))
                .putExtra("longitude", getIntent().getStringExtra("longitude"))
                .putExtra("Uid", getIntent().getStringExtra("Uid")))
            );
            drawerLayout.closeDrawer(GravityCompat.START);
            break;

        case R.id.nav_attendance:
            startActivity(new Intent(getApplicationContext(), AttendanceDetails.class)
                .putExtra("latitude", getIntent().getStringExtra("latitude"))
                .putExtra("longitude", getIntent().getStringExtra("longitude"))
                .putExtra("Uid", getIntent().getStringExtra("Uid")));
            drawerLayout.closeDrawer(GravityCompat.START);
            break;
    }
});

```



```

        case R.id.nav_location:
            startActivity(new Intent(Intent.ACTION_VIEW,
                Uri.parse("https://www.google.com/maps/search/?api=1&query="+getIntent().getStringExtra(
                    "latitude")+","+getIntent().getStringExtra("longitude"))));
            drawerLayout.closeDrawer(GravityCompat.START);
            break;

        case R.id.nav_logout:
            FirebaseAuth.getInstance().signOut();
            startActivity(new Intent(getApplicationContext(), MainActivity.class));
            finish();
            drawerLayout.closeDrawer(GravityCompat.START);
            break;
    }
    return false;
});

// Getting data from Firestore
nameValTV = findViewById(R.id.nameVal);
emailValTV = findViewById(R.id.emailVal);
classValTV = findViewById(R.id.classVal);
phoneValTV = findViewById(R.id.phoneVal);

FirebaseFirestore.getInstance().collection("Users").document(getIntent().getStringExtra("Uid"))
    .get().addOnCompleteListener(task -> {
        if(task.isSuccessful()) {
            DocumentSnapshot doc = task.getResult();
            if(doc.exists()) {
                Map<String, Object> map = doc.getData();
                nameValTV.setText(String.valueOf(map.get("FullName")));
                emailValTV.setText(String.valueOf(map.get("EmailID")));
            }
        }
    });

```

```

        classValTV.setText(String.valueOf(map.get("Class")));
        phoneValTV.setText(String.valueOf(map.get("PhoneNumber")));
    }
}
});
}
}

```

AdminHome.java

```

package com.example.geotendance;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Button;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.auth.FirebaseAuth;

public class AdminHome extends AppCompatActivity {

    // View Attributes
    Button add, update, remove;

    // Hamburger menu Attributes

```

```

DrawerLayout drawerLayout;
NavigationView navigationView;
ActionBarDrawerToggle actionBarDrawerToggle;

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    if(actionBarDrawerToggle.onOptionsItemSelected(item)) {
        return true;
    }
    return super.onOptionsItemSelected(item);
}

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_admin_home);
    add = findViewById(R.id.add);
    add.setOnClickListener(view -> {
        startActivity(new Intent(this, Registration.class));
    });

    // Hamburger menu
    drawerLayout = findViewById(R.id.drawerLayout);
    navigationView = findViewById(R.id.navigationView);
    actionBarDrawerToggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.menu_open, R.string.menu_close);
    drawerLayout.addDrawerListener(actionBarDrawerToggle);
    actionBarDrawerToggle.syncState();
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    navigationView.setNavigationItemSelectedListener(item -> {
        switch (item.getItemId()){
            case R.id.nav_home:
                startActivity(new Intent(getApplicationContext(),
Home.class).putExtra("Uid", getIntent().getStringExtra("Uid")));

```

```

        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_account:
        startActivity(new Intent(getApplicationContext(), Accounts.class)
            .putExtra("latitude", getIntent().getStringExtra("latitude"))
            .putExtra("longitude", getIntent().getStringExtra("longitude"))
            .putExtra("Uid", getIntent().getStringExtra("Uid")))
        );
        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_location:
        startActivity(new Intent(Intent.ACTION_VIEW,
            Uri.parse("https://www.google.com/maps/search/?api=1&query="+getIntent().getStringExtra(
                "latitude")+","+getIntent().getStringExtra("longitude"))));
        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_logout:
        FirebaseAuth.getInstance().signOut();
        startActivity(new Intent(getApplicationContext(), MainActivity.class));
        finish();
        drawerLayout.closeDrawer(GravityCompat.START);
        break;
    }
    return false;
});
}
}

```

AttendanceDetails.java

```

package com.example.geotendance;

```

```
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;
import androidx.core.content.ContextCompat;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import android.Manifest;
import android.app.AlertDialog;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.MenuItem;
import android.widget.Button;
import android.widget.Toast;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.navigation.NavigationView;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.FirebaseApp;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
```

```

import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.OnProgressListener;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.io.File;
import java.util.Map;
import java.util.UUID;

import de.hdodenhof.circleimageview.CircleImageView;

public class AttendanceDetails extends AppCompatActivity {

    // UI Attributes
    CircleImageView circleImageView;
    FloatingActionButton uploadBtn;
    Button submit;

    // Hamburger menu Attributes
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle actionBarDrawerToggle;

    // Firebase Attributes
    FirebaseStorage fStorage;
    StorageReference storageRef;

    // Other Attributes
    Uri imageUri;
    final double DESTINATION_LATITUDE = 9.5246967;
    final double DESTINATION_LONGITUDE = 77.8552909;

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {

```

```

        if(actionBarDrawerToggle.onOptionsItemSelected(item)) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_attendance_details);

