

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
“JNANA SANGAMA”, BELAGAVI - 590 018



A PROJECT REPORT

on

“Children Vaccination Management Application”

Submitted by

Thejas Rao

4SF20CI062

Shrushanth Kumar

4SF20CI059

In partial fulfillment of the requirements for VI Sem. B. E. CSE(AI&ML)

MOBILE APPLICATION DEVELOPMENT MINI PROJECT

18AIMP68

Under the Guidance of

Mr. Manjunath E.C

Assistant Professor, Department of CSE(AI&ML)

at



SAHYADRI

COLLEGE OF ENGINEERING & MANAGEMENT

An Autonomous Institution

Adyar, Mangaluru - 575 007

2022 - 23



SAHYADRI
COLLEGE OF ENGINEERING & MANAGEMENT
An Autonomous Institution
MANGALURU

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
(ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

CERTIFICATE

This is to certify that the Mobile Application Development Mini project-18AIMP68 work entitled “**Children Vaccination Management Application**” has been carried out by **Thejas Rao (4SF20CI062)**, and **Shrushanth Kumar (4SF20CI059)**, the bonafide students of Sahyadri College of Engineering & Management in partial fulfillment for the award of Bachelor of Engineering in Computer Science & Engineering (Artificial Intelligence & Machine Learning) of Visvesvaraya Technological University, Belagavi during the year 2022-23. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The MAD project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Guide

Mr. Manjunath E.C

HOD

Dr. Pushpalatha K

External Viva:

Examiner's Name

Signature with Date

1.

.....

2.

.....

SAHYADRI
COLLEGE OF ENGINEERING & MANAGEMENT
An Autonomous Institution

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
(ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)



DECLARATION

We hereby declare that the entire work embodied in this Mobile Application Development-18AIMP68 Project Report titled “**Children Vaccination Management Application**” has been carried out by us at Sahyadri College of Engineering and Management, Mangaluru under the supervision of **Mr. Manjunath E.C**, for the award of **Bachelor of Engineering in Computer Science & Engineering(Artificial Intelligence & Machine Learning)**. This report has not been submitted to this or any other University for the award of any other degree.

Thejas Rao (4SF20CI062)

Shrushanth Kumar (4SF20CI059)

Dept. of CSE (AI & ML), SCEM, Mangaluru

Abstract

Vaccinations are frequently overlooked or delayed because parents and carers struggle to keep track of their child's vaccine schedule. This poses a considerable risk to children's health and well-being since they may be vulnerable to vaccine-preventable infections. To address this issue, we created the Children Vaccination Reminder Application, which is an innovative solution. Getting children vaccinated on time is critical to protecting their health and limiting the spread of dangerous diseases. The Children Vaccination Reminder App is intended to help parents and carers keep track of their child's immunization regimen. Based on the child's age, the application sends personalized reminders. It connects to health databases to provide accurate information and simple access to immunization details. The user-friendly interface is designed to appeal to a wide range of users, even those with less technical skills. By streamlining the vaccination process, the system aims to improve vaccination rates, improve public health outcomes, and contribute to children's well-being.

Acknowledgement

It is with great satisfaction and euphoria that we are submitting the Project Report on “**Children Vaccination Management Application**”. We have completed it as a part of the curriculum of Visvesvaraya Technological University, Belagavi for the award of Bachelor of Engineering in Computer Science & Engineering.

We are profoundly indebted to our guide, **Mr. Manjunath E.C**, Assistant Professor, Department of Computer Science & Engineering (Artificial Intelligence & Machine Learning) for innumerable acts of timely advice, encouragement and We sincerely express our gratitude.

We express our sincere gratitude to **Dr. Pushpalatha K**, Head & Professor, Department of Computer Science & Engineering(Artificial Intelligence & Machine Learning) for his invaluable support and guidance.

We sincerely thank **Dr. Rajesh S**, Principal, Sahyadri College of Engineering & Management, who have always been a great source of inspiration.

We extend our sincere regards and respect to **Dr. Manjunath Bhndary**, Chairman, SCEM, having provided all the facilities that helped us in the timely completion of this project report.

Finally, yet importantly, We express our heartfelt thanks to our family & friends for their wishes and encouragement throughout the work.

