

ELDERLY CARE COMPANION

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

BY

T. Thanuja	22501A05I2
P. Harshitha	22501A05E5
P. Meghana	22501A05E3
S.Santhi Deepika	22501A05G0
S.Thanu Sri	22501A05H0

Under the Guidance of

Mr. Michael Sadgun Rao Kona

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 “ **Elderly Care Companion**” is the bonafied work of **T. Thanuja (22501A05I2)** , **P. Harshitha (22501A05E5)**, **P. Meghana (22501A05E3)**, **S.Santhi Deepika(22501A05G0)**, **S.Thanu Sri(22501A05H0)** in partial fulfilment of completing the Academic project in MERN Stack Development during the academic year 2024-25.

Signature of the Incharge

Signature of the HOD

INDEX

S.No.	Content	Page No. (s)
1	Abstract	1
2	Introduction	2
3	Objectives and Scope of the Project	3
4	Software used - Explanation	4
5	Proposed model	5-7
6	Sample Code	8-29
7	Result/Output Screen shots	30-31
8	Conclusion	32
9	References (web site URLs)	33

1. ABSTRACT

The **Elderly Care Companion** mobile application is designed to transform senior citizen support by delivering personalized care solutions through advanced technology. Key features include secure login verification to ensure only authorized users seniors and caregivers access the system, comprehensive health monitoring for tracking vital metrics like blood pressure, sugar levels, temperature, and pulse rate, and real-time medication management with caregiver-assigned prescriptions.

The app also facilitates social connectivity by enabling seniors to view and interact with assigned caregivers, addressing loneliness through meaningful engagement. With an emphasis on usability, the interface offers large text and high-contrast design, making it accessible to seniors with varying tech skills.

Beyond health tracking, the application empowers seniors to proactively manage their well-being by integrating personalized features into a cohesive platform. Caregivers can assign tailored medication schedules and monitor health updates, while seniors submit their records and follow individualized care plans. Backed by SQLite for secure data management, the app ensures privacy and reliability across its login, registration, and role-specific dashboards.

By addressing critical aspects of health, mobility, and social interaction, "Elderly Care Companion" aims to enhance seniors' quality of life, fostering healthier, happier aging through cutting-edge, individualized support.

1.1.SDG JUSTIFICATION REPORT

SDG Mapped: SDG 3 – Good Health and Well-being, SDG 4 – Quality Education

1.1.1 How This Project Supports SDG's 3 & 4

The **Elderly Care Companion** project aligns with **SDG 3 (Good Health and Well-being)** and **SDG 4 (Quality Education)** by utilizing technology to enhance elderly individuals' health, emotional well-being, and learning opportunities.

Contribution to SDG 3 (Good Health and Well-being)

Smart Health Monitoring

- Integrates health monitoring sensors to track vital signs such as heart rate and temperature.
- Sends alerts to caregivers or family members in case of anomalies.
- Aids in early detection and prevention of health issues.

Mental and Emotional Well-being

- Provides interactive communication to reduce loneliness and isolation.
- Includes features such as voice assistants, reminders, and emergency alerts.
- Encourages mental stimulation through games and conversational AI.

Promoting Independent Living

- Enables elderly individuals to live independently while staying connected to family.
- Reduces hospital visits through preventive care at home.

Contribution to SDG 4 (Quality Education)

Lifelong Learning Opportunities

- Provides educational content tailored for elderly individuals.
- Encourages cognitive engagement through learning modules and knowledge-sharing platforms.

Digital Literacy and Awareness

- Offers training on using digital tools and communication platforms.
- Enhances accessibility to online resources, improving technological adaptability.

2. INTRODUCTION

The aging population worldwide is growing rapidly, presenting unique challenges in healthcare, social engagement, and daily living support for seniors. Many elderly individuals face difficulties in managing their health conditions, adhering to medication schedules, and maintaining meaningful connections, often due to physical limitations or limited access to technology. Caregivers, meanwhile, struggle to monitor multiple seniors efficiently, lacking tools that provide real-time insights and streamlined communication. The "Elderly Care Companion" mobile application emerges as a response to these challenges, aiming to revolutionize senior care through a technology-driven, user-centric approach.

This project introduces a comprehensive platform designed to empower seniors and their caregivers with personalized, accessible tools. The app features secure login verification to protect user data, enabling seniors to submit vital health metrics—like blood pressure, sugar levels, temperature, and pulse rate—while caregivers assign and track medication schedules. Beyond health management, it fosters social interaction by connecting seniors with caregivers, combating loneliness through structured engagement. By integrating these functionalities, "Elderly Care Companion" seeks to bridge gaps in elderly care, offering a holistic solution that enhances both physical well-being and emotional health.

The significance of this project lies in its potential to transform how seniors and caregivers interact with healthcare technology. It addresses critical pain points—health monitoring, medication adherence, and social isolation—through a single, intuitive app, reducing the burden on caregivers and empowering seniors to take an active role in their care. As a cutting-edge tool, it leverages modern mobile development practices to deliver individualized support, aiming to improve seniors' quality of life and promote healthier aging. This introduction sets the stage for exploring the app's design, implementation, and impact, highlighting its role as a pioneering step toward technology-enabled elderly care.

3. OBJECTIVES AND SCOPE OF THE PROJECT

Objectives :

The "Elderly Care Companion" mobile application is envisioned as a transformative tool to support the aging population by addressing critical gaps in healthcare management and social well-being. With the elderly often facing challenges in tracking health conditions and maintaining connections, and caregivers needing efficient ways to monitor and assist, this project seeks to leverage technology for personalized care.

1. **Enable Health Monitoring:** Allow seniors to record and submit vital health metrics (e.g., blood pressure, sugar levels, temperature, pulse rate) for comprehensive tracking.
2. **Facilitate Medication Management:** Provide caregivers with tools to assign and monitor personalized medication schedules, ensuring timely adherence.
3. **Ensure Secure Access:** Implement login verification to protect user data and restrict access to authorized seniors and caregivers.
4. **Promote Social Connectivity:** Foster interaction between seniors and caregivers to combat loneliness and build supportive relationships.
5. **Enhance User Experience:** Design a senior-friendly interface with large text and high contrast for ease of use across varying tech proficiencies.

Scope :

The scope of the "Elderly Care Companion" project defines the boundaries of its development, focusing on delivering a functional, standalone mobile application tailored to the needs of seniors and caregivers. It encompasses essential features to support health and social care within a local environment, leveraging Android technology for accessibility. While aiming for simplicity and effectiveness, the project sets clear limits to maintain focus and feasibility, excluding complex integrations to prioritize core usability.

4. SOFTWARE USED

1. Android Studio

- Provides an integrated environment for Java coding, XML design, and app testing with tools like Gradle and the Layout Editor.
- Offers integrated debugging features, such as breakpoints and variable inspection, to identify and fix code errors efficiently.
- Supports project management with a built-in resource manager for organizing layouts, drawables, and strings used in the app.



2. SQLite

- Embedded database for storing user credentials, health records, and medication data locally, supporting secure and persistent data management.
- Provides lightweight, serverless storage optimized for mobile devices, requiring minimal setup and maintenance.