        // Hamburger menu
        drawerLayout = findViewById(R.id.drawerLayout);
        navigationView = findViewById(R.id.navigationView);
        actionBarDrawerToggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.menu_open, R.string.menu_close);

        drawerLayout.addDrawerListener(actionBarDrawerToggle);
        actionBarDrawerToggle.syncState();
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        navigationView.setNavigationItemSelectedListener(item -> {
            switch (item.getItemId()){
                case R.id.nav_home:
                    startActivity(new Intent(getApplicationContext(), Home.class)
                        .putExtra("latitude", getIntent().getStringExtra("latitude"))
                        .putExtra("longitude", getIntent().getStringExtra("longitude"))
                        .putExtra("Uid", getIntent().getStringExtra("Uid")));
                    drawerLayout.closeDrawer(GravityCompat.START);
                    break;

                case R.id.nav_account:
                    startActivity(new Intent(getApplicationContext(), Accounts.class)
                        .putExtra("latitude", getIntent().getStringExtra("latitude"))
                        .putExtra("longitude", getIntent().getStringExtra("longitude"))
                        .putExtra("Uid", getIntent().getStringExtra("Uid")));

```

```

        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_attendance:
        startActivity(new Intent(getApplicationContext(), AttendanceDetails.class)
            .putExtra("latitude", getIntent().getStringExtra("latitude"))
            .putExtra("longitude", getIntent().getStringExtra("longitude"))
            .putExtra("Uid", getIntent().getStringExtra("Uid")));
        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_location:
        startActivity(new Intent(Intent.ACTION_VIEW,
            Uri.parse("https://www.google.com/maps/search/?api=1&query="+
getIntent().getStringExtra("latitude")+" "+getIntent().getStringExtra("longitude"))));
        drawerLayout.closeDrawer(GravityCompat.START);
        break;

    case R.id.nav_logout:
        FirebaseAuth.getInstance().signOut();
        startActivity(new Intent(getApplicationContext(), MainActivity.class));
        finish();
        drawerLayout.closeDrawer(GravityCompat.START);
        break;
    }
    return false;
});

// UI Functionalities
circleImageView = findViewById(R.id.uploadImage);
uploadBtn = findViewById(R.id.uploadImageBtn);
uploadBtn.setOnClickListener(view -> {
    choosePicture();

```



```

});
submit = findViewById(R.id.button);
submit.setOnClickListener(view -> {

FirebaseFirestore.getInstance().collection("Users").document(getIntent().getStringExtra("Uid
")).get().addOnCompleteListener(task -> {
    if (task.isSuccessful()) {
        DocumentSnapshot doc = task.getResult();
        if (doc.exists()) {
            Map<String, Object> map = doc.getData();
            // number of km per degree = ~111km (111.32 in google maps, but range
varies between 110.567km at the equator and 111.699km at the poles)
            double kiloMeters = 0.1;
            double coef = kiloMeters/111.32;
            double oldLatitude =
Double.parseDouble(getIntent().getStringExtra("latitude"));
            double newLatitude = DESTINATION_LATITUDE + coef;
            // pi / 180 ~= 0.01745
            double oldLongitude =
Double.parseDouble(getIntent().getStringExtra("longitude"));
            double newLongitude = DESTINATION_LONGITUDE + coef /
Math.cos(DESTINATION_LATITUDE * 0.01745);
            if((oldLatitude <= newLatitude) && (oldLatitude >=
DESTINATION_LATITUDE)
                && (oldLongitude <= newLongitude) && (oldLongitude >=
DESTINATION_LONGITUDE)) {
                sendNotification();
                sendSMS(String.valueOf(map.get("FullName")),
String.valueOf(map.get("EmailID")),
                String.valueOf(map.get("Class")),
String.valueOf(map.get("PhoneNumber")));
            } else {
                AlertDialog.Builder builder = new AlertDialog.Builder(this);
                builder.setMessage("Sorry! You're not within the range.");

```

```

        builder.setCancelable(true);
        builder.setPositiveButton("Okay", (dialog, which) -> {
            finish();
        });
        builder.setNegativeButton("Cancel", (dialog, which) -> {
            dialog.cancel();
        });
        AlertDialog alertDialog = builder.create();
        alertDialog.show();
    }
}
});
});

```

// Firebase Functionalities

```

fStorage = FirebaseStorage.getInstance();
storageRef = fStorage.getReference();
}

```

```

private void sendNotification() {
    NotificationCompat.Builder builder =
        new NotificationCompat.Builder(this)
            .setSmallIcon(R.drawable.ic_baseline_message_24) //set icon for
notification
            .setContentTitle("GeoTendance") //set title of notification
            .setContentText("Your entry is registered successfully!")//this is
notification message
            .setAutoCancel(true) // makes auto cancel of notification
            .setPriority(NotificationCompat.PRIORITY_DEFAULT); //set priority of
notification

    Intent notificationIntent = new Intent(this, NotificationView.class);
    notificationIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
    //notification message will get at NotificationView
}

```

```

notificationIntent.putExtra("message", "This is a notification message");

PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, notificationIntent,
    PendingIntent.FLAG_UPDATE_CURRENT);
builder.setContentIntent(pendingIntent);

// Add as notification
NotificationManager manager = (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);
    manager.notify(0, builder.build());
}

private void sendSMS(String fullName, String email, String claas, String
mobileNumber) {
    if(ContextCompat.checkSelfPermission(this, Manifest.permission.SEND_SMS) ==
PackageManager.PERMISSION_GRANTED) {
        SmsManager smsManager = SmsManager.getDefault();
        StringBuilder message = new StringBuilder();
        message.append("Name: " + fullName + "\n");
        message.append("Email ID: " + email + "\n");
        message.append("Class: " + claas + "\n");
        message.append("Your attendance marked successfully!");
        smsManager.sendTextMessage(mobileNumber,null,
String.valueOf(message),null,null);
        Toast.makeText(this, "Message sent successfully!",
Toast.LENGTH_LONG).show();
    }
}

private void choosePicture() {
    Intent intent = new Intent();
    intent.setType("image/*");
    intent.setAction(Intent.ACTION_GET_CONTENT);
    startActivityForResult(intent, 1);
}

```