Thejas Rao (4SF20CI062)

Shrushanth Kumar (4SF20CI059)

Table of Contents

Abstract	i
Acknowledgement	ii
Table of Contents	iv
List of Figures	v
1 Introduction	1
1.1 Overview	2
1.2 Purpose	2
1.3 Scope	2
2 Requirement Specification	3
2.1 Hardware Requirements	3
2.2 Software Specification	3
3 System Design	4
3.1 System Architecture	4
3.2 Application Module	5
3.3 End Users	5
3.4 Limitations	5
4 Implementation	7
4.1 Overview	7
4.2 Languages Used	7
4.2.1 Java	7
4.2.2 XML	8
4.3 Android Studio	8
4.4 Firebase	8
4.5 Pseudocode	9

4.5.1	Pseudocode for Registering a user	9
4.5.2	Pseudocode for Schedule Page	10
4.5.3	Pseudocode for View Child Page	11
4.5.4	Pseudocode for Reminder Page	13
4.5.5	Pseudocode for Child Addition Page	14
5	Results and Discussion	16
5.1	Register Activity	16
5.2	Login Activity	17
5.3	View Child Activity	17
5.4	Add child Activity	18
5.5	Schedule Activity	19
5.6	Home Activity	20
5.7	Remainder Activity	21
6	Conclusion and Future work	23
7	References	24

List of Figures

3.1	Architecture Diagram of Children Vaccination Management	4
5.1	Register Activity	16
5.2	Login Activity	17
5.3	View Child Activity	18
5.4	Add child Activity	19
5.5	Schedule Activity	20
5.6	Home Activity	21
5.7	Remainder Activity	22

Chapter 1

Introduction

Vaccinations serve an important role in protecting children's health and well-being by delivering protection against a wide range of infectious diseases. However, due to the intricacy of vaccination schedules and the fast-paced nature of modern life, it can be difficult for parents and carers to keep up with their child's immunization requirements. Vaccinations that are missed or delayed can have major repercussions, leaving children vulnerable to preventable diseases and contributing to public health hazards.

To solve this issue, the development of a vaccination reminder system for children is becoming increasingly crucial. This approach makes use of technological improvements to send parents and carers timely and personalized reminders, ensuring that children receive their immunisations on time. The vaccine reminder system promises to make tracking and managing immunization schedules easier.

Vaccinations play a crucial role in safeguarding the health and well-being of children by providing immunity against a wide range of infectious diseases. However, with the complexity of vaccination schedules and the fast-paced nature of modern life, it can be challenging for parents and caregivers to stay on top of their child's immunization requirements. Missed or delayed vaccinations can have serious consequences, leaving children vulnerable to preventable diseases and contributing to public health risks.

To address this issue, the development of a children vaccination reminder system becomes increasingly important. This system utilizes technological advancements to provide timely and personalized reminders to parents and caregivers, ensuring that children receive their vaccinations on schedule. The vaccination reminder system aims to simplify the process of tracking and managing immunization schedules

1.1 Overview

The children vaccine reminder application is a comprehensive solution designed to help parents and carers properly manage their child's immunization schedule. This strategy makes use of technology, such as mobile applications and automated notifications, to deliver personalized reminders and make vaccination information easily accessible. Because of the user-friendly interface, it is accessible to a wide range of users, providing convenience and ease of use. The children vaccine reminder system aims to enhance vaccination rates, protect children from preventable diseases, and contribute to overall public health by simplifying the process of tracking and managing vaccinations.

1.2 Purpose

The goal of the children vaccination reminder application is to prioritize and protect children's health by ensuring they obtain necessary vaccines on time, thereby contributing to individual and community protection against vaccine-preventable diseases. Vaccinations are essential for preventing children from preventable diseases and improving their general health and well-being. However, keeping track of the prescribed immunization schedule, which frequently involves numerous vaccines delivered at different intervals, can be difficult for parents or carers. A vaccination reminder system contributes to addressing this issue by delivering timely notices or reminders to parents or carers about upcoming or overdue vaccinations for their children. As a result, it is a beneficial tool for improving vaccine adherence and preventing the potential hazards connected with vaccination.

1.3 Scope

The children vaccine reminder application covers a variety of topics linked to ensuring children's timely and comprehensive immunization. The system has a broad range of applications. A variety of vaccinations are advised for children, including routine childhood immunisations as well as specialized immunisations based on regional or national guidelines. The procedure aspires to be adaptable to varied healthcare settings and accessible to parents and carers from diverse backgrounds. It also takes into account privacy and data security measures to safeguard sensitive health information. Overall, the children vaccine reminder application seeks to provide a comprehensive solution for efficiently managing and promoting children's immunisations.

Chapter 2

Requirement Specification

2.1 Hardware Requirements

- Processor: Intell(R) Core(TM) i5-10400F CPU @4.3GHz
- RAM : 8GB
- Hard Disk : 500GB
- Input Device : Standard keyboard and Mouse
- Output Device : Monitor

2.2 Software Specification

- Programming Language :Java 19.0.2
- Markup Language : eXtensible Markup Language [XML]
- IDE :Android Studio 2022.2.1
- Database: Google Firebase

Chapter 3

System Design

3.1 System Architecture

The architecture diagram of the application is as shown in the below figure 3.1:

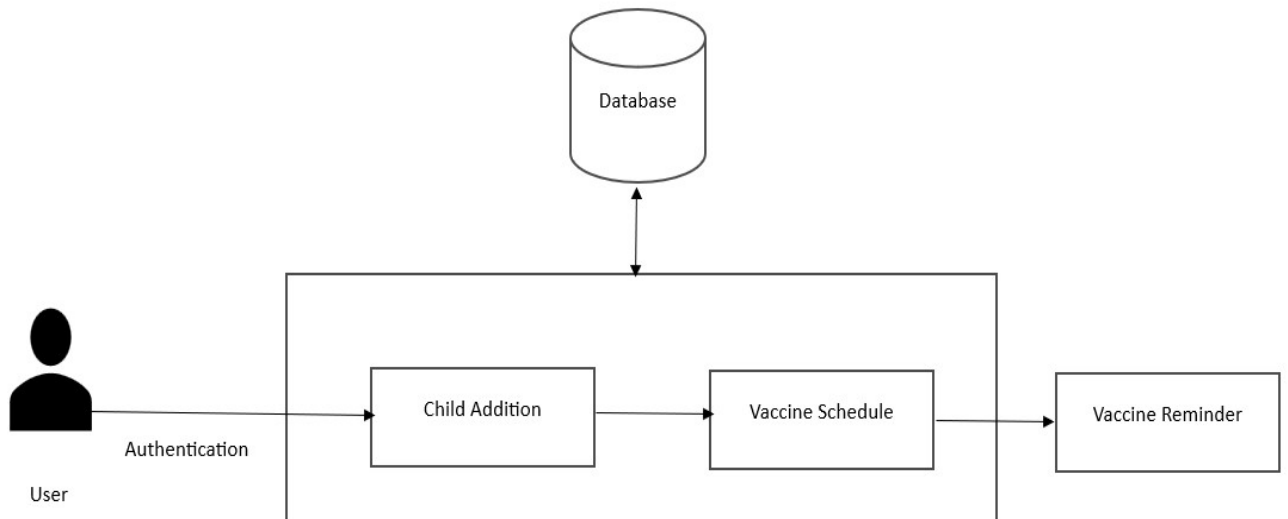


Figure 3.1: Architecture Diagram of Children Vaccination Management

Through the sign-in activity, the Children Vaccination Reminder Application delivers a streamlined sign-in process for clients. Customers can sign in to the application and add their children's information from their smartphones. The application has a user-friendly interface where users may enter basic information about their children and retrieve the government-recommended immunization schedule. They can also keep track of impending vaccinations.

3.2 Application Module

The vaccine reminder application's user module is critical in creating a fluid and user-friendly experience for both parents and healthcare providers. Parents may easily build profiles for their children using this module, providing necessary information such as age and vaccination records. The app sends parents regular reminders about impending vaccines, ensuring that their children's immunisations are always up to date. Furthermore, the user module includes a complete vaccine database with detailed information on the purpose, dose, and probable side effects of each vaccine. This enables parents to make informed vaccination decisions for their children. The vaccine reminder application's user module is an essential component in raising immunization awareness and ensuring children's well-being through timely immunisations.

3.3 End Users

Individuals, parents, and healthcare practitioners who actively participate in the process of maintaining vaccination schedules and promoting immunization are among the end users of the vaccine reminder application. The application is used by parents or guardians to ensure that their children's vaccinations are up to date. They can build profiles for their children, receive timely vaccine reminders, and gain access to extensive vaccine information. As end users, healthcare professionals play an important role in managing their patients' immunization records. End users actively participate in the management and promotion of vaccinations by utilising the vaccination reminder application, contributing to public health and reducing disease spread. The app is a complete platform that enables individuals and healthcare providers to actively participate in immunisation initiatives, ensuring a healthier and safer community.

3.4 Limitations

One disadvantage of a children vaccine reminder app is the lack of location tracking for doctors. Without geolocation features, the application may be unable to present users with a quick and accurate list of nearby doctors or healthcare providers who offer immunization services. This constraint may impair the user's ability to locate and select an appropriate healthcare professional for their child's immunization needs. Furthermore, the programme may fail to account for geographical differences in immunisations across the

country. Vaccination standards, timings, and vaccines prescribed can vary depending on local health policies and disease prevalence. If these regional variances are not taken into consideration, vaccination information may be erroneous or inadequate, potentially leading to missing or needless immunisations for children. Addressing these issues by incorporating geolocation tracking and regional-specific vaccine information will significantly improve the effectiveness and reliability of the children's vaccination reminder app.

Chapter 4

Implementation

4.1 Overview

Google’s Android operating system for mobile devices is extremely popular. It offers a diverse framework for running multiple programmes as well as a variety of features that improve the user experience. Because Android is open-source, it provides for customization and flexibility, allowing developers to create creative and different applications. It offers a large ecosystem of apps and services, making it a popular choice among users and developers alike. Android provides easy interaction with Google services, broad device interoperability, and a massive library of apps available via the Google Play Store. Its user-friendly interface, frequent updates, and active community support all contribute to its widespread adoption and success in the mobile industry.