3. Android SDK (Software Development Kit)

- Core framework providing libraries and APIs for implementing Android-specific features, UI components, and navigation across activities.
- Includes tools like the Android Emulator and ADB (Android Debug Bridge) for testing and deploying the app on various devices.
- Ensures compatibility with different Android versions, allowing the app to run smoothly across a wide range of devices.



5. PROPOSED MODEL

Proposed Model for Elderly Care Companion

1. User Registration and Login System:

- **User Registration** : Allows seniors and caregivers to register with essential details (email, password, role—Senior Citizen or Caregiver), validating inputs for unique emails and secure storage in SQLite.
- **User Login**: Implements secure authentication using email-password verification, with passwords stored securely in the local database to protect user access.

2. Health Monitoring System:

- **Health Record Submission**: Enables seniors to input vital metrics (blood pressure, sugar levels, temperature, pulse rate) through simple forms, stored locally for tracking.
- **Real-Time Monitoring**: Provides caregivers with access to submitted health records, allowing immediate review and response to seniors' health status.

3. Medication Management System:

- **Medication Assignment**: Permits caregivers to assign medications (name, dose, time) via a spinner-based interface, saved and linked to specific seniors.
- **Medication Viewing**: Displays assigned medications in a list format for seniors, ensuring clarity on dosage and timing to promote adherence.

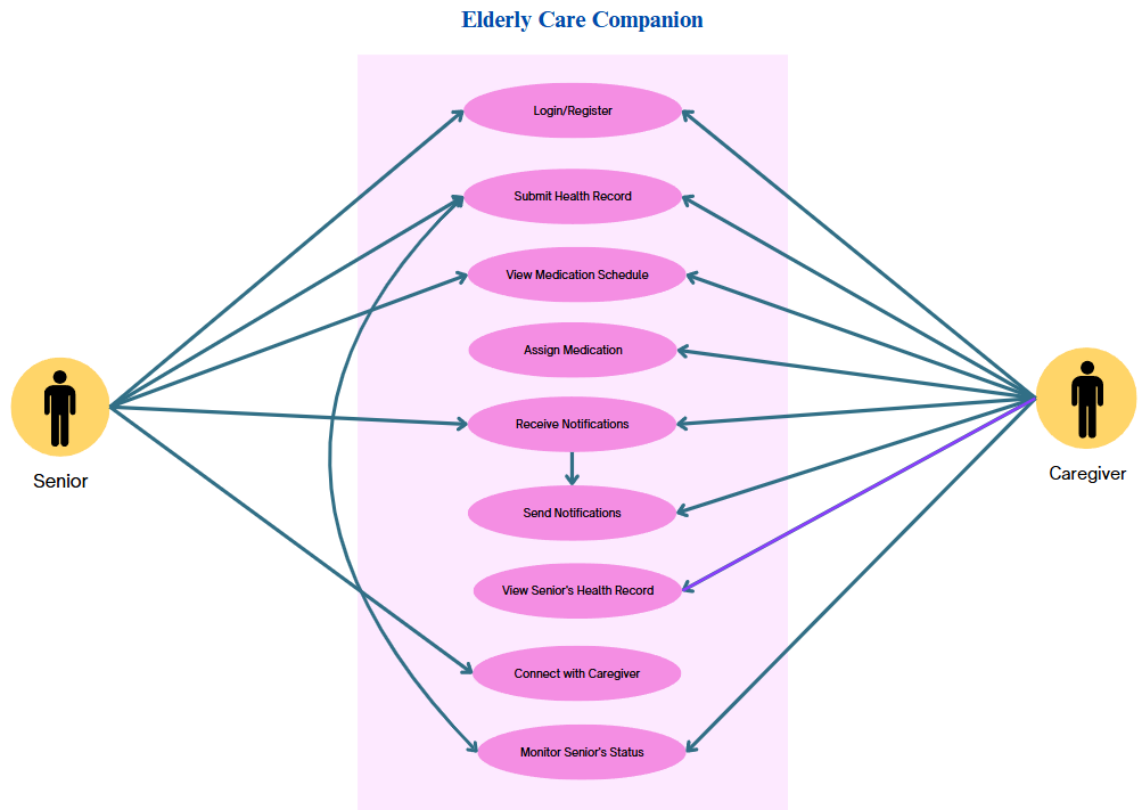
4. Caregiver-Senior Connectivity:

- **Caregiver Assignment**: Allows seniors to select or view their assigned caregiver from a dropdown, fostering direct communication channels.
- **Social Engagement**: Facilitates basic interaction through health update sharing, reducing loneliness by connecting seniors with their caregivers.

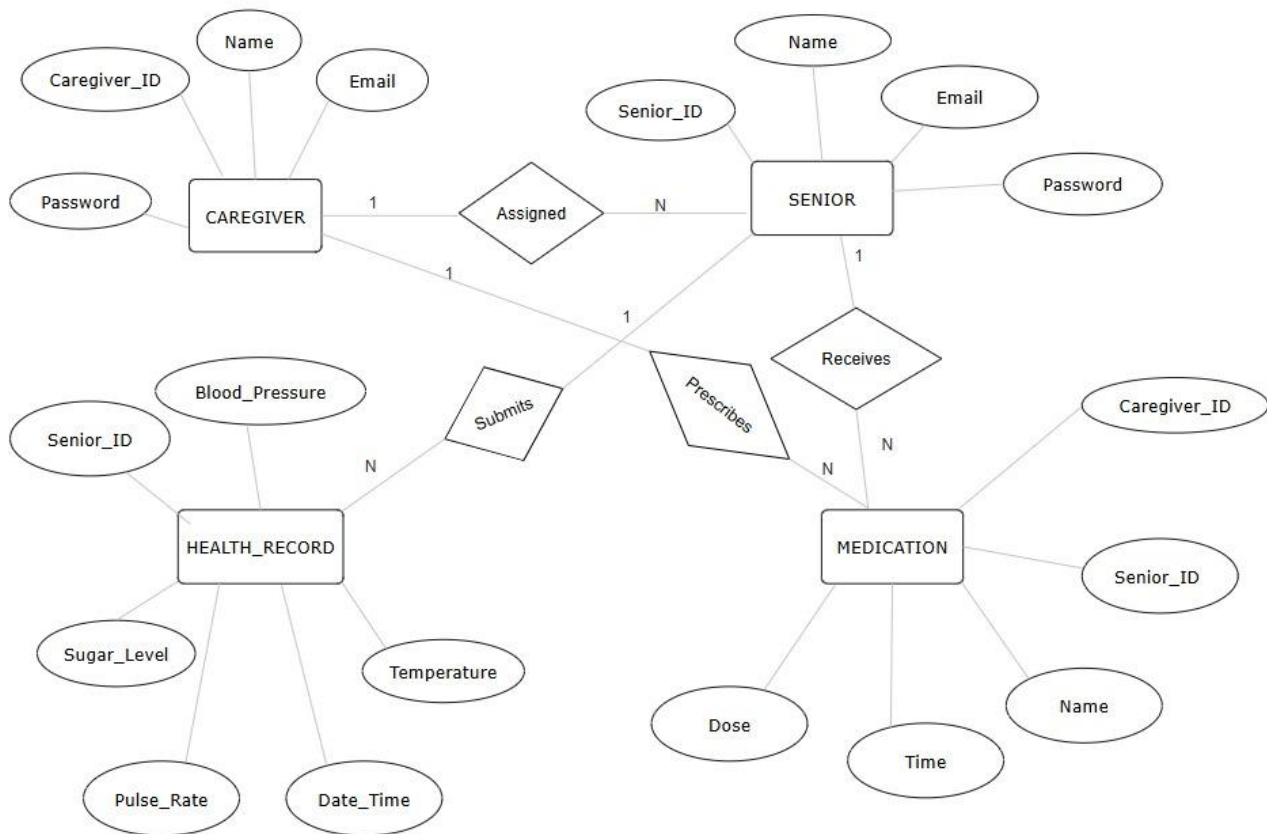
5. User-Friendly and Accessible Design:

- **UI Design**: Features large text, high-contrast visuals, and simplified layouts using Android's built-in resources, optimized for senior usability.
- **Device Compatibility**: Ensures seamless operation on Android devices, focusing on standalone functionality without external dependencies.