```

    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent
data) {
        super.onActivityResult(requestCode, resultCode, data);
        if(requestCode == 1 && resultCode == RESULT_OK && data != null &&
data.getData() != null) {
            imageUrl = data.getData();
            circleImageView.setImageURI(imageUri);
            uploadImage();
        }
    }

    private void uploadImage() {
        Uri file = Uri.fromFile(new File("path/to/images/"));
        ProgressDialog pd = new ProgressDialog(this);
        pd.setTitle("Uploading Image...");
        pd.show();
        String randomKey = UUID.randomUUID().toString();
        StorageReference mountainsRef = storageRef.child("images/"+randomKey );
        mountainsRef.putFile(imageUri)
            .addOnSuccessListener(taskSnapshot -> {
                Snackbar.make(findViewById(android.R.id.content), "Image Uploaded.",
Snackbar.LENGTH_LONG).show();
                pd.dismiss();
            })
            .addOnFailureListener(e -> {
                pd.dismiss();
                Toast.makeText(getApplicationContext(), "upload Failed",
Toast.LENGTH_LONG).show();
            })
            .addOnProgressListener(snapshot -> {
                double progressPercent = (100.00 * snapshot.getBytesTransferred() /

```

```
snapshot.getTotalByteCount());
        pd.setMessage("Progress: " + (int) progressPercent + "%");
    });
}
}
```

Constants.java

```
package com.example.geotendance;

public class Constants {
    static final int LOCATION_SERVICE_ID = 175;
    static final String ACTION_START_LOCATION_SERVICE =
"startLocationService";
    static final String ACTION_STOP_LOCATION_SERVICE = "stopLocationService";
}
```

LocationService.java

```
package com.example.geotendance;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.app.Service;
import android.content.Context;
import android.content.Intent;
import android.os.Build;
import android.os.IBinder;
import android.os.Looper;
import android.util.Log;

import androidx.annotation.NonNull;
```

```

import androidx.annotation.Nullable;
import androidx.core.app.NotificationCompat;

import com.google.android.gms.location.LocationCallback;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationResult;
import com.google.android.gms.location.LocationServices;

public class LocationService extends Service {

    private LocationCallback locationCallback = new LocationCallback() {
        @Override
        public void onLocationResult(@NonNull LocationResult locationResult) {
            super.onLocationResult(locationResult);
            if(locationResult != null && locationResult.getLastLocation() != null) {
                double latitude = locationResult.getLastLocation().getLatitude();
                double longitude = locationResult.getLastLocation().getLongitude();
                Log.d("LOCATION_UPDATE", latitude+ ", " +longitude);
            }
        }
    };

    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        throw new UnsupportedOperationException("Not yet implemented");
    }

    private void startLocationService() {
        String channelId = "location_notification_channel";
        NotificationManager notificationManager = (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);

        Intent resultIntent = new Intent();
        PendingIntent pendingIntent = PendingIntent.getActivity(getApplicationContext(),0,

```

```

        resultIntent, PendingIntent.FLAG_UPDATE_CURRENT);
NotificationCompat.Builder builder = new NotificationCompat.Builder(
    getApplicationContext(),
    channelId
);
builder.setSmallIcon(R.mipmap.ic_launcher);
//    builder.setContentIntent("LocationService");
builder.setDefaults(NotificationCompat.DEFAULT_ALL);
builder.setContentText("Running");
builder.setContentIntent(pendingIntent);
builder.setAutoCancel(false);
builder.setPriority(NotificationCompat.PRIORITY_MAX);

if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
    if ((notificationManager != null) &&
(notificationManager.getNotificationChannel(channelId) == null)) {
        NotificationChannel notificationChannel = new NotificationChannel(
            channelId, "Location Service",
NotificationManager.IMPORTANCE_HIGH);
        notificationChannel.setDescription("This channel is used by location service");
        notificationManager.createNotificationChannel(notificationChannel);
    }
}

LocationRequest locationRequest = new LocationRequest();
locationRequest.setInterval(4000);
locationRequest.setFastestInterval(2000);
locationRequest.setPriority(locationRequest.PRIORITY_HIGH_ACCURACY);

LocationServices.getFusedLocationProviderClient(this).requestLocationUpdates(locationReq
uest,

    locationCallback, Looper.getMainLooper());
startForeground(Constants.LOCATION_SERVICE_ID, builder.build());

```

```

    }

    private void stopLocationService() {

LocationServices.getFusedLocationProviderClient(this).removeLocationUpdates(locationCal
lback);
        stopForeground(true);
        stopSelf();
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        if(intent != null) {
            String action = intent.getAction();
            if(action.equals(Constants.ACTION_START_LOCATION_SERVICE)) {
                startLocationService();
            } else if (action.equals(Constants.ACTION_STOP_LOCATION_SERVICE)) {
                stopLocationService();
            }
        }
        return super.onStartCommand(intent, flags, startId);
    }
}

```

NotificationView.java

```

package com.example.geotendance;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;

```



```

public class NotificationView extends AppCompatActivity {
    TextView textView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_notification_view);
        textView = findViewById(R.id.text_view);
        //getting the notification message
        String message=getIntent().getStringExtra("message");
        textView.setText(message);
    }
}

```

Registration.java

```

package com.example.geotendance;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import android.app.ProgressDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Button;
import android.widget.EditText;

```

```

import android.widget.Toast;

import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.navigation.NavigationView;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.mikhaellopez.circularimageview.CircularImageView;

import java.io.File;
import java.util.HashMap;
import java.util.Map;
import java.util.UUID;

public class Registration extends AppCompatActivity {

    // View Attributes
    CircularImageView userImage;
    FloatingActionButton userImageButton;
    EditText fullNameET, emailIDET, passwordET, classET, phoneNumberET;
    Button submitButton;