4.2 Languages Used

4.2.1 Java

Java is the programming language used in Android Studio for application development. It has various advantages for Android app development, including platform compatibility, a huge developer community, and a comprehensive range of libraries and frameworks. Because of Java’s platform independence, developers may write code that runs on numerous systems, including Android. The large number of Java libraries and frameworks available simplifies development by providing pre-built functionality for typical tasks. Furthermore, the large Java developer community provides a multitude of tools, documentation, and support for Android developers, making Java a popular and dependable choice for developing

strong and feature-rich Android applications.

4.2.2 XML

In Android Studio, XML (eXtensible Markup Language) is a markup language used extensively for developing user interfaces and defining the structure and appearance of app layouts. XML offers a versatile and hierarchical syntax for developers to specify the UI elements, layouts, and resource references utilized in an Android app. Developers can use XML to design and organise graphic components, establish event handlers, and specify resource variables such as strings, colors, and dimensions. The separation of UI design and application functionality in XML allows for efficient collaboration between designers and developers, and its readability and ease of maintenance make it an essential aspect of the Android app development process in Android Studio.

4.3 Android Studio

Android Studio is a specialized integrated development environment (IDE) for creating applications for the Android platform. It includes a complete range of tools, libraries, and features that help to speed up the app development process. Developers may use Android Studio to efficiently write, debug, and test code, create user interfaces with a visual editor, access a wide range of libraries and APIs, and simply distribute their applications to Android devices. The IDE's broad ecosystem and thorough documentation make it a popular choice for Android app development, giving developers a stable and efficient environment in which to construct high-quality, feature-rich, and visually appealing Android applications.

4.4 Firebase

Google Firebase is a sophisticated platform that is extensively utilised in Android Studio for application development. It offers a comprehensive set of tools and services that help to simplify and improve many elements of app development. Among other things, Firebase provides real-time database, cloud storage, authentication, cloud messaging, and analytics. These features allow developers to incorporate user authentication, store and retrieve data from a scalable database, deliver push notifications, and gain insights into user behaviour. Firebase also offers strong APIs and SDKs that help to expedite the development

process and facilitate rapid app deployment. Firebase is a popular choice for Android Studio developers due to its ease of use, scalability, and comprehensive feature set, allowing them to focus on building high-quality applications with rich functionality and seamless user experiences.

4.5 Pseudocode

4.5.1 Pseudocode for Registering a user

This activity is responsible for user registration and authentication using Firebase Authentication.

```
mAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            if (task.isSuccessful()) {
                Toast.makeText(register.this, "User is Registered Successfully"
                    , Toast.LENGTH_SHORT).show();
                DocumentReference documentReference = mStore.collection("users")
                    .document(Objects.requireNonNull(mAuth.getCurrentUser()
                        ).getUid());
                Map<String, Object> user = new HashMap<>();
                user.put("fName", fullName);
                user.put("email", email);
                user.put("mobile", mobile);

                documentReference.set(user)
                    .addOnSuccessListener(new OnSuccessListener<Void>() {
                        @Override
                        public void onSuccess(Void unused) {
                            Log.d(TAG, "onSuccess: User Profile created");
                            startActivity(new Intent(getApplicationContext(),
                                NavigationActivity.class));
                        }
                    })
            }
        }
    })
```

```

    })

    .addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Log.d(TAG, "onFailure: " + e.toString());
        }
    });

    progressBar.setVisibility(View.INVISIBLE);
} else {
    Toast.makeText(register.this, "Registration failed"
        + task.getException().getMessage(), Toast.LENGTH_SHORT).show();
    progressBar.setVisibility(View.INVISIBLE);
}
}
});

```

The register activity includes functions to handle user registration in the child vaccine reminder app. It validates user input for email, password, name, and phone number, creates a new user account using Firebase authentication, and stores user information in the Firestore database.

4.5.2 Pseudocode for Schedule Page

This activity displays the schedule of vaccination applicable to the child.

```

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

    sharedPreferences = getSharedPreferences("SelectedItems", MODE_PRIVATE);

    ExpandableListView expandableListView;
    listDataGroup = new ArrayList<String>();
    listDataChild = new HashMap<String, List<String>>>();
}

```

```

selectedItems = new HashMap<String,List<Boolean>>();
expandableListView = findViewById(R.id.expandableVaccineView);
saveButton = findViewById(R.id.saveButton);

String childName = getIntent().getStringExtra("childName");
TextView childNameTextView = findViewById(R.id.childNameTextView);
childNameTextView.setText(childName);

expandableListViewAdapter = new ExpandableListViewAdapter(this,
listDataGroup, listDataChild, selectedItems);
expandableListView.setAdapter(expandableListViewAdapter);
initListData();
loadSelectedItems();
saveButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        saveSelectedItems(childName);
        Toast.makeText(schedule.this, "Selected items saved."
        , Toast.LENGTH_SHORT).show();
        Intent intent = new Intent(schedule.this, view_child.class);
        startActivity(intent);
    }
});
}

```

This activity displays a schedule of vaccines for children using an ExpandableListView. It allows users to select vaccines for each child and saves the selected items to shared preferences, enabling the user to retrieve and modify the selections.