UML Diagram:



Entity-Relationship (ER) Diagram:



6. SAMPLE CODE

1. MainActivity.java :

```
package com.example.elderly_care_companion;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.util.Patterns;
import android.widget.Button;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
    private EditText emailField, passwordField;
    private Button loginButton;
    private TextView registerLink;
    private DatabaseHelper dbHelper;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        dbHelper = new DatabaseHelper(this);
        emailField = findViewById(R.id.username);
        passwordField = findViewById(R.id.password);
        loginButton = findViewById(R.id.loginButton);
        registerLink = findViewById(R.id.registerLink);
        loginButton.setOnClickListener(v -> loginUser());
        registerLink.setOnClickListener(v -> {
            Intent intent = new Intent(MainActivity.this, RegistrationActivity.class);
            startActivity(intent);
        });
    }
    private void loginUser() {
        String enteredEmail = emailField.getText().toString().trim();
        String enteredPassword = passwordField.getText().toString().trim();
        if (enteredEmail.isEmpty() || enteredPassword.isEmpty()) {
            showToast("Please fill all fields");
            return;
        }
        if (!Patterns.EMAIL_ADDRESS.matcher(enteredEmail).matches()) {
            showToast("Enter a valid email");
            return;
        }
        Log.d("LOGIN", "Attempting login for: " + enteredEmail);
        String role = dbHelper.getUserRole(enteredEmail, enteredPassword);
        if (role != null) {
            Log.d("LOGIN", "Retrieved role: " + role);
            showToast("Login Successful");
            Intent intent;
            role = role.toLowerCase().trim();
            if (role.equals("senior citizen")) {
```

```

intent = new Intent(MainActivity.this, SeniorCitizen.class);
} else if (role.equals("caregiver")) {
    intent = new Intent(MainActivity.this, CareGiver.class);
} else {
    showToast("Invalid Role: " + role);
    return;
}
intent.putExtra("USER_EMAIL", enteredEmail);
startActivity(intent);
Log.d("LOGIN", "Navigating to: " + intent.getComponent().getClassName());
finish();
} else {
    Log.e("LOGIN_FAILED", "Invalid credentials for email: " + enteredEmail);
    showToast("Invalid Credentials");
}
}
private void showToast(String message) {
    Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
}
}

```

2. activity_main.xml :

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="24dp"
    android:background="#E6F0FA"
    android:gravity="center_horizontal">
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="150dp"
        android:layout_marginBottom="64dp"
        android:gravity="center"
        android:text="Elderly Care Companion"
        android:textColor="#4A90E2"
        android:textSize="36sp"
        android:textStyle="bold" />
    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:hint="Email"
        android:inputType="textEmailAddress"
        android:padding="14dp"
        android:background="#FFFFFF"
        android:textColor="#212121"

```

```

        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
<EditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="24dp"
    android:hint="Password"
    android:inputType="textPassword"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<Button
    android:id="@+id/loginButton"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:layout_marginBottom="24dp"
    android:padding="14dp"
    android:text="Login"
    android:textSize="18sp"
    android:textAllCaps="false"
    android:background="#4A90E2"
    android:textColor="#FFFFFF" />
<TextView
    android:id="@+id/registerLink"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:text="New User? Register Here"
    android:textColor="#90CAF9"
    android:textSize="16sp"
    android:padding="12dp"
    android:gravity="center" />
</LinearLayout>

```

3. **RegistrationActivity.java :**

```

package com.example.elderly_care_companion;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioGroup;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class RegistrationActivity extends AppCompatActivity {
    private EditText name, email, password;
    private RadioGroup roleGroup;

```

```

private Button registerButton;
private DatabaseHelper dbHelper;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_registration);
    dbHelper = new DatabaseHelper(this);
    name = findViewById(R.id.name);
    email = findViewById(R.id.email);
    password = findViewById(R.id.password);
    roleGroup = findViewById(R.id.roleGroup);
    registerButton = findViewById(R.id.registerButton);
    registerButton.setOnClickListener(v -> registerUser());
}
private void registerUser() {
    String userName = name.getText().toString().trim();
    String userEmail = email.getText().toString().trim();
    String userPassword = password.getText().toString().trim();
    int selectedRoleId = roleGroup.getCheckedRadioButtonId();
    String userRole = (selectedRoleId == R.id.seniorCitizen) ? "Senior Citizen" : "Caregiver";
    if (userName.isEmpty() || userEmail.isEmpty() || userPassword.isEmpty()) {
        Toast.makeText(this, "Please fill all fields", Toast.LENGTH_SHORT).show();
        return;
    }
    if (dbHelper.registerUser(userName, userEmail, userPassword, userRole)) {
        Toast.makeText(this, "Registration Successful", Toast.LENGTH_SHORT).show();
        Intent intent = new Intent(RegistrationActivity.this, MainActivity.class);
        startActivity(intent);
        finish();
    } else {
        Toast.makeText(this, "Registration Failed", Toast.LENGTH_SHORT).show();
    }
}
}

```

4. activity_registration.xml :

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="24dp"
    android:background="#E6F0FA"
    android:gravity="center_horizontal">
    <EditText
        android:id="@+id/userId"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="150dp"
        android:layout_marginBottom="16dp"

```

```

        android:background="#FFFFFF"
        android:hint="Enter User ID"
        android:padding="14dp"
        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
<EditText
    android:id="@+id/password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Password"
    android:inputType="textPassword"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<EditText
    android:id="@+id/phone"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Phone Number"
    android:inputType="phone"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<EditText
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<EditText
    android:id="@+id/address"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Address"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"

```



```

        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
    <EditText
        android:id="@+id/age"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="24dp"
        android:hint="Age"
        android:inputType="number"
        android:padding="14dp"
        android:background="#FFFFFF"
        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
    <RadioGroup
        android:id="@+id/roleGroup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="24dp"
        android:orientation="horizontal"
        android:gravity="center_horizontal">
        <RadioButton
            android:id="@+id/seniorCitizen"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="16dp"
            android:text="Senior Citizen"
            android:textColor="#4A90E2"
            android:textSize="16sp" />
        <RadioButton
            android:id="@+id/careGiver"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Care Giver"
            android:textColor="#4A90E2"
            android:textSize="16sp" />
    </RadioGroup>
    <Button
        android:id="@+id/registerButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="16dp"
        android:text="Register"
        android:textSize="18sp"
        android:textAllCaps="false"
        android:background="#4A90E2"
        android:textColor="#FFFFFF" />

</LinearLayout>