    // Firebase Attributes
    FirebaseAuth fAuth;
    FirebaseFirestore fStore;
    FirebaseStorage fStorage;
    StorageReference storageRef;

    // Other Attributes
    boolean valid = true;

```

```

Uri imageUrl;

// Hamburger menu Attributes
DrawerLayout drawerLayout;
NavigationView navigationView;
ActionBarDrawerToggle actionBarDrawerToggle;

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    if(actionBarDrawerToggle.onOptionsItemSelected(item)) {
        return true;
    }
    return super.onOptionsItemSelected(item);
}

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_registration);

    // Hamburger menu
    drawerLayout = findViewById(R.id.drawerLayout);
    navigationView = findViewById(R.id.navigationView);
    actionBarDrawerToggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.menu_open, R.string.menu_close);
    drawerLayout.addDrawerListener(actionBarDrawerToggle);
    actionBarDrawerToggle.syncState();
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    navigationView.setNavigationItemSelectedListener(item -> {
        switch (item.getItemId()){
            case R.id.nav_home:
                startActivity(new Intent(this, Home.class)
                    .putExtra("latitude", getIntent().getStringExtra("latitude"))
                    .putExtra("longitude",getIntent().getStringExtra("longitude")))

```

```

        .putExtra("Uid", getIntent().getStringExtra("Uid"))));
drawerLayout.closeDrawer(GravityCompat.START);
break;

case R.id.nav_account:
    startActivity(new Intent(this, Accounts.class)
        .putExtra("latitude", getIntent().getStringExtra("latitude"))
        .putExtra("longitude",getIntent().getStringExtra("longitude"))
        .putExtra("Uid", getIntent().getStringExtra("Uid"))));
drawerLayout.closeDrawer(GravityCompat.START);
break;

case R.id.nav_location:
    startActivity(new Intent(Intent.ACTION_VIEW,
        Uri.parse("https://www.google.com/maps/search/?api=1&query="+
getIntent().getStringExtra("latitude")+" "+getIntent().getStringExtra("longitude"))));
drawerLayout.closeDrawer(GravityCompat.START);
break;

case R.id.nav_logout:
    FirebaseAuth.getInstance().signOut();
    startActivity(new Intent(this, MainActivity.class));
    finish();
    drawerLayout.closeDrawer(GravityCompat.START);
    break;
}
return false;
});

fullNameET = (EditText) findViewById(R.id.fullName);
emailIDET = (EditText) findViewById(R.id.emailID);
passwordET = (EditText) findViewById(R.id.password);
classET = (EditText) findViewById(R.id.department);
phoneNumberET = findViewById(R.id.phoneNumber);

```

```

submitButton = (Button) findViewById(R.id.submit);
fAuth = FirebaseAuth.getInstance();
fStore = FirebaseFirestore.getInstance();

submitButton.setOnClickListener(view -> {
    if(valid) {
        checkFields(fullNameET);
        checkFields(emailIDET);
        checkFields(passwordET);
        checkFields(classET);
        fAuth.createUserWithEmailAndPassword(emailIDET.getText().toString(),
passwordET.getText().toString())
            .addOnSuccessListener(authResult -> {
                FirebaseUser user = fAuth.getCurrentUser();
                DocumentReference dReference =
fStore.collection("Users").document(user.getUid());
                Map<String, Object> userInfo = new HashMap<>();
                userInfo.put("FullName", fullNameET.getText().toString());
                userInfo.put("EmailID", emailIDET.getText().toString());
                userInfo.put("Class", classET.getText().toString());
                userInfo.put("PhoneNumber", phoneNumberET.getText().toString());
                userInfo.put("isUser", "1");
                dReference.set(userInfo);
                Toast.makeText(this, "User Added Successfully!",
Toast.LENGTH_LONG).show();
                startActivity(new Intent(this, Home.class));
                finish();
            })
            .addOnFailureListener(exception -> {
                Toast.makeText(this, "Action Failed, Try Again!",
Toast.LENGTH_LONG).show();
            });
    }
});

```

```

// UI Functionalities
userImage = (CircularImageView) findViewById(R.id.userImage);
userImageButton = (FloatingActionButton) findViewById(R.id.userImageButton);
userImageButton.setOnClickListener(view -> {
    choosePicture();
});

// Firebase Functionalities
fStorage = FirebaseStorage.getInstance();
storageRef = fStorage.getReference();
}

public boolean checkFields(EditText textField) {
    if(textField.getText().toString().isEmpty()) {
        valid = false;
        textField.setError("Field is empty");
    } else {
        valid = true;
    }
    return valid;
}

private void choosePicture() {
    Intent intent = new Intent();
    intent.setType("image/*");
    intent.setAction(Intent.ACTION_GET_CONTENT);
    startActivityForResult(intent, 1);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent
data) {
    super.onActivityResult(requestCode, resultCode, data);

```

```

        if(requestCode == 1 && resultCode == RESULT_OK && data != null &&
data.getData() != null) {
            imageUri = data.getData();
            userImage.setImageURI(imageUri);
            uploadImage();
        }
    }

