SHEGuard

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

BY

T.N.S.Samanvitha (22501A05H6)

Under the Guidance of

Mr. B. Vishnu Vardhan,

Assistant Professor

PRASAD V POTLURI SIDDHARTHA INSTITUTE OF TECHNOLOGY

(Permanently affiliated to JNTU :: Kakinada, Approved by AICTE)

(An NBA & NAAC A+ accredited and ISO 9001:2015 Certified Institution)

Kanuru, Vijayawada - 520007

2024-25

PRASAD V POTLURI

SIDDHARTHA INSTITUTE OF TECHNOLOGY

(Permanently affiliated to JNTU :: Kakinada, Approved by AICTE)

(An NBA & NAAC A+ accredited and ISO 9001:2015 certified institution)

Kanuru, Vijayawada – 520007

CERTIFICATE

This is to certify that the project report title "SHEGuard" is the Bonafide work of T.N.S.Samanvitha (22501A05H6) ,in partial fulfilment of completing the Academic project in Advanced Mobile Application Development (20CS6721) during the academic year 2024-25.

Signature of the Incharge **Signature of the HOD**

INDEX

1. ABSTRACT

SHEGuard is a comprehensive safety application created to give women a sense of security and confidence in their daily lives. With a single tap on the SOS button, the app instantly alerts emergency contacts, nearby police, and safety networks during moments of danger. Its live location feature enables family members or trusted contacts to monitor the user's route, and automatic alerts in unsafe regions offer added protection. SHEGuard also supports voice-based and gesture-based SOS triggers, ensuring that help can be reached even when the user cannot manually use the phone. Powered by secure end-to-end encryption and supported by verified responders and NGOs round the clock, the app focuses not just on handling emergencies but on making women feel safe and supported at all times.

The clean and intuitive interface makes SHEGuard accessible for women of all age groups. It also fosters a sense of community by enabling users to connect with others in their area for additional support. With AI-driven risk detection, the app can identify potential threats early and help users move more confidently. Whether traveling alone, returning home late, or exploring unfamiliar environments, SHEGuard ensures peace of mind at every step. Safety is a fundamental right, and SHEGuard commits to safeguarding it.

1.1. SDG JUSTIFICATION REPORT

SDGs Covered:

- **SDG 3:** Good Health & Well-Being
- **SDG 5:** Gender Equality
- **SDG 11:** Sustainable Cities & Communities

How SHEGuard Supports These SDGs

- **Improved Personal Safety (SDG 3, 5, 11):**
  SHEGuard delivers instant alerts, real-time tracking, and emergency responses that help protect women's well-being in public areas.
- **Women Empowerment Through Knowledge (SDG 5):**
  The app offers learning materials related to safety, legal rights, and self-defense, helping women stay more informed and confident.
- **Community Safety Networks (SDG 5, 11):**
  By connecting users with verified responders and local groups, SHEGuard promotes safer and more supportive communities.
- **Quick Emergency Handling (SDG 3, 11):**
  Features like live tracking, SOS alerts, and AI-based threat detection ensure faster and more effective emergency responses.
- **Scalable Safety Solution (SDG 3, 5, 11):**
  SHEGuard can be expanded across cities and institutions to enhance citywide safety infrastructure.

2. INTRODUCTION

In the modern world, personal safety is becoming increasingly important, especially for women who often travel alone. SHEGuard is designed to be more than a typical mobile app—it functions as a personal safety partner that provides immediate help, continuous tracking, and intelligent threat detection. Whether walking home at night or moving through unfamiliar areas, SHEGuard ensures immediate access to support.

With its one-tap SOS feature, users can quickly notify police, emergency contacts, and nearby safety networks. The app supports real-time GPS tracking, allowing trusted individuals to monitor the user's movements for added reassurance. AI-powered alerts warn users when they enter potentially risky zones. For situations where manual use may not be possible, SHEGuard offers gesture and voice-based SOS activation.

The app prioritizes privacy, ensuring that all sensitive data is secured through encryption. SHEGuard also collaborates with verified safety groups, NGOs, and responders to provide 24/7 support. Designed with an easy-to-use interface, the app caters to women from all backgrounds and technical skill levels.

At its core, SHEGuard aims to prevent danger, offer support, and enable women to navigate the world confidently. It blends powerful technology with community support to turn safety into a guaranteed right—not a privilege.

3. OBJECTIVES AND SCOPE OF THE PROJECT

Objectives

The main goal of SHEGuard is to create a user-friendly and highly efficient safety app that enhances personal protection, especially for women. Key objectives include:

1. **Fast Emergency Response:**
   Introduce a one-tap SOS option to immediately inform emergency contacts and authorities.

2. **Live Location Monitoring:**
   Provide real-time GPS tracking so trusted contacts can follow the user's movements.

3. **AI-Based Alerts:**
   Detect dangerous zones and automatically alert users before risks increase.

4. **Hidden Emergency Triggers:**
   Add voice and gesture-based SOS features for situations where manual access is impossible.

5. **Strong Data Security:**
   Protect user privacy through encrypted communication.