```

5.SeniorCitizen.java :

```
package com.example.elderly_care_companion;
import android.app.TimePickerDialog;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
public class SeniorCitizen extends AppCompatActivity {
    private EditText bp, sugarLevel, temperature, pulseRate, doseLevel;
    private TextView caregiverId, medicationTime;
    private Spinner medicationSpinner, caregiverSpinner;
    private Button submitButton, pickTimeButton, viewMedicationsButton;
    private DatabaseHelper dbHelper;
    private String selectedTime = "";
    private String selectedCaregiverName;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_senior_citizen);
        dbHelper = new DatabaseHelper(this);
        userEmail = getIntent().getStringExtra("USER_EMAIL");
        if (userEmail == null) {
            Log.e("SeniorCitizen", "USER_EMAIL is null");
            Toast.makeText(this, "Error: User email not found", Toast.LENGTH_SHORT).show();
            finish();
            return;
        }
        bp = findViewById(R.id.bp);
        sugarLevel = findViewById(R.id.sugarLevel);
        temperature = findViewById(R.id.temperature);
        pulseRate = findViewById(R.id.pulseRate);
        doseLevel = findViewById(R.id.doseLevel);
        medicationTime = findViewById(R.id.medicationTime);
        pickTimeButton = findViewById(R.id.pickTimeButton);
        submitButton = findViewById(R.id.submitHealthRecord);
        viewMedicationsButton = findViewById(R.id.viewMedicationsButton);
        medicationSpinner = findViewById(R.id.medicationSpinner);
        caregiverSpinner = findViewById(R.id.caregiverSpinner);
        String[] medications = {
            "Aspirin", "Paracetamol", "Metformin", "Insulin", "Atorvastatin",
            "Amlodipine", "Furosemide", "Omeprazole", "Lansoprazole", "Simvastatin"
        };
    }
```

```

        ArrayAdapter<String> medicationAdapter = new ArrayAdapter<>(this,
android.R.layout.simple_spinner_dropdown_item, medications);
        medicationSpinner.setAdapter(medicationAdapter);
        List<String> caregivers = dbHelper.getCaregivers();
        if (caregivers.isEmpty()) {
            caregivers.add("No caregivers available");
        }
        ArrayAdapter<String> caregiverAdapter = new ArrayAdapter<>(this,
android.R.layout.simple_spinner_dropdown_item, caregivers);
        caregiverSpinner.setAdapter(caregiverAdapter);
        caregiverSpinner.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                selectedCaregiverName = parent.getItemAtPosition(position).toString();
                Log.d("SeniorCitizen", "Selected caregiver: " + selectedCaregiverName);
            }
            @Override
            public void onNothingSelected(AdapterView<?> parent) {
                selectedCaregiverName = null;
                Log.d("SeniorCitizen", "No caregiver selected");
            }
        });
        pickTimeButton.setOnClickListener(v -> showTimePicker());
        submitButton.setOnClickListener(v -> saveHealthRecord());
        viewMedicationsButton.setOnClickListener(v -> {
            Intent intent = new Intent(SeniorCitizen.this, SeniorCitizenViewMedicationsActivity.class);
            intent.putExtra("USER_EMAIL", userEmail);
            startActivity(intent);
        });
    }
    private void showTimePicker() {
        Calendar calendar = Calendar.getInstance();
        int hour = calendar.get(Calendar.HOUR_OF_DAY);
        int minute = calendar.get(Calendar.MINUTE);
        TimePickerDialog timePickerDialog = new TimePickerDialog(this, (view, hourOfDay, minuteOfHour) -> {
            selectedTime = String.format("%02d:%02d", hourOfDay, minuteOfHour);
            medicationTime.setText(selectedTime);
        }, hour, minute, true);
        timePickerDialog.show();
    }
    dbHelper.assignCaregiverToSenior(userEmail, caregiverEmail);
    HealthRecord record = new HealthRecord(userEmail, bpValue, sugar, temp, pulse, medication, dose,
time, "Pending", date, caregiverEmail);
    boolean success = dbHelper.insertHealthRecord(record);
    if (success) {
        Log.d("SeniorCitizen", "Health record submitted successfully for user: " + userEmail);
        Toast.makeText(this, "Health Record Submitted", Toast.LENGTH_SHORT).show();
    } else {
        Log.e("SeniorCitizen", "Failed to submit health record for user: " + userEmail + ", caregiver: " +
caregiverEmail);
        Toast.makeText(this, "Failed to submit health record", Toast.LENGTH_SHORT).show();
    }
}

```

```

    }
}
}

```

6. activity_senior_citizen.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="24dp"
    android:background="#E6F0FA"
    android:gravity="center_horizontal">
    <EditText
        android:id="@+id/bp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="125dp"
        android:layout_marginBottom="16dp"
        android:background="#FFFFFF"
        android:hint="Blood Pressure"
        android:padding="14dp"
        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
    <EditText
        android:id="@+id/sugarLevel"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:hint="Sugar Level"
        android:padding="14dp"
        android:background="#FFFFFF"
        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
    <EditText
        android:id="@+id/temperature"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:hint="Temperature"
        android:padding="14dp"
        android:background="#FFFFFF"
        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
    <EditText
        android:id="@+id/pulseRate"
        android:layout_width="match_parent"

```

```

    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Pulse Rate"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<EditText
    android:id="@+id/doseLevel"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Dose Level"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<Spinner
    android:id="@+id/medicationSpinner"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:background="#FFFFFF"
    android:padding="10dp"
    android:spinnerMode="dropdown" />
<TextView
    android:id="@+id/medicationTime"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    android:text="Select Medication Time"
    android:textSize="16sp"
    android:textColor="#4A90E2"
    android:padding="8dp" />
<Button
    android:id="@+id/pickTimeButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:background="#4A90E2"
    android:padding="10dp"
    android:text="Pick Time"
    android:textAllCaps="false"
    android:textColor="#FFFFFF"
    android:textSize="14sp" />
<TextView
    android:id="@+id/caregiverId"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

```

```

        android:layout_marginBottom="8dp"
        android:text="Caregiver ID"
        android:textColor="#4A90E2"
        android:textSize="16sp" />
    <Button
        android:id="@+id/submitHealthRecord"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:padding="16dp"
        android:text="Submit Health Record"
        android:textSize="18sp"
        android:textAllCaps="false"
        android:background="#4A90E2"
        android:textColor="#FFFFFF" />
    <Button
        android:id="@+id/viewMedicationsButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="12dp"
        android:text="View Assigned Medications"
        android:textSize="16sp"
        android:textAllCaps="false"
        android:background="#4A90E2"
        android:textColor="#FFFFFF" />

</LinearLayout>

```

7. CareGiver.java :

```

package com.example.elderly_care_companion;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.util.ArrayList;
import java.util.List;
public class CareGiver extends AppCompatActivity {
    private static final String TAG = "CareGiver";
    private TextView seniorId;
    private EditText prescribeMedication, doseLevel, prescribeTime;
    private Spinner seniorListSpinner;
    private Button sendNotificationButton, viewSeniorsButton;
    private ListView assignedSeniorsListView;
    private DatabaseHelper dbHelper;
    private String userEmail;

```