    private void uploadImage() {
        Uri file = Uri.fromFile(new File("path/to/images/"));
        ProgressDialog pd = new ProgressDialog(this);
        pd.setTitle("Uploading Image...");
        pd.show();
        String randomKey = UUID.randomUUID().toString();
        StorageReference mountainsRef = storageRef.child("images/"+randomKey );
        mountainsRef.putFile(imageUri)
            .addOnSuccessListener(taskSnapshot -> {
                Snackbar.make(findViewById(android.R.id.content), "Image Uploaded.",
Snackbar.LENGTH_LONG).show();
                pd.dismiss();
            })
            .addOnFailureListener(e -> {
                pd.dismiss();
                Toast.makeText(getApplicationContext(), "upload Failed",
Toast.LENGTH_LONG).show();
            })
            .addOnProgressListener(snapshot -> {
                double progressPercent = (100.00 * snapshot.getBytesTransferred() /
snapshot.getTotalByteCount());
                pd.setMessage("Progress: " + (int) progressPercent + "%");
            });
    }
}

```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.ACCESS_BACKGROUND_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE"/>
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.SEND_SMS"/>
    <uses-permission android:name="android.permission.RECEIVE_SMS"/>
    <uses-permission android:name="android.permission.FOREGROUND_SERVICE"/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.GeoTendance"
        tools:targetApi="31">
        <activity
```



```

        android:name=".AttendanceDetails"
        android:exported="false">
        <meta-data
            android:name="android.app.lib_name"
            android:value="" />
    </activity>
    <activity
        android:name=".AdminHome"
        android:exported="false">
        <meta-data
            android:name="android.app.lib_name"
            android:value="" />
    </activity>
    <activity
        android:name=".Accounts"
        android:exported="false">
        <meta-data
            android:name="android.app.lib_name"
            android:value="" />
    </activity>
    <activity
        android:name=".NotificationView"
        android:exported="false">
        <meta-data
            android:name="android.app.lib_name"
            android:value="" />
    </activity>
    <!--

```

TODO: Before you run your application, you need a Google Maps API key.

To get one, follow the directions here:

<https://developers.google.com/maps/documentation/android-sdk/get-api-key>

your
Once you have your API key (it starts with "AIza"), define a new property in
project's local.properties file (e.g. MAPS_API_KEY=Aiza...), and replace the
"YOUR_API_KEY" string in this file with "\${MAPS_API_KEY}".

-->

```
<meta-data
```

```
    android:name="com.google.android.geo.API_KEY"
```

```
    android:value="API-KEY" />
```

```
<activity
```

```
    android:name=".MainActivity"
```

```
    android:exported="true">
```

```
    <intent-filter>
```

```
        <action android:name="android.intent.action.MAIN" />
```

```
        <category android:name="android.intent.category.LAUNCHER" />
```

```
    </intent-filter>
```

```
    <meta-data
```

```
        android:name="android.app.lib_name"
```

```
        android:value="" />
```

```
</activity>
```

```
<activity
```

```
    android:name=".Registration"
```

```
    android:exported="false">
```

```
    <meta-data
```

```
        android:name="android.app.lib_name"
```

```
        android:value="" />
```

```
</activity>
```

```
<activity
```

```
    android:name=".Home"
```

```
    android:exported="false">
```

```
    <meta-data
```

```
        android:name="android.app.lib_name"
```

```
        android:value="" />
```

```
        </activity>
    </application>
</manifest>
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="GeoTendance"
        android:textColor="@color/black"
        android:textSize="50sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.134" />

    <ImageView
        android:layout_width="325dp"
        android:layout_height="236dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.813"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.325"
```

```
app:srcCompat="@drawable/images"
android:contentDescription="Login Image" />
```

```
<TextView
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Login Credentials"
    android:textColor="@color/black"
    android:textSize="75px"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.517"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.584" />
```

```
<EditText
```

```
    android:id="@+id/emailID"
    android:layout_width="238dp"
    android:layout_height="49dp"
    android:ems="10"
    android:hint="Email ID"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.542"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.699" />
```

```
<EditText
```

```
    android:id="@+id/userPassword"
    android:layout_width="237dp"
    android:layout_height="54dp"
```

```

    android:ems="10"
    android:hint="Password"
    android:inputType="textPassword"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.54"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.799" />

```

<Button

```

    android:id="@+id/loginButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="LogIn"
    android:backgroundTint="@color/green_200"
    android:textColor="@color/black"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.897" />

```

</androidx.constraintlayout.widget.ConstraintLayout>

activity_accounts.xml

```

<?xml version="1.0" encoding="utf-8"?>
<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:layout_width="match_parent"
    android:layout_height="match_parent"

```

```
android:id="@+id/drawerLayout"  
tools:context=".Accounts">
```

```
<androidx.constraintlayout.widget.ConstraintLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">
```

```
<com.mikhaellopez.circularimageview.CircularImageView  
    android:layout_width="225dp"  
    android:layout_height="225dp"  
    android:src="@drawable/user"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.298" />
```

```
<TextView  
    android:id="@+id/textView4"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Account Details"  
    android:textSize="50sp"  
    android:textColor="@color/black"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.491"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.082" />
```

```
<TextView  
    android:id="@+id/nameTV"  
    android:layout_width="wrap_content"
```

```

android:layout_height="wrap_content"
android:text="Name"
android:textColor="@color/black"
android:textSize="30sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.132"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.587" />

```

<TextView

```

android:id="@+id/nameVal"
android:layout_width="212dp"
android:layout_height="49dp"
android:text="nameVal"
android:textColor="@color/black"
android:textSize="20sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.919"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.596" />

```

<TextView

```

android:id="@+id/emailTV"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Email ID"
android:textColor="@color/black"
android:textSize="30sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"

```

```
app:layout_constraintHorizontal_bias="0.146"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.706" />
```

<TextView

```
android:id="@+id/emailVal"
android:layout_width="214dp"
android:layout_height="42dp"
android:text="emailVal"
android:textColor="@color/black"
android:textSize="20sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.934"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.708" />
```

<TextView

```
android:id="@+id/classTV"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Class"
android:textColor="@color/black"
android:textSize="30sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.13"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.819" />
```

<TextView


```
android:id="@+id/classVal"
android:layout_width="216dp"
android:layout_height="47dp"
android:text="classVal"
android:textColor="@color/black"
android:textSize="20sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.943"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.827" />
```

<TextView

```
android:id="@+id/phoneTV"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Mobile No"
android:textColor="@color/black"
android:textSize="30sp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.13"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.914" />
```