4.5.3 Pseudocode for View Child Page

This activity is displays details of all the children saved by the user.

```

fbb.setOnClickListener(new View.OnClickListener() {
    @Override

```

```
        public void onClick(View v) {
            Intent intent = new Intent(view_child.this, MainActivity.class);
            startActivity(intent);
        }
    });

    EventChangeListener();
}

private void EventChangeListener() {
    String userId = Objects.requireNonNull(mAuth.getCurrentUser()).getUid();
    db.collection("users").document(userId).collection("child")
        .orderBy("cName", Query.Direction.DESENDING) // Order by
        descending to show newly added child first
        .addSnapshotListener(new EventListener<QuerySnapshot>() {
            @Override
            public void onEvent(@Nullable QuerySnapshot value, @Nullable
                FirebaseFirestoreException error) {
                if (error != null) {
                    Log.e("Firestore error", error.getMessage());
                    return;
                }

                for (DocumentChange dc : value.getDocumentChanges()) {
                    if (dc.getType() == DocumentChange.Type.ADDED) {
                        DataClass dataClass =
                            dc.getDocument().toObject(DataClass.class);

                        dataClassArrayList.add(0, dataClass); // Add the
                            new child's data at the top
                    }
                }

                // Notify the adapter that the dataset has changed
            }
        })
    }
```

```
        myAdapter.notifyDataSetChanged();
    }
});
}
```

This activity that displays a list of child data fetched from a Firestore database using a RecyclerView. It utilizes a custom adapter to populate the RecyclerView with the child data, and it listens for real-time changes in the database to dynamically update the displayed list when new child documents are added.

4.5.4 Pseudocode for Reminder Page

This activity gives the details of upcoming vaccine.

```
private static final String CHANNEL_ID = "VaccinationReminderChannel";
private static final int NOTIFICATION_ID = 1;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_notify);

    recyclerView = findViewById(R.id.recyclerView2);
    recyclerView.setHasFixedSize(true);
    recyclerView.setLayoutManager(new LinearLayoutManager(this));

    childList = new ArrayList<>();
    childAdapter = new ChildAdapter(childList);
    recyclerView.setAdapter(childAdapter);

    mAuth = FirebaseAuth.getInstance();
    db = FirebaseFirestore.getInstance();
    sharedPreferences = getSharedPreferences("MySharedPref", MODE_PRIVATE);

    if (mAuth.getCurrentUser() != null) {
```

```
        retrieveChildData();

        // Schedule reminders for each child
        for (Child child : childList) {
            scheduleReminder(child);
        }
    } else {
        promptLogin();
    }

    fbb=findViewById(R.id.fbb);
```

The activity manages the vaccination reminders for children in a Child Vaccine Reminder application. It retrieves child data from a Firestore database, calculates their age and remaining days until the next vaccination, and schedules reminders using alarms and notifications to prompt the user about upcoming vaccinations.

4.5.5 Pseudocode for Child Addition Page

This activity is responsible for adding details of the child to be saved.

```
String[] strDOB = dob.split("/");
LocalDate now = LocalDate.now();
LocalDate birthDate = LocalDate.of(Integer.parseInt(strDOB[2])
, Integer.parseInt(strDOB[1]), Integer.parseInt(strDOB[0]));
days = ChronoUnit.DAYS.between(birthDate, now);

sharedPreferences.edit().putInt("age", (int) days).apply();

Intent serviceIntent = new Intent(this, AgeUpdateService.class);
startService(serviceIntent);

DocumentReference documentReference = mStore.collection
("users").document(Objects.requireNonNull
```

```

        (mAuth.getCurrentUser()).getUid()).collection("child").document(name);
    Map<String, Object> child = new HashMap<>();
    child.put("cName", name);
    child.put("dob", dob);
    child.put("bloodGrp", bloodGrp);
    child.put("height", height);
    child.put("weight", weight);
    child.put("gender", gender);

    documentReference.set(child).addOnSuccessListener
    (new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            Toast.makeText(getApplicationContext(),"Child added
            Successful",Toast.LENGTH_SHORT).show();
            startActivity(new Intent(getApplicationContext(),
            MainActivity.class));
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Log.d(TAG, "onFailure: " + e.toString());
        }
    });
}

}
}

```

Add child section in the home page handles adding a child's information in a child vaccine reminder app. It includes functionality to set up notification manager, retrieve user input for child's name, date of birth, blood group, height, and weight. It performs input validation, handles date picker dialog, calculates the age of the child, and stores child's information in Firestore.

Chapter 5

Results and Discussion

5.1 Register Activity

Users without an existing account are granted the ability to create a new account.