```

private boolean isInitialLoad = true;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_care_giver);
    Log.d(TAG, "onCreate started");
    try {
        dbHelper = new DatabaseHelper(this);
        userEmail = getIntent().getStringExtra("USER_EMAIL");
        Log.d(TAG, "User email: " + userEmail);
        if (userEmail == null) {
            Log.e(TAG, "USER_EMAIL is null");
            Toast.makeText(this, "Error: User email not found", Toast.LENGTH_SHORT).show();
            finish();
            return;
        }
        seniorId = findViewById(R.id.seniorId);
        prescribeMedication = findViewById(R.id.prescribeMedication);
        doseLevel = findViewById(R.id.doseLevel);
        prescribeTime = findViewById(R.id.prescribeTime);
        seniorListSpinner = findViewById(R.id.seniorListSpinner);
        sendNotificationButton = findViewById(R.id.sendNotificationButton);
        viewSeniorsButton = findViewById(R.id.viewSeniorsButton);
        assignedSeniorsListView = findViewById(R.id.assignedSeniorsListView);
        Log.d(TAG, "UI components initialized");
        loadSeniors();
        List<String> seniors = dbHelper.getAssignedSeniors(userEmail);
        Log.d(TAG, "Assigned seniors count: " + seniors.size());
        ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1,
seniors);
        assignedSeniorsListView.setAdapter(adapter);
        seniorListSpinner.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                if (isInitialLoad) {
                    isInitialLoad = false;
                    Log.d(TAG, "Skipping initial spinner selection");
                    return;
                }
                String selectedSeniorEmail = parent.getItemAtPosition(position).toString();
                Log.d(TAG, "Selected senior: " + selectedSeniorEmail);
                Intent intent = new Intent(CareGiver.this, CareGiverViewSeniorsActivity.class);
                intent.putExtra("USER_EMAIL", userEmail);
                intent.putExtra("SELECTED_SENIOR_EMAIL", selectedSeniorEmail);
                startActivity(intent);
            }
            public void onNothingSelected(AdapterView<?> parent) {
                Log.d(TAG, "No senior selected in spinner");
            }
        });
    }
}

```



```

        sendNotificationButton.setOnClickListener(v -> sendNotification());
        viewSeniorsButton.setOnClickListener(v -> {
            Log.d(TAG, "View Assigned Seniors button clicked");
            Intent intent = new Intent(CareGiver.this, CareGiverViewSeniorsActivity.class);
            intent.putExtra("USER_EMAIL", userEmail);
            startActivity(intent);
            Log.d(TAG, "Intent launched for CareGiverViewSeniorsActivity");
        });
        Log.d(TAG, "onCreate completed successfully");
    } catch (Exception e) {
        Log.e(TAG, "Error in onCreate: " + e.getMessage(), e);
        Toast.makeText(this, "An error occurred: " + e.getMessage(), Toast.LENGTH_LONG).show();
        finish();
    }
}

private void loadSeniors() {
    try {
        List<String> seniors = new ArrayList<>(dbHelper.getAssignedSeniors(userEmail));
        Log.d(TAG, "Loading seniors, count: " + seniors.size());
        if (seniors.isEmpty()) {
            Toast.makeText(this, "No seniors assigned!", Toast.LENGTH_SHORT).show();
            return;
        }
        ArrayAdapter<String> adapter = new ArrayAdapter<>(this,
        android.R.layout.simple_spinner_dropdown_item, seniors);
        seniorListSpinner.setAdapter(adapter);
    } catch (Exception e) {
        Log.e(TAG, "Error loading seniors: " + e.getMessage(), e);
        Toast.makeText(this, "Failed to load seniors", Toast.LENGTH_SHORT).show();
    }
}

private void sendNotification() {
    try {
        String selectedSenior = seniorListSpinner.getSelectedItem() != null ?
        seniorListSpinner.getSelectedItem().toString() : "";
        String medication = prescribeMedication.getText().toString().trim();
        String dose = doseLevel.getText().toString().trim();
        String time = prescribeTime.getText().toString().trim();
        if (selectedSenior.isEmpty() || medication.isEmpty() || dose.isEmpty() || time.isEmpty()) {
            Toast.makeText(this, "Please fill all fields!", Toast.LENGTH_SHORT).show();
            return;
        }
        boolean success = dbHelper.insertPrescription(selectedSenior, medication, dose, time);
        if (success) {
            Toast.makeText(this, "Notification sent!", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(this, "Failed to send notification.", Toast.LENGTH_SHORT).show();
        }
    } catch (Exception e) {
        Log.e(TAG, "Error sending notification: " + e.getMessage(), e);
        Toast.makeText(this, "Error sending notification", Toast.LENGTH_SHORT).show();
    }
}

```



```

    }
}
}

```

8. activity_care_giver.xml :

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="24dp"
    android:background="#E6F0FA"
    android:gravity="center_horizontal">
    <TextView
        android:id="@+id/seniorId"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="125dp"
        android:layout_marginBottom="16dp"
        android:text="Senior ID: "
        android:textSize="18sp"
        android:textStyle="bold"
        android:textColor="#4A90E2" />
    <TextView
        android:id="@+id/seniorHealthDetails"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:text="Health Details: "
        android:textSize="16sp"
        android:textColor="#4A90E2" />
    <EditText
        android:id="@+id/prescribeMedication"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:hint="Prescribe Medication"
        android:padding="14dp"
        android:background="#FFFFFF"
        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
    <EditText
        android:id="@+id/doseLevel"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:hint="Dose Level"
        android:padding="14dp"
        android:background="#FFFFFF"

```

```

        android:textColor="#212121"
        android:textColorHint="#90CAF9"
        android:textSize="16sp" />
<EditText
    android:id="@+id/prescribeTime"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:hint="Prescribed Time (HH:MM)"
    android:padding="14dp"
    android:background="#FFFFFF"
    android:textColor="#212121"
    android:textColorHint="#90CAF9"
    android:textSize="16sp" />
<Button
    android:id="@+id/sendNotificationButton"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:padding="16dp"
    android:text="Send Notification"
    android:textSize="18sp"
    android:textAllCaps="false"
    android:background="#4A90E2"
    android:textColor="#FFFFFF" />
<Button
    android:id="@+id/viewSeniorsButton"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:padding="16dp"
    android:text="View Assigned Seniors"
    android:textSize="18sp"
    android:textAllCaps="false"
    android:background="#4A90E2"
    android:textColor="#FFFFFF" />
</LinearLayout>

```

9. DatabaseHelper.java :

```

package com.example.elderly_care_companion;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import java.util.ArrayList;
import java.util.List;