Scope

SHEGuard aims to support individuals—especially women—through both reactive and proactive safety tools. Main scope areas include:

- **User Sign-up & Login:**
  Secure registration with personal details and emergency contact management.
- **SOS Feature:**
  A single button to notify police, emergency contacts, and safety responders instantly.
- **Real-Time Location Sharing:**
  Users can share live GPS data with trusted people during risky situations.
- **Community & Support Integration:**
  Access verified safety volunteers and organizations at any time.

4. SOFTWARE USED – EXPLANATION

Android

Google's open-source OS for mobile devices.

Used for:

- Building the entire app interface
- Designing layouts with XML
- Handling components like Activities, Fragments, RecyclerView, etc.

Java

A secure, object-oriented language ideal for Android apps.

Used for:

- App logic and event handling
- Firebase communication
- Managing SOS activation and background services

Firebase

A cloud platform offering authentication, storage, and real-time data handling.

Used for:

- Storing user profiles, contacts, and SOS alerts
- Managing real-time location updates
- User login and secure authentication

APIs

Used for interacting with various external services:

- **Google Maps API:** For live location and map display

- **Firebase API:** For cloud data operations

- **Flask API:** For AI-based threat detection and analysis

Development Tools

- **Android Studio:**

  Official IDE with layout editors, code assistance, debugging tools, and emulators.

- **GitHub:**

  Used for version control, collaboration, and backing up code history.

5. PROPOSED MODEL

1. Overview

SHEGuard is a smartphone-based safety solution offering live tracking, quick SOS alerts, and AI-powered risk detection. It supports women by enabling fast communication during emergencies and offering community help through verified responders.

2. System Components

User Authentication

Stores details like:

- Name
- Email
- Phone number
- Emergency contacts
- Trusted rescuers

Live Location Tracking

Allows trusted users to view the current location in real time using Google Maps.

SOS Alert System

One-tap SOS to contact:

- Emergency contacts
- Police
- Safety responders

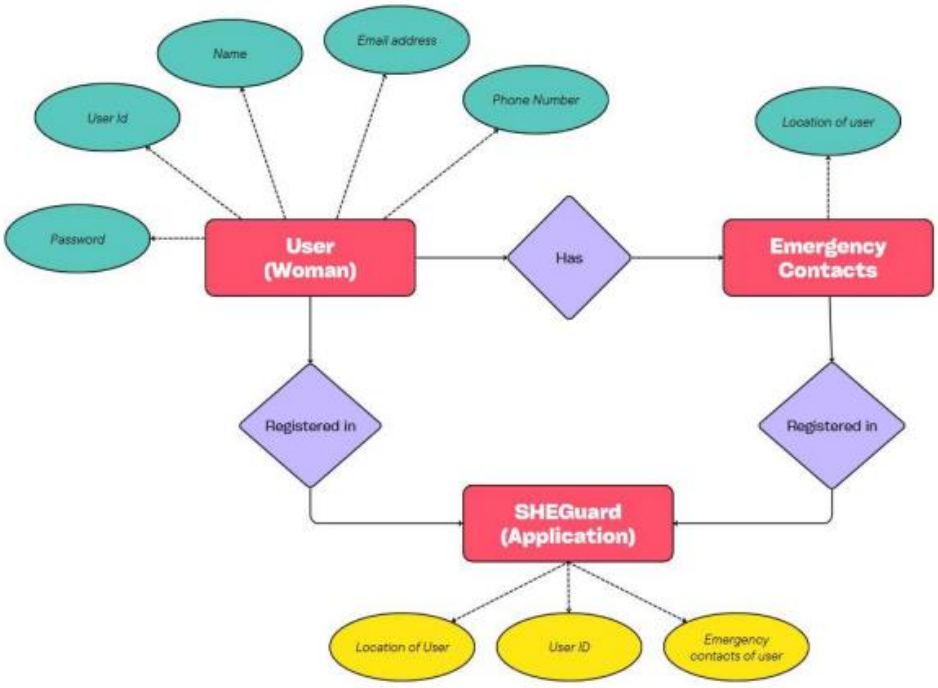
User Dashboard

Provides:

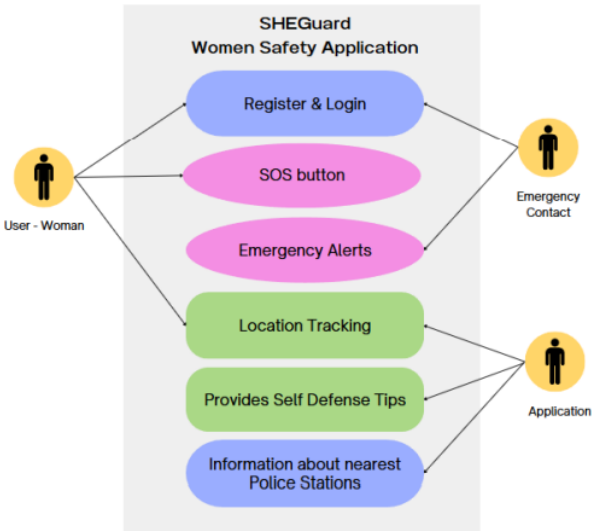
- Current location
- Emergency contacts

- Recent alerts
- SOS history
- Customizable safety settings

3. ER Model



4. Use Case Model



6. SAMPLE CODE

Github Repository Link : https://github.com/SamanvithaTadepalli/SHEGuard

Folder Structure :