<TextView

```
android:id="@+id/phoneVal"
android:layout_width="216dp"
android:layout_height="47dp"
android:text="phoneVal"
android:textColor="@color/black"
android:textSize="20sp"
```

```

        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.943"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.923" />

</androidx.constraintlayout.widget.ConstraintLayout>

<com.google.android.material.navigation.NavigationView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:id="@+id/navigationView"
    app:headerLayout="@layout/header_menu"
    app:menu="@menu/main_menu"
    android:fitsSystemWindows="true"
    android:layout_gravity="start">

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

```

activity_admin_home.xml

```

<?xml version="1.0" encoding="utf-8"?>

<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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/drawerLayout"
    tools:context=".AdminHome">

```

```
<androidx.constraintlayout.widget.ConstraintLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">
```

```
<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Welcome Admin"
    android:textColor="@color/black"
    android:textSize="50sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.659"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.087" />
```

```
<Button
    android:id="@+id/add"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Add"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.439" />
```

```
<Button
    android:id="@+id/update"
    android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:text="Update"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.553" />

```

```

<Button

```

```

    android:id="@+id/remove"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Remove"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintHorizontal_bias="0.509"
    app:layout_constraintVertical_bias="0.658" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

```

<com.google.android.material.navigation.NavigationView

```

```

    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:id="@+id/navigationView"
    app:headerLayout="@layout/header_menu"
    app:menu="@menu/admin_main_menu"
    android:fitsSystemWindows="true"
    android:layout_gravity="start">

```

```

</com.google.android.material.navigation.NavigationView>

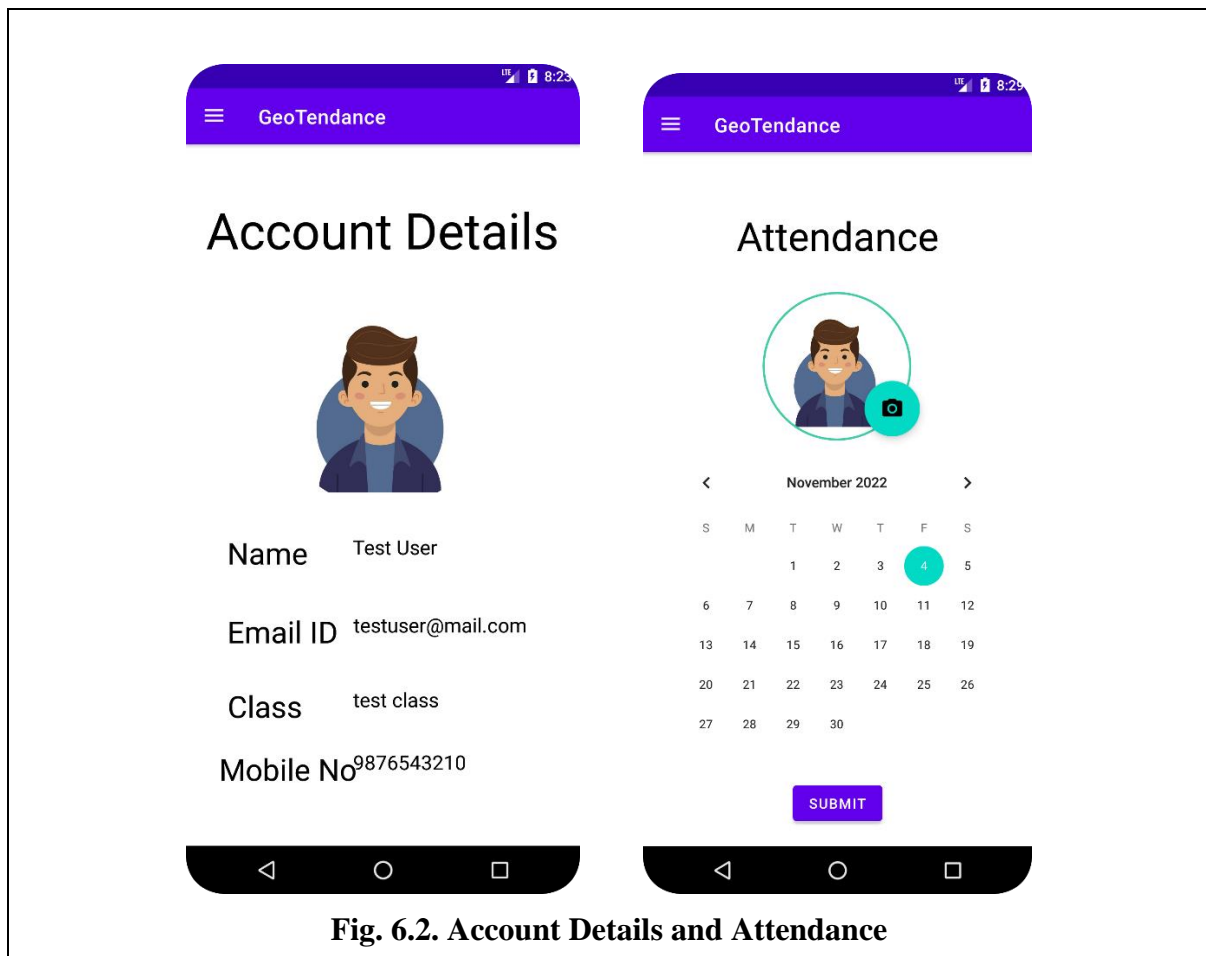
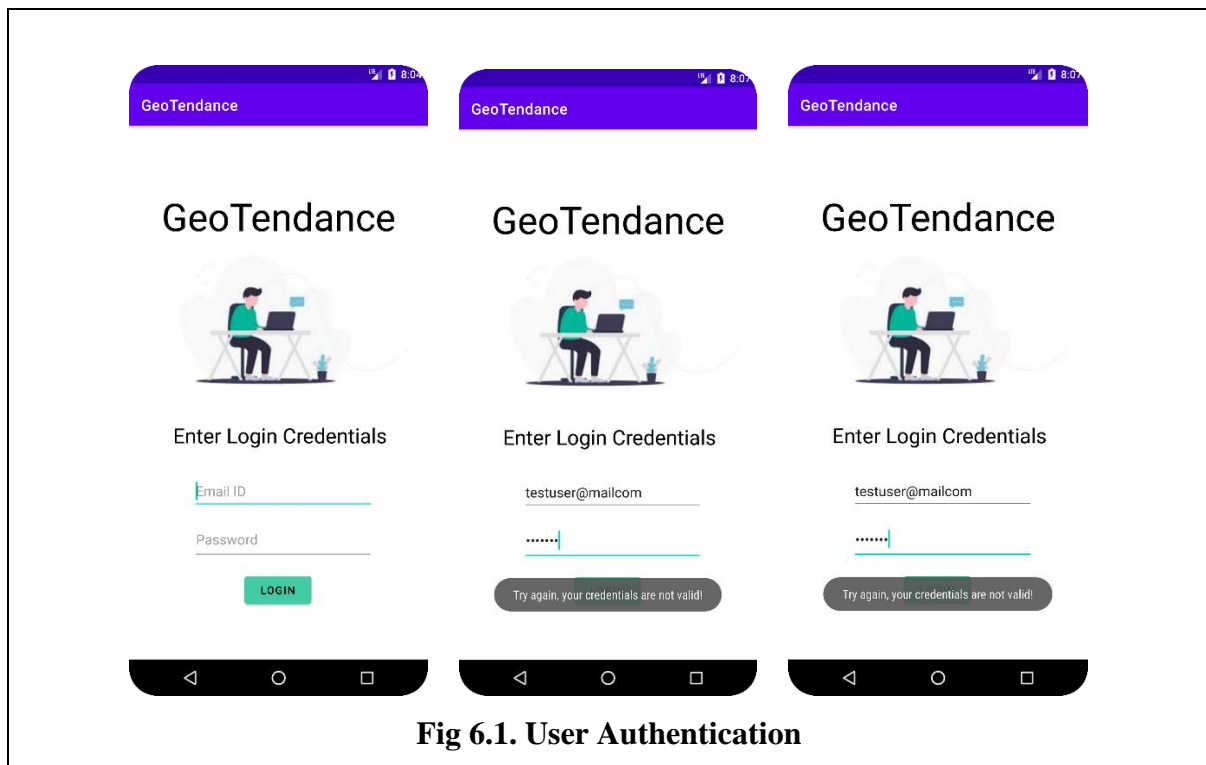
```