```

```

public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "ElderlyCare.db";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_USERS = "users";
    private static final String COLUMN_ID = "id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_EMAIL = "email";
    private static final String COLUMN_PASSWORD = "password";
    private static final String COLUMN_ROLE = "role";
    private static final String COLUMN_ASSIGNED_CAREGIVER = "assigned_caregiver";
    private static final String TABLE_HEALTH = "health_records";
    private static final String COLUMN_USER_ID = "user_id";
    private static final String COLUMN_BP = "bp";
    private static final String COLUMN_SUGAR = "sugar_level";
    private static final String COLUMN_TEMPERATURE = "temperature";
    private static final String COLUMN_PULSE = "pulse_rate";
    private static final String COLUMN_MEDICATION = "medication";
    private static final String COLUMN_DOSE = "dose_level";
    private static final String COLUMN_MEDICATION_TIME = "medication_time";
    private static final String COLUMN_STATUS = "status";
    private static final String COLUMN_DATE = "record_date";
    private static final String COLUMN_CAREGIVER = "caregiver_id";
    private static final String TABLE_PRESCRIPTIONS = "prescriptions";
    private static final String COLUMN_PRESCRIPTION_SENIOR_ID = "seniorId";
    private static final String COLUMN_PRESCRIPTION_MEDICATION = "medication";
    private static final String COLUMN_PRESCRIPTION_DOSE = "dose";
    private static final String COLUMN_PRESCRIPTION_TIME = "time";
    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String createUsersTable = "CREATE TABLE " + TABLE_USERS + " (" +
            COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
            COLUMN_NAME + " TEXT, " +
            COLUMN_EMAIL + " TEXT UNIQUE, " +
            COLUMN_PASSWORD + " TEXT, " +
            COLUMN_ROLE + " TEXT, " +
            COLUMN_ASSIGNED_CAREGIVER + " TEXT)";
        String createHealthTable = "CREATE TABLE " + TABLE_HEALTH + " (" +
            COLUMN_USER_ID + " TEXT, " +
            COLUMN_BP + " TEXT, " +
            COLUMN_SUGAR + " TEXT, " +
            COLUMN_TEMPERATURE + " TEXT, " +
            COLUMN_PULSE + " TEXT, " +
            COLUMN_MEDICATION + " TEXT, " +
            COLUMN_DOSE + " TEXT, " +
            COLUMN_MEDICATION_TIME + " TEXT, " +
            COLUMN_STATUS + " TEXT, " +
            COLUMN_DATE + " TEXT, " +

```

```

        COLUMN_CAREGIVER + " TEXT)";
String createPrescriptionsTable = "CREATE TABLE " + TABLE_PRESCRIPTIONS + " (" +
        COLUMN_PRESCRIPTION_SENIOR_ID + " TEXT, " +
        COLUMN_PRESCRIPTION_MEDICATION + " TEXT, " +
        COLUMN_PRESCRIPTION_DOSE + " TEXT, " +
        COLUMN_PRESCRIPTION_TIME + " TEXT)";
db.execSQL(createUsersTable);
db.execSQL(createHealthTable);
db.execSQL(createPrescriptionsTable);
}
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_USERS);
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_HEALTH);
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_PRESCRIPTIONS);
    onCreate(db);
}
public boolean registerUser(String name, String email, String password, String role) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COLUMN_NAME, name);
    values.put(COLUMN_EMAIL, email);
    values.put(COLUMN_PASSWORD, password);
    values.put(COLUMN_ROLE, role);
    values.putNull(COLUMN_ASSIGNED_CAREGIVER);
    long result = db.insert(TABLE_USERS, null, values);
    db.close();
    return result != -1;
}
public List<String> getCaregivers() {
    List<String> caregivers = new ArrayList<>();
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT " + COLUMN_EMAIL + " FROM " + TABLE_USERS + " WHERE " +
COLUMN_ROLE + " ='Caregiver'", null);
    if (cursor.moveToFirst()) {
        do {
            caregivers.add(cursor.getString(0));
        } while (cursor.moveToNext());
    }
    cursor.close();
    db.close();
    Log.d("DatabaseHelper", "Fetched caregivers: " + caregivers.toString());
    return caregivers;
}

public String getUserRole(String email, String password) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.query(TABLE_USERS, new String[]{COLUMN_ROLE}, COLUMN_EMAIL + "=? AND " +
COLUMN_PASSWORD + "=?",
        new String[]{email, password}, null, null, null);

```

```

    if (cursor != null && cursor.moveToFirst()) {
        String role = cursor.getString(0);
        cursor.close();
        db.close();
        return role;
    }
    if (cursor != null) cursor.close();
    db.close();
    return null;
}

public boolean insertHealthRecord(HealthRecord record) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COLUMN_USER_ID, record.getUserId());
    values.put(COLUMN_BP, record.getBp());
    values.put(COLUMN_SUGAR, record.getSugarLevel());
    values.put(COLUMN_TEMPERATURE, record.getTemperature());
    values.put(COLUMN_PULSE, record.getPulseRate());
    values.put(COLUMN_MEDICATION, record.getMedications());
    values.put(COLUMN_DOSE, record.getDoseLevels());
    values.put(COLUMN_MEDICATION_TIME, record.getMedicationTime());
    values.put(COLUMN_STATUS, record.getStatus());
    values.put(COLUMN_DATE, record.getDate());
    values.put(COLUMN_CAREGIVER, record.getCaregiverId());

    Log.d("DatabaseHelper", "Inserting health record - Senior: " + record.getUserId() + ", Caregiver: " +
record.getCaregiverId() + ", Date: " + record.getDate());
    long result = -1;
    try {
        result = db.insertOrThrow(TABLE_HEALTH, null, values);
        Log.d("DatabaseHelper", "Health record insert result: " + result + " for senior: " +
record.getUserId());
    } catch (SQLException e) {
        Log.e("DatabaseHelper", "Error inserting health record: " + e.getMessage());
        e.printStackTrace();
    } finally {
        db.close();
    }
    return result != -1;
}

public String getCaregiverIdForSenior(String userEmail) { // Renamed for clarity
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT " + COLUMN_ASSIGNED_CAREGIVER + " FROM " +
TABLE_USERS + " WHERE " + COLUMN_EMAIL + "=?", new String[]{userEmail});

    if (cursor != null && cursor.moveToFirst()) {
        String caregiverEmail = cursor.getString(0);
        cursor.close();
        db.close();
    }
}

```

```

        return caregiverEmail != null ? caregiverEmail : "N/A";
    }
    if (cursor != null) cursor.close();
    db.close();
    return "N/A";
}

public List<String> getAssignedSeniors(String caregiverEmail) {
    List<String> seniors = new ArrayList<>();
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT DISTINCT " + COLUMN_USER_ID + " FROM " + TABLE_HEALTH
+
    " WHERE " + COLUMN_CAREGIVER + "=?", new String[]{caregiverEmail});

    if (cursor.moveToFirst()) {
        do {
            seniors.add(cursor.getString(0));
        } while (cursor.moveToNext());
    }
    cursor.close();
    db.close();
    Log.d("DatabaseHelper", "Fetched assigned seniors for caregiver " + caregiverEmail + ": " +
seniors.toString());
    return seniors;
}

public List<HealthRecord> getHealthRecordsForSenior(String seniorEmail) {
    List<HealthRecord> records = new ArrayList<>();
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.rawQuery("SELECT * FROM " + TABLE_HEALTH + " WHERE " + COLUMN_USER_ID
+ "=?", new String[]{seniorEmail});

    if (cursor.moveToFirst()) {
        do {
            HealthRecord record = new HealthRecord(
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_USER_ID)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_BP)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_SUGAR)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_TEMPERATURE)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_PULSE)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_MEDICATION)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_DOSE)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_MEDICATION_TIME)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_STATUS)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_DATE)),
                cursor.getString(cursor.getColumnIndexOrThrow(COLUMN_CAREGIVER))
            );
            records.add(record);
        } while (cursor.moveToNext());
    }
    cursor.close();
}