SheGuard/

```
├── .idea/
│   ├── inspectionProfiles/
│   ├── (various XML configuration files)
├── app/
│   ├── src/
│   │   ├── main/
│   │   │   ├── assets/
│   │   │   │   ├── police-operation-siren.mp3
│   │   │   ├── java/com/android/sheguard/
│   │   │   │   ├── api/
│   │   │   │   │   ├── MyFirebaseAuth.java
│   │   │   │   │   ├── NotificationAPI.java
│   │   │   │   ├── common/
│   │   │   │   │   ├── Constants.java
│   │   │   │   ├── config/
│   │   │   │   │   ├── Prefs.java
│   │   │   │   ├── model/
│   │   │   │   │   ├── ContactModel.java
│   │   │   │   │   ├── HelplineModel.java
│   │   │   │   │   ├── NotificationDataModel.java
│   │   │   │   │   ├── NotificationSenderModel.java
│   │   │   │   │   ├── NotificationTokenModel.java
│   │   │   │   │   ├── UserModel.java
│   │   │   │   ├── service/
```

```
│ │ │ │ ├── FireBaseMessageService.java
│ │ │ │ ├── SosService.java
│ │ │ ├── ui/
│ │ │ │ ├── activity/
│ │ │ │ │ ├── LoginRegisterActivity.java
│ │ │ │ │ ├── MainActivity.java
│ │ │ │ │ ├── OnBoardingActivity.java
│ │ │ │ ├── adapter/
│ │ │ │ ├── core/
│ │ │ │ ├── entity/
│ │ │ │ ├── fragment/
│ │ │ │ ├── view/
│ │ │ ├── util/
│ │ │ │ ├── AppUtil.java
│ │ │ │ ├── FirebaseUtil.java
│ │ │ │ ├── NotificationClient.java
│ │ │ │ ├── SosUtil.java
│ │ │ ├── SheGuard.java
│ │ ├── res/
│ │ │ ├── anim/
│ │ │ ├── color/
│ │ │ ├── drawable/
│ │ │ ├── layout/
│ │ │ ├── menu/
│ │ │ ├── mipmap/
│ │ │ ├── navigation/
│ │ │ ├── raw/
│ │ │ ├── transition/
│ │ │ ├── values/
```

```
│ │ │ │ ├── xml/
│ │ │ ├── AndroidManifest.xml
│ ├── build.gradle
│ ├── proguard-rules.pro
├── gradle/
│ ├── wrapper/
├── build.gradle
├── settings.gradle
├── gradlew
├── gradlew.bat
```

MainActivity.java

```
package com.android.sheguard.ui.activity;

import android.content.Intent;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

import androidx.navigation.NavController;

import androidx.navigation.Navigation;

import androidx.navigation.ui.NavigationUI;

import com.android.sheguard.R;

import com.android.sheguard.databinding.ActivityMainBinding;

import com.android.sheguard.util.ObservableVariable;

import com.google.firebase.auth.FirebaseAuth;

@SuppressWarnings("FieldCanBeLocal")

public class MainActivity extends AppCompatActivity {

public static ObservableVariable<Boolean> shakeDetection = new
ObservableVariable<>();

private ActivityMainBinding binding;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivityMainBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

}

public void toggleDrawer() {

if (binding.drawerLayout.isDrawerOpen(binding.navView)) {
```

```
binding.drawerLayout.closeDrawer(binding.navView);
} else {
binding.drawerLayout.openDrawer(binding.navView);
}
}
@Override
public boolean onSupportNavigateUp() {
NavController navController = Navigation.findNavController(this,
R.id.fragmentContainerView);
return NavigationUI.navigateUp(navController, binding.drawerLayout) ||
super.onSupportNavigateUp();
}
@Override
protected void onStart() {
super.onStart();
FirebaseAuth firebaseAuth = FirebaseAuth.getInstance();
if (firebaseAuth.getCurrentUser() == null ||
!firebaseAuth.getCurrentUser().isEmailVerified()) {
startActivity(new Intent(MainActivity.this, OnBoardingActivity.class));
finishAffinity();
}
}
}
```

LoginRegisterActivity.java

```
package com.android.sheguard.ui.activity;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

import androidx.navigation.NavController;

import androidx.navigation.Navigation;

import com.android.sheguard.R;

import com.android.sheguard.databinding.ActivityLoginRegisterBinding;

@SuppressWarnings("FieldCanBeLocal")

public class LoginRegisterActivity extends AppCompatActivity {

private ActivityLoginRegisterBinding binding;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivityLoginRegisterBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

}

@Override

public boolean onSupportNavigateUp() {

NavController navController = Navigation.findNavController(this,

R.id.fragmentContainerView);

return navController.navigateUp() || super.onSupportNavigateUp();

}

}
```