```

</androidx.drawerlayout.widget.DrawerLayout>

```

6. OUTPUT SCREENSHOTS



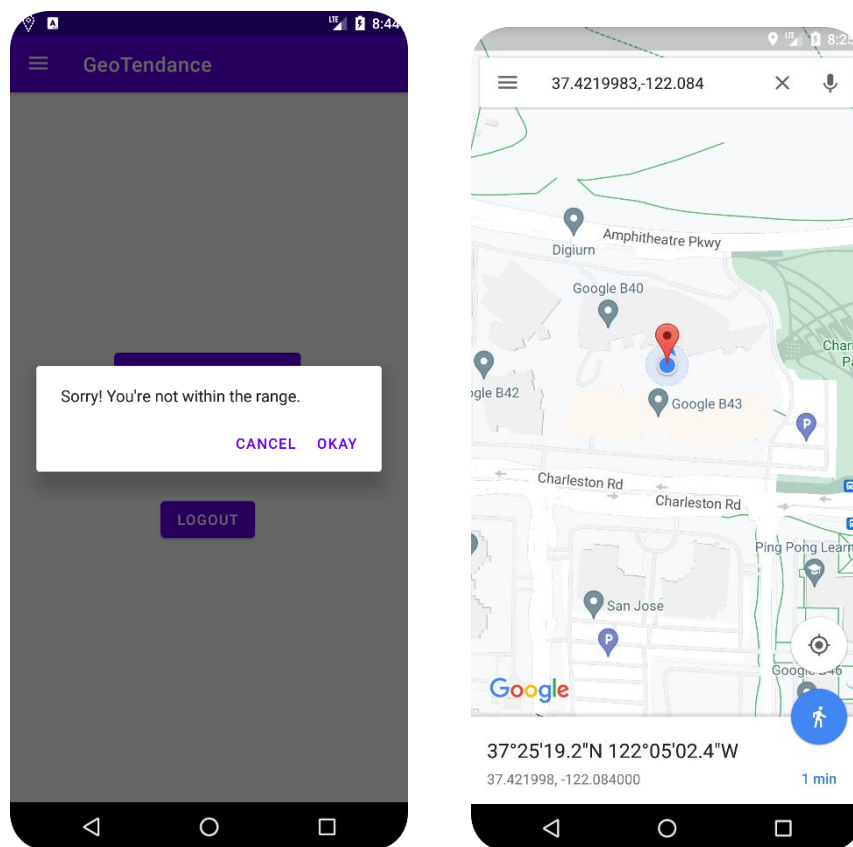


Fig. 6.3. Location Monitoring

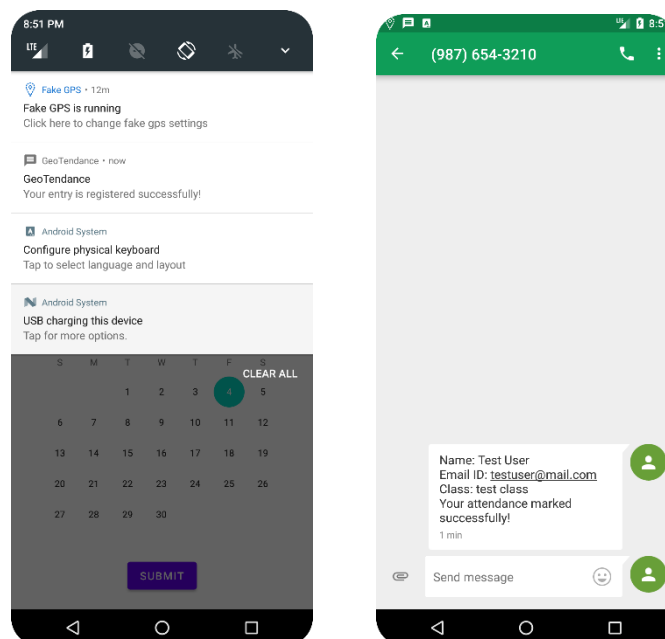


Fig. 6.4. Notifications and SMS

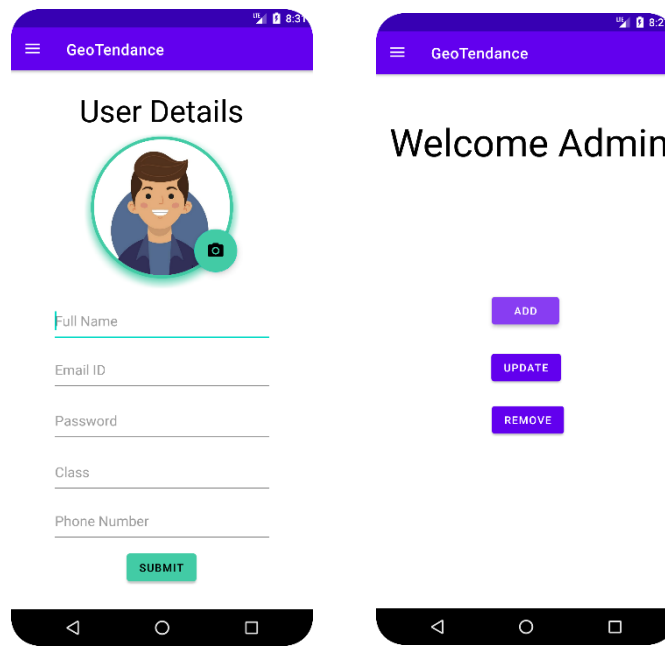


Fig. 6.5. Admin Screens

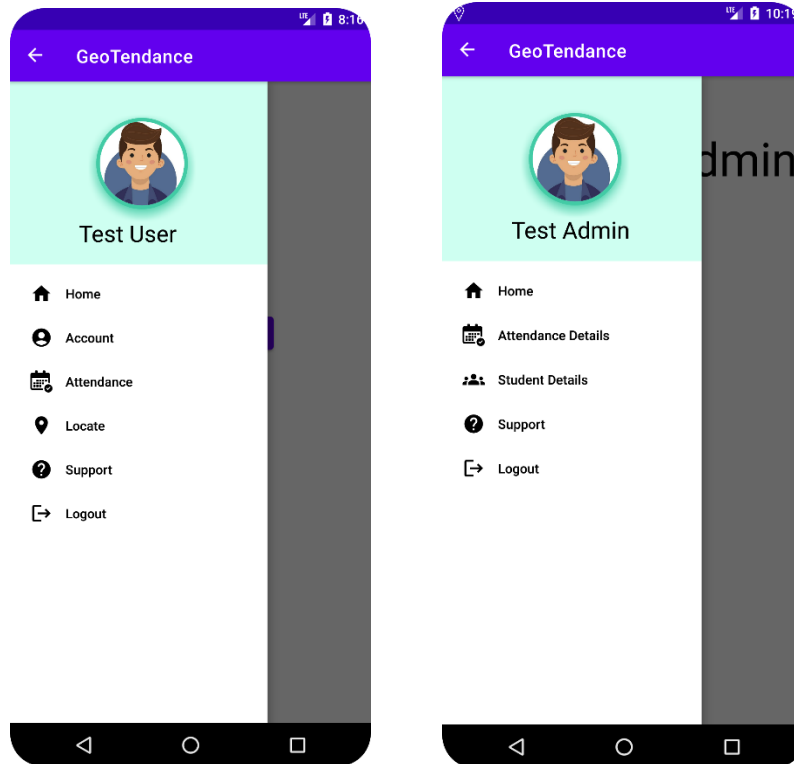


Fig. 6.6. Hamburger Menus

7. CONCLUSION

Registering attendance during a class can be very time consuming process, which requires manual steps and is error prone. This application solves all the downfalls of registering attendance manually. It allows the student's presence to be registered automatically, which not only saves time but is also error free. Additionally, the admin has the luxury to look at attendance record of students whenever he wants. The work can be extended into an ERP platform for a school or a college. Furthermore, like students, we can incorporate this system for a company of employers.

8. REFERENCES

- [1] Shinde, R.N., Nilose, A., & Chandankhede, P.H. (2022). Design and Development of Geofencing Based Attendance System for Mobile Application. 2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22), 1-6.
- [2] B.S.Satpute, Nikitha K., Pooja K., Rupali B., Priya K., Smart Location Based Student Attendance Management System, 2022 10th International Conference on Emerging Trends in Engineering and Technology ,DOI: 10.17148/IJARCCE.2021.10446
- [3] Rahate, S.W. (2016). Geo-fencing Infrastructure: Location Based Service.
- [4] Nilesh Prasad, Swetha S.. GeoFencing: Location based services, International Journal of Advance Research, Ideas and Innovations in Technology, www.IJARIIT.com.
- [5] <https://developer.android.com/docs>
- [6] <https://firebase.google.com/docs>
- [7] <https://uxplanet.org/the-ultimate-guide-to-the-hamburger-menu-and-its-alternatives-e2da8dc7f1db>