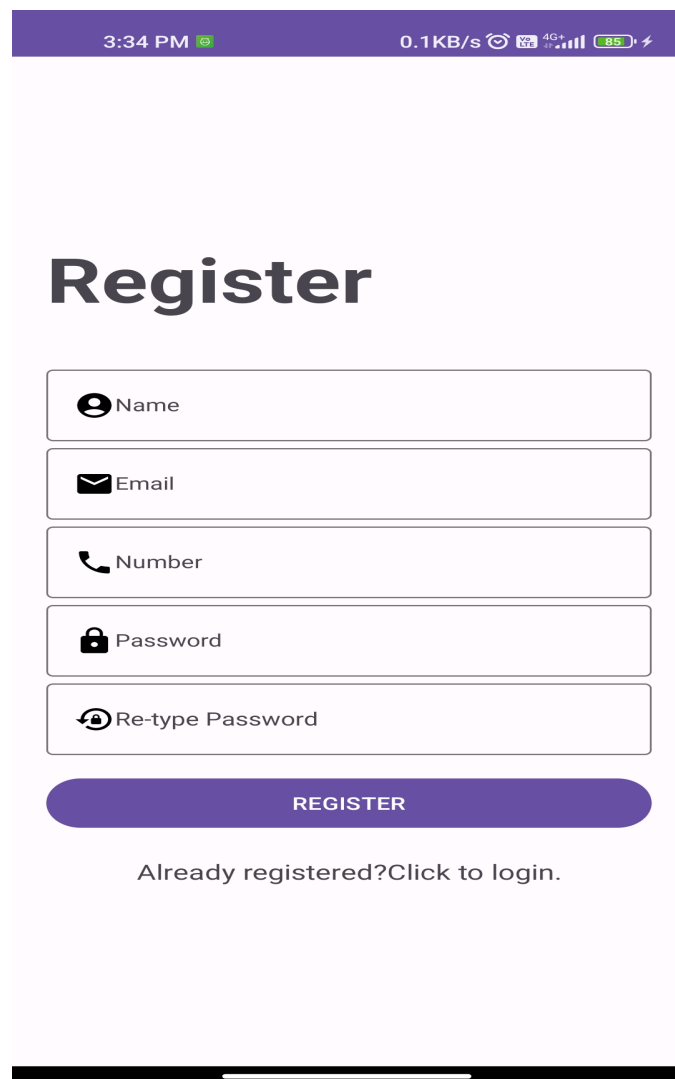
A mobile application interface for a registration screen. At the top is a purple status bar with white text showing '3:34 PM', a green location icon, '0.1KB/s', a Wi-Fi icon, '4G+', cellular signal bars, and a battery icon with '95'. The main background is light purple. The word 'Register' is centered in a large, bold, dark grey font. Below it are five white input fields with rounded corners and thin grey borders. Each field has a small icon on the left: a person icon for 'Name', an envelope icon for 'Email', a telephone handset icon for 'Number', a padlock icon for 'Password', and a circular arrow icon for 'Re-type Password'. Below the input fields is a purple rounded rectangular button with the word 'REGISTER' in white capital letters. Under the button, the text 'Already registered?Click to login.' is displayed in a smaller, dark grey font. At the very bottom is a black horizontal bar with a white line in the center, representing the home indicator on an iPhone.

Figure 5.1: Register Activity

In this page, user must enter their credentials in order to register to the application. If the user is already registered he can then move on to login.

5.2 Login Activity

The login page allows users to securely log in using their email and password. By utilizing