HomeFragment.java

```
package com.android.sheguard.ui.fragment;

import android.app.NotificationChannel;

import android.app.NotificationManager;

import android.content.Context;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.os.Looper;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import androidx.activity.result.ActivityResultCallback;

import androidx.activity.result.ActivityResultLauncher;

import androidx.activity.result.contract.ActivityResultContracts;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import androidx.fragment.app.Fragment;

import androidx.navigation.NavOptions;

import androidx.navigation.Navigation;

import com.android.sheguard.R;

import com.android.sheguard.common.Constants;

import com.android.sheguard.config.Prefs;

import com.android.sheguard.databinding.FragmentHomeBinding;

import com.android.sheguard.service.SosService;
```

```
import com.android.sheguard.ui.activity.LoginRegisterActivity;
import com.android.sheguard.ui.activity.MainActivity;
import com.android.sheguard.util.AppUtil;
import com.android.sheguard.util.FirebaseUtil;
import com.android.sheguard.util.SosUtil;
import com.google.android.material.navigation.NavigationView;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import java.util.Iterator;
import java.util.Map;
import java.util.Objects;
public class HomeFragment extends Fragment {
private FragmentHomeBinding binding;
@Override
public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup container,
Bundle
savedInstanceState) {
binding = FragmentHomeBinding.inflate(inflater, container, false);
View view = binding.getRoot();
((AppCompatActivity) requireActivity()).setSupportActionBar(binding.header.toolbar);
binding.header.collapsingToolbar.setTitle(getString(R.string.activity_home_title));
binding.header.collapsingToolbar.setSubtitle(getString(R.string.activity_home_desc,
getString(R.string.unknown_user)));
```

```
setUserNameOnTitle();
Objects.requireNonNull(((AppCompatActivity)
requireActivity()).getSupportActionBar()).setDisplayHomeAsUpEnabled(true);
Objects.requireNonNull(((AppCompatActivity)
requireActivity()).getSupportActionBar()).setHomeAsUpIndicator(R.drawable.ic_nav_dr
awer);
NotificationManager notificationManager = (NotificationManager)
requireContext().getSystemService(Context.NOTIFICATION_SERVICE);
NotificationChannel channel1 = new
NotificationChannel(getString(R.string.notification_channel_push),
getString(R.string.notification_channel_push),
NotificationManager.IMPORTANCE_HIGH);
NotificationChannel channel2 = new
NotificationChannel(getString(R.string.notification_channel_emergency),
getString(R.string.notification_channel_emergency),
NotificationManager.IMPORTANCE_DEFAULT);
notificationManager.createNotificationChannel(channel1);
notificationManager.createNotificationChannel(channel2);
binding.sosButton.setOnClickListener(v -> {
if (AppUtil.permissionsGranted(getContext()) &&
SosUtil.isGPSEnabled(requireContext()))
{
SosUtil.activateInstantSosMode(requireContext());
} else if (!AppUtil.permissionsGranted(getContext())) {
multiplePermissions.launch(AppUtil.REQUIRED_PERMISSIONS);
```

```
} else {
SosUtil.turnOnGPS(requireContext());
}
}MainActivity.shakeDetection.setValue(Prefs.getBoolean(Constants.SETTINGS_SHAK
E_DETECTI
ON, false));
MainActivity.shakeDetection.setOnChangeListener(newValue -> {
binding.btnShakeDetection.setVisibility(newValue ? View.VISIBLE : View.GONE);
updateButtonText();
if (!newValue) {
SosUtil.stopSosNotificationService(requireContext());
}
});
binding.btnShakeDetection.setVisibility(Prefs.getBoolean(Constants.SETTINGS_SHAK
E_DETECT
ION, false) ? View.VISIBLE : View.GONE);
updateButtonText();
binding.btnShakeDetection.setOnClickListener(v -> {
if (!SosService.isRunning) {
if (AppUtil.permissionsGranted(getContext()) &&
SosUtil.isGPSEnabled(requireContext())) {
SosUtil.startSosNotificationService(requireContext());
Snackbar.make(requireActivity().findViewById(android.R.id.content),