```

```

        db.close();
        return records;
    }

    public List<String> getAssignedMedications(String seniorEmail) {
        List<String> medications = new ArrayList<>();
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT " + COLUMN_PRESCRIPTION_MEDICATION + ", " +
            COLUMN_PRESCRIPTION_DOSE + ", " + COLUMN_PRESCRIPTION_TIME +
            " FROM " + TABLE_PRESCRIPTIONS + " WHERE " + COLUMN_PRESCRIPTION_SENIOR_ID + "=?",
            new String[]{seniorEmail});
        if (cursor.moveToFirst()) {
            do {
                String med = cursor.getString(0);
                String dose = cursor.getString(1);
                String time = cursor.getString(2);
                medications.add("Medication: " + med + ", Dose: " + dose + ", Time: " + time);
            } while (cursor.moveToNext());
        }
        cursor.close();
        db.close();
        return medications;
    }

    public boolean insertPrescription(String seniorEmail, String medication, String dose, String time) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_PRESCRIPTION_SENIOR_ID, seniorEmail);
        values.put(COLUMN_PRESCRIPTION_MEDICATION, medication);
        values.put(COLUMN_PRESCRIPTION_DOSE, dose);
        values.put(COLUMN_PRESCRIPTION_TIME, time);

        long result = db.insert(TABLE_PRESCRIPTIONS, null, values);
        db.close();
        return result != -1;
    }

    public void assignCaregiverToSenior(String seniorEmail, String caregiverEmail) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_ASSIGNED_CAREGIVER, caregiverEmail);
        db.update(TABLE_USERS, values, COLUMN_EMAIL + "=?", new String[]{seniorEmail});
        db.close();
    }

    public String getCaregiverEmailByName(String caregiverName) { // Changed to return email
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT " + COLUMN_EMAIL + " FROM " + TABLE_USERS + " WHERE " +
            COLUMN_NAME + "=? AND " + COLUMN_ROLE + "='Caregiver'", new String[]{caregiverName});

        if (cursor != null && cursor.moveToFirst()) {
            String caregiverEmail = cursor.getString(0);
            cursor.close();
            db.close();
        }
    }

```

```

        return caregiverEmail;
    }
    if (cursor != null) cursor.close();
    db.close();
    return "N/A";
}
}

```

10 . HealthRecord.java :

```
package com.example.elderly_care_companion;
```

```

public class HealthRecord {
    private String userId, bp, sugarLevel, temperature, pulseRate, medications, doseLevels, medicationTime,
    status, date, caregiverId;

    public HealthRecord(String userId, String bp, String sugarLevel, String temperature, String pulseRate,
        String medications, String doseLevels, String medicationTime, String status, String date, String
        caregiverId) {
        this.userId = userId;
        this.bp = bp;
        this.sugarLevel = sugarLevel;
        this.temperature = temperature;
        this.pulseRate = pulseRate;
        this.medications = medications;
        this.doseLevels = doseLevels;
        this.medicationTime = medicationTime;
        this.status = status;
        this.date = date;
        this.caregiverId = caregiverId;
    }
    public String getUserId() { return userId; }
    public String getBp() { return bp; }
    public String getSugarLevel() { return sugarLevel; }
    public String getTemperature() { return temperature; }
    public String getPulseRate() { return pulseRate; }
    public String getMedications() { return medications; }
    public String getDoseLevels() { return doseLevels; }
    public String getMedicationTime() { return medicationTime; }
    public String getStatus() { return status; }
    public String getDate() { return date; }
    public String getCaregiverId() { return caregiverId; }
}

```

11. SeniorCitizenViewMedicationsActivity.java :

```
package com.example.elderly_care_companion;
```

```

import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;

```



```

import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.util.List;

public class SeniorCitizenViewMedicationsActivity extends AppCompatActivity {
    private ListView medicationsListView;
    private DatabaseHelper dbHelper;
    private String seniorEmail; // Changed from seniorId

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

        medicationsListView = findViewById(R.id.medicationsListView);
        seniorEmail = getIntent().getStringExtra("USER_EMAIL");

        dbHelper = new DatabaseHelper(this);
        List<String> medications = dbHelper.getAssignedMedications(seniorEmail);

        if (medications.isEmpty()) {
            Toast.makeText(this, "No medications assigned!", Toast.LENGTH_SHORT).show();
        } else {
            ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1,
medications);
            medicationsListView.setAdapter(adapter);
        }
    }
}