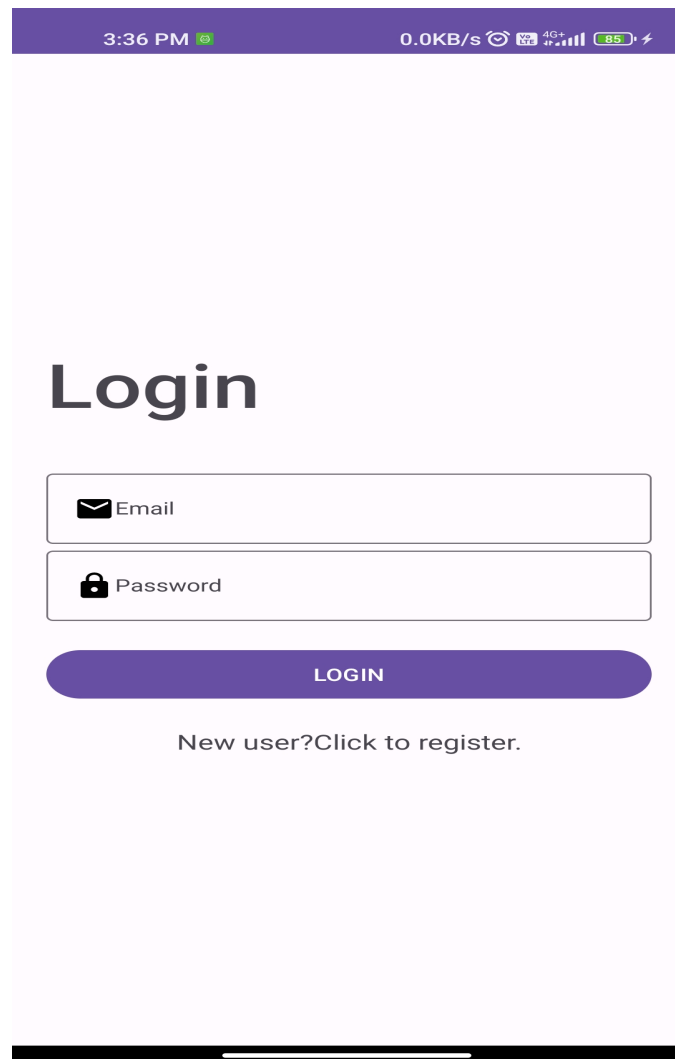


Figure 5.2: Login Activity

this platform, the users can sign in, leading them to the Home Page. In the Home Page, they will have the option to add the child details and get the vaccination schedules.

5.3 View Child Activity

In this page the user can view the details of their children.

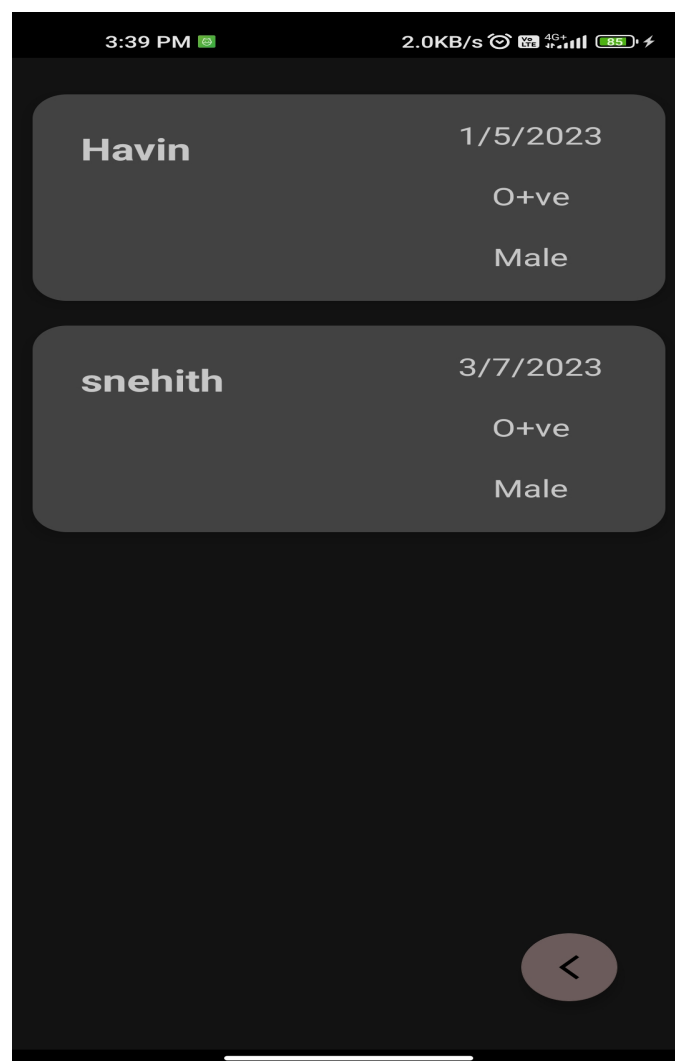


Figure 5.3: View Child Activity

The page displays the details of the children. Upon clicking any child's name, the schedule of the child's vaccine is displayed.

5.4 Add child Activity

In this page the user can add the details of the children.