getString(R.string.service_started), Snackbar.LENGTH_LONG).show();
} else if (!AppUtil.permissionsGranted(getContext())) {
```

```
multiplePermissions.launch(AppUtil.REQUIRED_PERMISSIONS);
} else {
SosUtil.turnOnGPS(requireContext());
}
} else {
SosUtil.stopSosNotificationService(requireContext());
Snackbar.make(requireActivity().findViewById(android.R.id.content),
getString(R.string.service_stopped), Snackbar.LENGTH_LONG).show();
}
updateButtonText();
});
binding.contacts.setOnClickListener(v ->
Navigation.findNavController(view).navigate(R.id.action_homeFragment_to_contactsFragment));
binding.helpline.setOnClickListener(v ->
Navigation.findNavController(view).navigate(R.id.action_homeFragment_to_helplineFragment));
binding.safetyTips.setOnClickListener(v ->
Navigation.findNavController(view).navigate(R.id.action_homeFragment_to_safetyTipsFragment));
binding.about.setOnClickListener(v ->
Navigation.findNavController(view).navigate(R.id.action_homeFragment_to_aboutFragment));
FirebaseUtil.updateToken();
initializeDrawerItems();
```

```
if (!AppUtil.permissionsGranted(getContext())) {
multiplePermissions.launch(AppUtil.REQUIRED_PERMISSIONS);
}
return view;
}
private void initializeDrawerItems() {
((NavigationView)
requireActivity().findViewById(R.id.navView)).setNavigationItemSelectedListener(item
-> {
int id = item.getItemId();
NavOptions navOptions = new NavOptions.Builder()
.setEnterAnim(0)
.setExitAnim(0)
.setPopEnterAnim(R.anim.slide_out)
.setPopExitAnim(R.anim.fade_in)
.build();
if (id == R.id.nav_profile) {
Navigation.findNavController(binding.getRoot()).navigate(R.id.action_homeFragment_to
_profileFra
gment, null, navOptions);
} else if (id == R.id.nav_settings) {
Navigation.findNavController(binding.getRoot()).navigate(R.id.action_homeFragment_to
_settingsFr
agment, null, navOptions);
} else if (id == R.id.nav_logout) {
```

```
FirebaseAuth.getInstance().signOut();
Intent intent = new Intent(getContext(), LoginRegisterActivity.class);
intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
startActivity(intent);
}
((MainActivity) requireActivity()).toggleDrawer();
return true;
});
}
public void setUserNameOnTitle() {
final String[] userName = {getString(R.string.unknown_user)};
FirebaseFirestore.getInstance()
.collection(Constants.FIRESTORE_COLLECTION_USERLIST)
.document(Objects.requireNonNull(FirebaseAuth.getInstance().getCurrentUser()).getUid(
))
.get()
.addOnCompleteListener(task -> {
if (task.isSuccessful()) {
DocumentSnapshot document = task.getResult();
if (document.exists()) {
userName[0] = document.getString("name");
Prefs.putString(Constants.PREFS_USER_NAME, userName[0]);
}
}
```

```
if (getContext() != null) {
binding.header.collapsingToolbar.setSubtitle(getString(R.string.activity_home_desc,
userName[0]));
}
});
}
private void updateButtonText() {
new Handler(Looper.getMainLooper()).postDelayed(() -> {
if (getContext() != null) {
binding.btnShakeDetection.setText(SosService.isRunning ?
getString(R.string.btn_stop_service) : getString(R.string.btn_start_service));
}
}, 200);
}
private final ActivityResultLauncher<String[]> multiplePermissions =
registerForActivityResult(new ActivityResultContracts.RequestMultiplePermissions(),
new
ActivityResultCallback<Map<String, Boolean>>() {
@Override
public void onActivityResult(Map<String, Boolean> result) {
Iterator<Map.Entry<String, Boolean>> it = result.entrySet().iterator();
while (it.hasNext()) {
Map.Entry<String, Boolean> pair = it.next();
if (!pair.getValue()) {
Snackbar snackbar =
```

```
Snackbar.make(requireActivity().findViewById(android.R.id.content),
R.string.permission_must_be_granted, Snackbar.LENGTH_INDEFINITE);
snackbar.setAction(R.string.grant, v -> {
multiplePermissions.launch(new String[]{pair.getKey()});
snackbar.dismiss();
});
snackbar.show();
}
if (!it.hasNext() && AppUtil.permissionsGranted(getActivity())) {
binding.btnShakeDetection.performClick();
}
}
}
});
}
```