```

7. RESULT /OUTPUT SCREENSHOTS

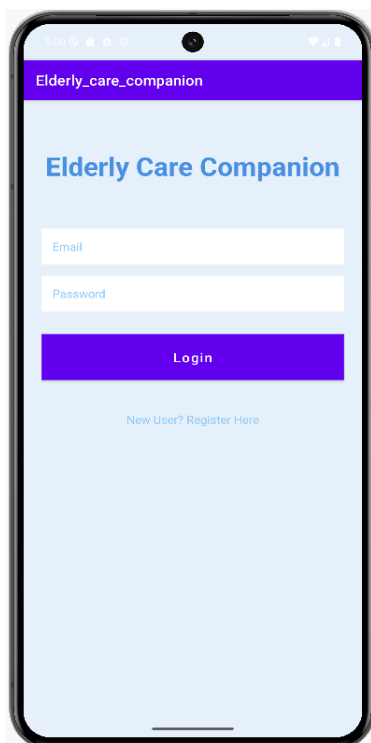


Fig 1: Login

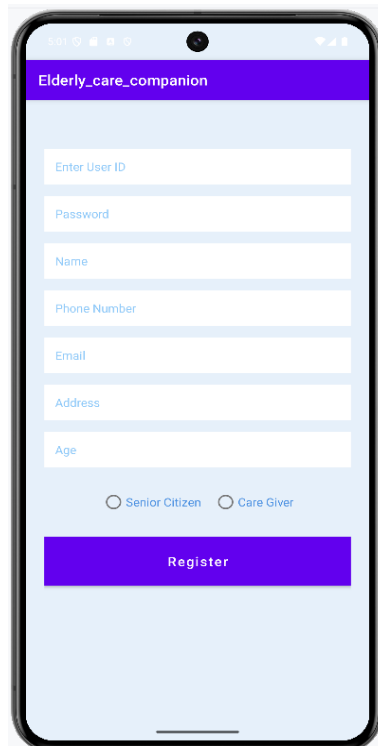


Fig 2: Registration

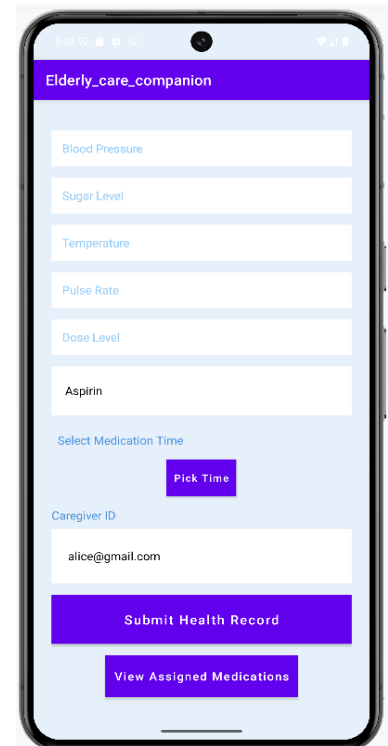


Fig 3: Senior Citizen Activity

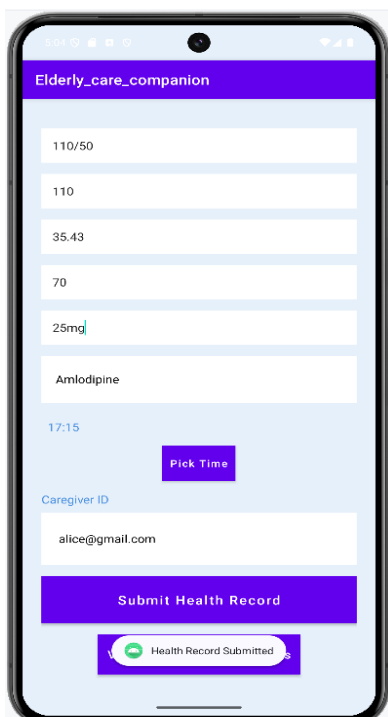


Fig 4: The senior citizen is providing their health record to the caregiver.

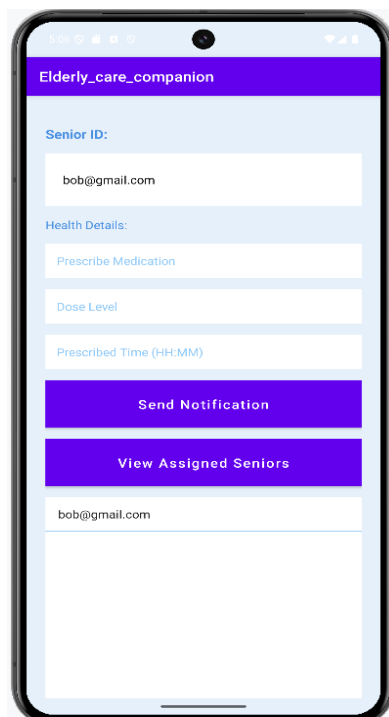


Fig 5 : After submitting the health details, the caregiver will log in and review the assigned seniors and their health records.

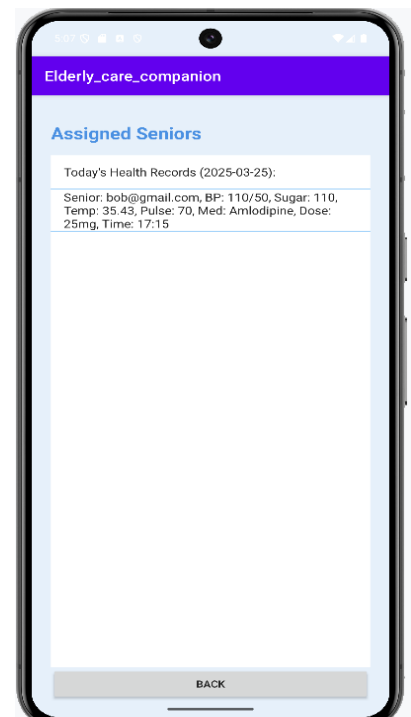


Fig 6: Based on the health details given by senior citizen ,care giver will prescribe medication and send notifications

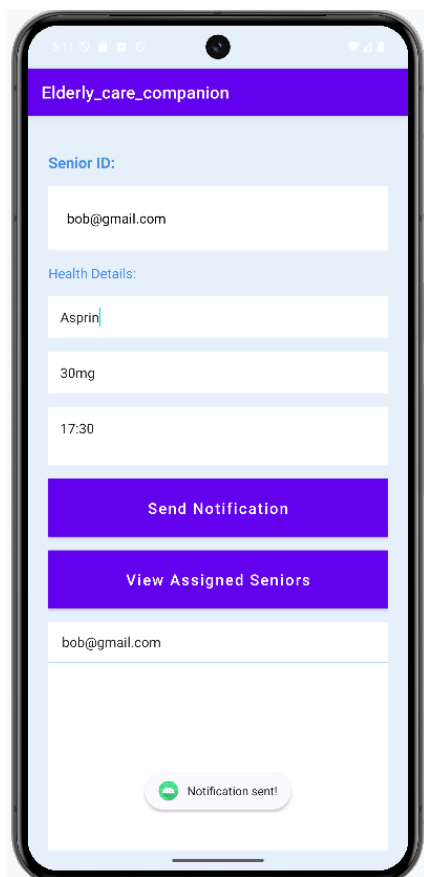


Fig 7 : Sending notifications to the Senior citizen

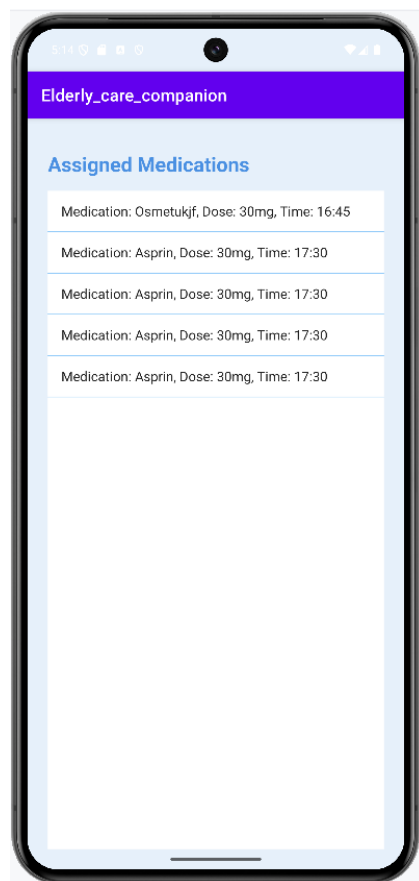


Fig 8: Senior citizen viewing the medications prescribed by care giver

8. CONCLUSION

The **Elderly Care Companion** mobile application successfully delivers a robust, user-centric solution to enhance elderly care by integrating health monitoring, medication management, and caregiver-senior connectivity into a single platform. Through its secure authentication, intuitive interface with large text and high-contrast design, and SQLite-backed local data management, the app empowers seniors to actively manage their health while enabling caregivers to provide personalized support efficiently. This project demonstrates the potential of technology to address critical challenges like medication adherence and social isolation, significantly improving seniors' quality of life and easing caregiver burdens. While currently a standalone Android solution, its modular design lays a strong foundation for future enhancements—such as real-time notifications or wearable integration—paving the way for broader impact in elderly healthcare.

9. REFERENCES(web site URLs)

1 Android Developer Documentation

- **URL:** developer.android.com
- **Description:** Guides Android app development, including UI design and activity management for our login and dashboards.

2 SQLite Official Website

- **URL:** www.sqlite.org
- **Description:** Supports local database creation for storing health records and medication data in our app.

3 National Institute on Aging (NIA) - Services for Older Adults Living at Home

- **URL:** www.nia.nih.gov/health/caregiving/services-older-adults-living-home
- **Description:** Provides insights into elderly care needs, informing our health monitoring and caregiver features.

4 AARP - Caregiving Resources

- **URL:** www.aarp.org/caregiving/
- **Description:** Offers caregiving strategies, inspiring our medication management and senior-friendly design.