ProfileFragment.java

```
package com.android.sheguard.ui.fragment;
import android.Manifest;
import android.content.Context;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.Looper;
```

```
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.fragment.app.Fragment;
import androidx.navigation.Navigation;
import com.android.sheguard.R;
import com.android.sheguard.common.Constants;
import com.android.sheguard.databinding.FragmentProfileBinding;
import com.android.sheguard.model.UserModel;
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.gms.location.Priority;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import java.io.IOException;
import java.util.List;
import java.util.Locale;
```

```
import java.util.Objects;
public class ProfileFragment extends Fragment {
private FragmentProfileBinding binding;
private LocationManager locationManager = null;
private LocationRequest locationRequest = null;
@Override
public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup container,
Bundle
savedInstanceState) {
binding = FragmentProfileBinding.inflate(inflater, container, false);
View view = binding.getRoot();
((AppCompatActivity) requireActivity()).setSupportActionBar(binding.header.toolbar);
ActionBar actionBar = ((AppCompatActivity) requireActivity()).getSupportActionBar();
if (actionBar != null) {
actionBar.setDisplayHomeAsUpEnabled(true);
actionBar.setDisplayShowHomeEnabled(true);
binding.header.collapsingToolbar.setTitle(getString(R.string.activity_profile_title));
binding.header.collapsingToolbar.setSubtitle(getString(R.string.activity_profile_desc));
}
if (locationRequest == null) {
locationRequest = new
LocationRequest.Builder(Priority.PRIORITY_HIGH_ACCURACY,
5000)
.setWaitForAccurateLocation(false)
.setMinUpdateIntervalMillis(2000)
```

```
.setMaxUpdateDelayMillis(5000)
build();
}
getUserDetails();
getCurrentLocation();
binding.btnEditProfile.setOnClickListener(v ->
Navigation.findNavController(view).navigate(R.id.action_profileFragment_to_editProfile
Fragment));
return view;
}
private void getUserDetails() {
FirebaseFirestore.getInstance()
.collection(Constants.FIRESTORE_COLLECTION_USERLIST)
.document(Objects.requireNonNull(FirebaseAuth.getInstance().getCurrentUser()).getUid(
))
.get()
.addOnCompleteListener(task -> {
if (task.isSuccessful()) {
DocumentSnapshot document = task.getResult();
if (document.exists()) {
UserModel user = new UserModel(
document.getString("name"),
document.getString("email"),
document.getString("phone")
);
```

```
binding.tvName.setText(user.getName());
binding.tvEmail.setText(user.getEmail());
binding.tvPhone.setText(user.getPhone());
}
}
});
}
private boolean isGPSEnabled() {
if (locationManager == null) {
locationManager = (LocationManager)
requireContext().getSystemService(Context.LOCATION_SERVICE);
}
return locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);
}
private void getCurrentLocation() {
if (!isGPSEnabled()) {
binding.tvLocation.setText(R.string.gps_is_not_enabled);
return;
} else {
binding.tvLocation.setText(R.string.getting_location);
}
if (ActivityCompat.checkSelfPermission(requireContext(),
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED
&&
```

```
ActivityCompat.checkSelfPermission(requireContext(),
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
return;
}
final int[] numberOfUpdates = {0};
LocationServices.getFusedLocationProviderClient(requireContext())
.requestLocationUpdates(locationRequest, new LocationCallback() {
@Override
public void onLocationResult(@NonNull LocationResult locationResult) {
super.onLocationResult(locationResult);
if (getContext() == null) {
return;
}
numberOfUpdates[0]++;
if (numberOfUpdates[0] >= 3) {
LocationServices.getFusedLocationProviderClient(getContext())
.removeLocationUpdates(this);
if (locationResult.getLocations().size() > 0) {
int idx = locationResult.getLocations().size() - 1;
double latitude = locationResult.getLocations().get(idx).getLatitude();
double longitude = locationResult.getLocations().get(idx).getLongitude();
Geocoder geocoder;
List<Address> addresses;
```

```
geocoder = new Geocoder(getContext(), Locale.getDefault());
try {
addresses = geocoder.getFromLocation(latitude, longitude, 1);
if (addresses != null) {
StringBuilder address = new StringBuilder();
for (int i = 0; i <= addresses.get(0).getMaxAddressLineIndex(); i++) {
address.append(addresses.get(0).getAddressLine(i));
if (i < addresses.get(0).getMaxAddressLineIndex()) {
address.append("\n");
}
}
binding.tvLocation.setText(address.toString());
} else {
binding.tvLocation.setText(getString(R.string.failed_to_get_location));
}
} catch (IOException e) {
binding.tvLocation.setText(getString(R.string.failed_to_get_location));
e.printStackTrace();
}
}
}
}
}, Looper.getMainLooper());
}
```

ActivityMain.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:id="@+id/drawerLayout"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:clipChildren="false"
android:fillViewport="true
android:fitsSystemWindows="false"
tools:context=".ui.activity.MainActivity">
<androidx.fragment.app.FragmentContainerView
android:id="@+id/fragmentContainerView"
android:name="androidx.navigation.fragment.NavHostFragment"
android:layout_width="match_parent"
android:layout_height="match_parent"
app:defaultNavHost="true"
app:navGraph="@navigation/nav_home" />
<com.google.android.material.navigation.NavigationView
android:id="@+id/navView"
android:layout_width="wrap_content"
android:layout_height="match_parent"
```

```
android:layout_gravity="start"
app:headerLayout="@layout/view_header_navigation_drawer"
app:menu="@menu/nav_drawer" />
</androidx.drawerlayout.widget.DrawerLayout>
}
```

Fragment_home.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
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:clipChildren="false"
android:fitsSystemWindows="true"
tools:context=".ui.fragment.HomeFragment">
<androidx.core.widget.NestedScrollView
android:layout_width="match_parent"
android:layout_height="match_parent"
android:fillViewport="true"
app:layout_behavior="com.google.android.material.appbar.AppBarLayout$ScrollingViewBehavior">
<LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
```

```
android:orientation="vertical">
<LinearLayout
android:id="@+id/sos_button"
android:layout_width="280dp"
android:layout_height="280dp"
android:layout_gravity="center"
android:layout_marginVertical="28dp"
android:background="@drawable/btn_large_sos"
android:elevation="12dp"
android:gravity="center"
android:orientation="vertical">
<TextView
android:layout_width="168dp"
android:layout_height="wrap_content"
android:shadowColor="@color/shadow"
android:shadowDx="-4"
android:shadowDy="-4"
android:shadowRadius="4"
android:text="@string/sos"
android:textAlignment="center"
android:textColor="?attr/colorOnError"
android:textSize="60sp" />
</LinearLayout>
<com.google.android.material.button.MaterialButton
```

```
android:id="@+id/btn_shake_detection"
style="@style/Widget.Material3.Button.OutlinedButton"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center"
android:layout_marginBottom="8dp"
android:minHeight="@dimen/button_height"
android:text="@string/btn_start_service"
android:textSize="16sp"
app:cornerRadius="@dimen/component_corner_radius" />
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginHorizontal="@dimen/margin_16"
android:layout_marginTop="@dimen/margin_16"
android:baselineAligned="false"
android:orientation="horizontal">
<LinearLayout
android:id="@+id/contacts"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:layout_marginEnd="@dimen/margin_4"
android:layout_marginBottom="@dimen/margin_8"
android:layout_weight="1"
```

```
android:background="@drawable/container"
android:baselineAligned="false"
android:clickable="true"
android:focusable="true"
android:orientation="vertical"
android:paddingVertical="15dp"
android:paddingStart="20dp"
android:paddingEnd="20dp">
<TextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="@string/activity_contacts_title"
android:textColor="?attr/colorOnSurface"
android:textSize="16sp"
android:textStyle="bold" />
<TextView
android:layout_width="match_parent"
android:layout_height="0dp"
android:layout_weight="1"
android:text="@string/activity_contacts_desc"
android:textColor="?attr/colorOnSurfaceVariant"
android:textSize="15sp"
tools:ignore="NestedWeights" />
<ImageView
```

```
android:layout_width="28dp"
android:layout_height="28dp"
android:layout_gravity="start|bottom"
android:layout_marginTop="@dimen/margin_16"
android:contentDescription="@android:string/untitled"
android:src="@drawable/ic_contacts"
app:tint="?attr/colorPrimary" />
</LinearLayout>
<LinearLayout
android:id="@+id/helpline"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:layout_marginStart="@dimen/margin_4"
android:layout_marginBottom="@dimen/margin_8"
android:layout_weight="1"
android:background="@drawable/container"
android:baselineAligned="false"
android:clickable="true"
android:focusable="true"
android:orientation="vertical
android:paddingVertical="15dp"
android:paddingStart="20dp"
android:paddingEnd="20dp">
<TextView
```

```
android:layout_width="match_parent"

android:layout_height="wrap_content"

android:text="@string/activity_helpline_title"

android:textColor="?attr/colorOnSurface"

android:textSize="16sp"

android:textStyle="bold" />

<TextView

android:layout_width="match_parent"

android:layout_height="0dp"

android:layout_weight="1"

android:text="@string/activity_helpline_desc"

android:textColor="?attr/colorOnSurfaceVariant"

android:textSize="15sp"

tools:ignore="NestedWeights" />

<ImageView

android:layout_width="28dp"

android:layout_height="28dp"

android:layout_gravity="start|bottom"

android:layout_marginTop="@dimen/margin_16"

android:contentDescription="@android:string/untitled"

android:src="@drawable/ic_helpline"

app:tint="?attr/colorPrimary" />

</LinearLayout>

</LinearLayout>
```

```
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginHorizontal="@dimen/margin_16"
android:baselineAligned="false"
android:orientation="horizontal">
<LinearLayout
android:id="@+id/safety_tips"
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_marginEnd="@dimen/margin_4"
android:layout_marginBottom="@dimen/margin_8"
android:layout_weight="1"
android:background="@drawable/container"
android:baselineAligned="false"
android:clickable="true"
android:focusable="true"
android:orientation="vertical"
android:paddingVertical="15dp"
android:paddingStart="20dp"
android:paddingEnd="20dp">
<TextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
```

```
android:text="@string/activity_safety_tips_title"

android:textColor="?attr/colorOnSurface"

android:textSize="16sp"

android:textStyle="bold" />

<TextView

android:layout_width="match_parent"

android:layout_height="0dp"

android:layout_weight="1"

android:text="@string/activity_safety_tips_desc"

android:textColor="?attr/colorOnSurfaceVariant"

android:textSize="15sp"

tools:ignore="NestedWeights" />

<ImageView

android:layout_width="28dp"

android:layout_height="28dp"

android:layout_gravity="start|bottom"

android:layout_marginTop="@dimen/margin_16"

android:contentDescription="@android:string/untitled"

android:src="@drawable/ic_safety_tips"

app:tint="?attr/colorPrimary" />

</LinearLayout>

<LinearLayout

android:id="@+id/about"

android:layout_width="0dp"
```

```
android:layout_height="wrap_content"
android:layout_marginStart="@dimen/margin_4"
android:layout_marginBottom="@dimen/margin_8"
android:layout_weight="1"
android:background="@drawable/container"
android:baselineAligned="false"
android:clickable="true"
android:focusable="true"
android:orientation="vertical"
android:paddingVertical="15dp"
android:paddingStart="20dp"
android:paddingEnd="20dp">
<TextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="@string/activity_about_title"
android:textColor="?attr/colorOnSurface"
android:textSize="16sp"
android:textStyle="bold" />
<TextView
android:layout_width="match_parent"
android:layout_height="0dp"
android:layout_weight="1"
android:text="@string/activity_about_desc"
```

```
android:textColor="?attr/colorOnSurfaceVariant"
android:textSize="15sp"
tools:ignore="NestedWeights" />
<ImageView
android:layout_width="28dp"
android:layout_height="28dp"
android:layout_gravity="start|bottom"
android:layout_marginTop="@dimen/margin_16"
android:contentDescription="@android:string/untitled"
android:src="@drawable/ic_about"
app:tint="?attr/colorPrimary" />
</LinearLayout>
</LinearLayout>
</LinearLayout>
</androidx.core.widget.NestedScrollView>
<include
android:id="@+id/header"
layout="@layout/view_header_expandable" />
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

7. RESULT / OUTPUT SCREENSHOTS

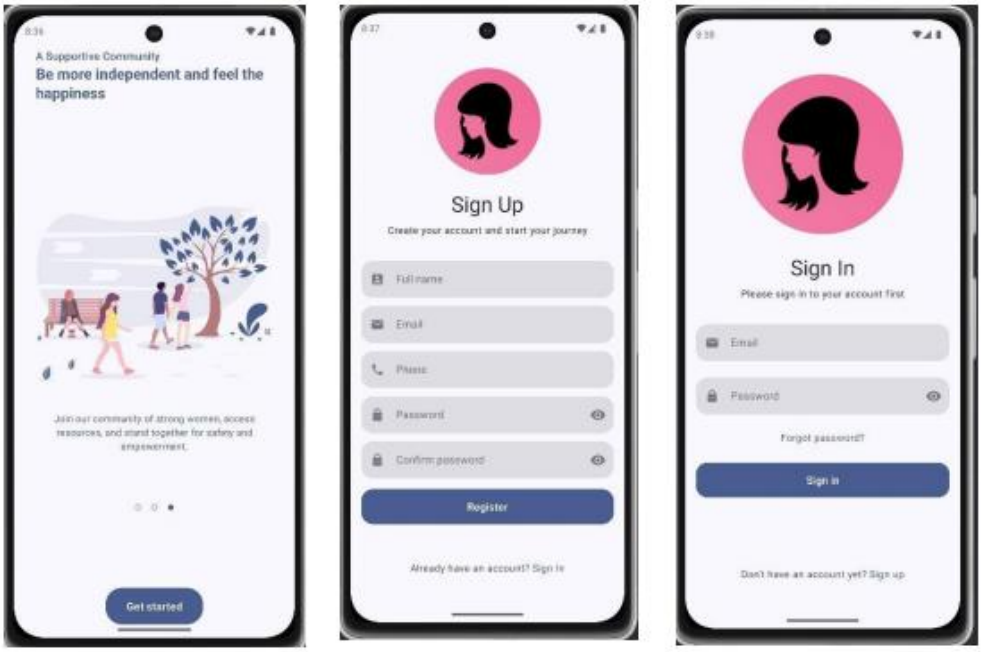


Figure 1: Sign-in and Sign-up page

Figure 2: Home page

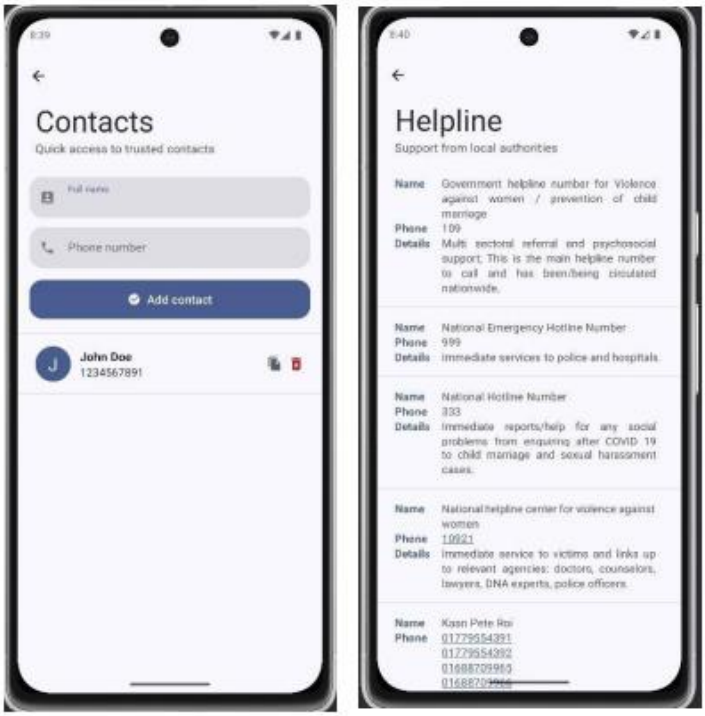


Figure 3: Contacts and Helpline Page

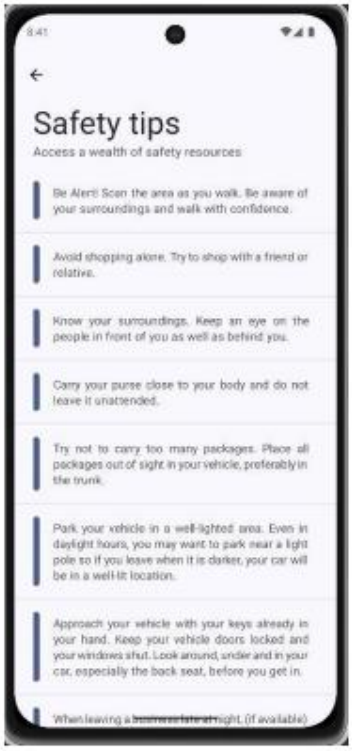


Figure 4: Safety tips Page

8. CONCLUSION

SHEGuard is designed as a powerful digital safety companion for women, helping them stay secure in all kinds of situations. With features like quick SOS alerts, real-time tracking, and proactive safety tools, the app ensures users always have help within reach. Whether someone feels unsafe while walking, traveling, or commuting, SHEGuard provides support, guidance, and immediate assistance.

Beyond emergencies, the app also empowers users with safety knowledge, preparation tips, and preventive measures. With strong privacy controls, reliable encryption, and a user-friendly interface, SHEGuard aims to give women confidence and independence. It is more than an application—it is a step toward creating a safer environment for every woman.

9. REFERENCES

- **Firebase Docs** – https://firebase.google.com/docs
- **Firebase Realtime DB** – https://firebase.google.com/docs/database
- **Android Developer Guide** – https://developer.android.com/guide
- **Android Studio** – https://developer.android.com/studio
- **Flask REST API Tutorial** – https://realpython.com/flask-restful-api/