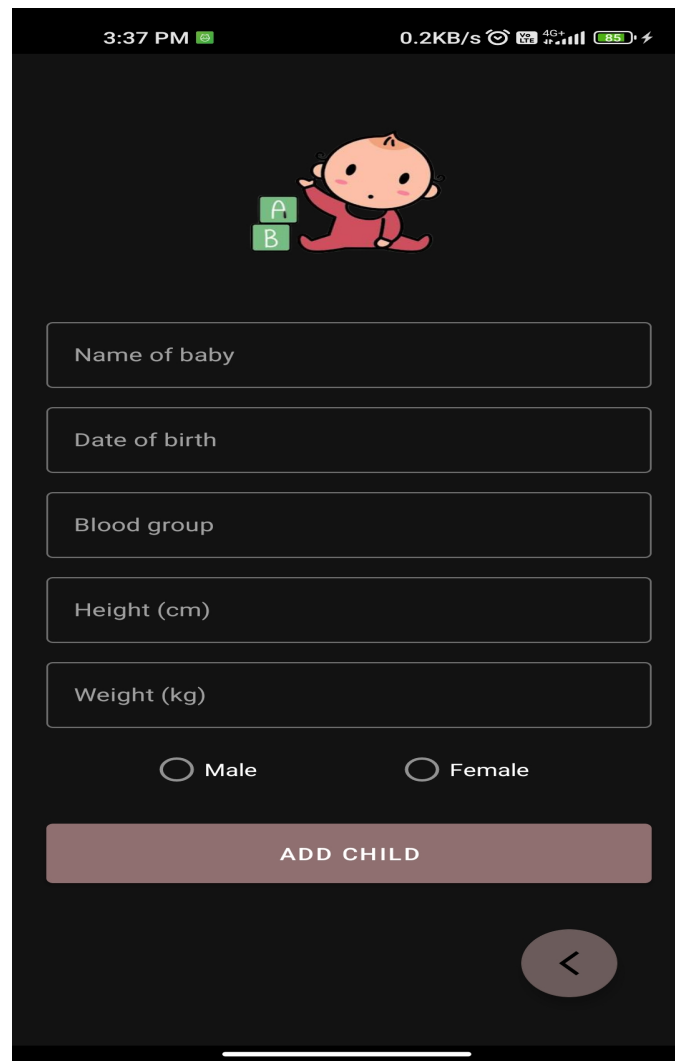


Figure 5.4: Add child Activity

In this activity the user can enter the basic detail of their child that includes name, date of birth, gender, blood group, height and weight. Once the details are added the information is stored in the cloud firestore and accessed whenever needed.

5.5 Schedule Activity

This page provides the schedule of vaccinations as recommended by the government.

3:41 PM 0.4KB/s 4G+ 85%

Havin

At Birth

Bacillus Calmette–Guérin (BCG)	<input checked="" type="checkbox"/>
Oral polio vaccine (OPV 0)	<input checked="" type="checkbox"/>
Hepatitis B (Hep – B1)	<input checked="" type="checkbox"/>

6 Weeks

10 Weeks

14 Weeks

6 Months

9 Months

9-12 Months

12 Months

SAVE

Figure 5.5: Schedule Activity

The user can mark the vaccinations that have been given to the child in the list. This way the user can keep track of the vaccination journey of the child.

5.6 Home Activity

The page gives options for the user to perform the various operations in application.

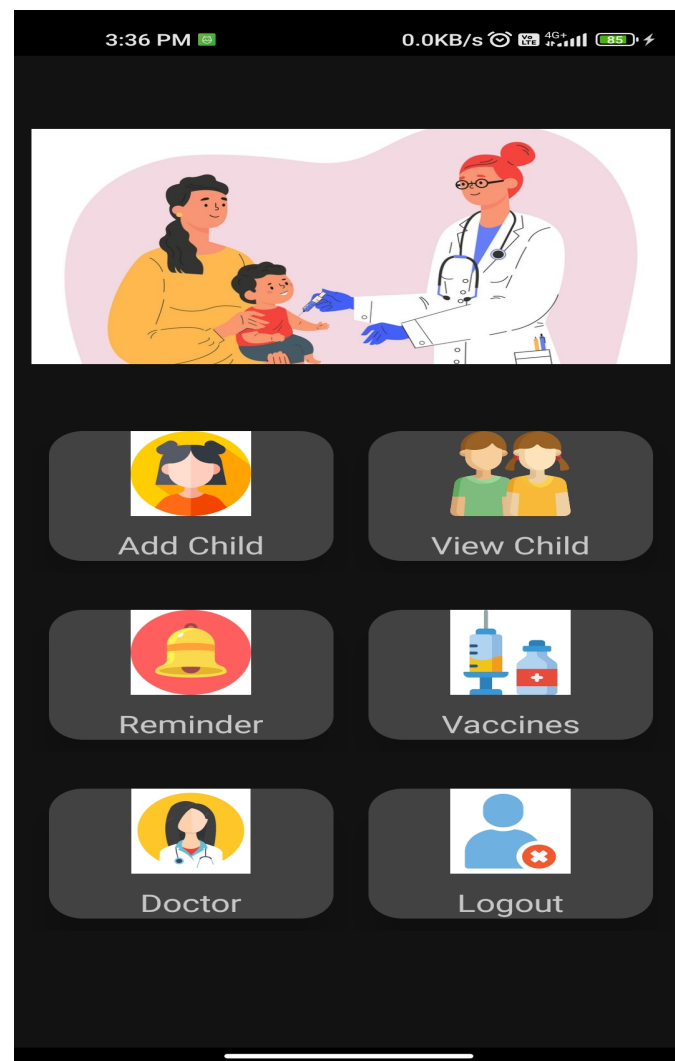


Figure 5.6: Home Activity

The page provides options to add the child, view the child details, reminder for the next earliest vaccinations. It also provides vaccination details and details of doctors present in a location.

5.7 Remainder Activity

This page provides the remainder for next vaccine to be given to the child.



Figure 5.7: Remainder Activity

This page provides the remainder for next vaccine to be given to the child. It consists of the name, their age, days remaining upto the vaccination and also the vaccination list.

Chapter 6

Conclusion and Future work

The creation of a children vaccination reminder application is critical in promoting immunization and protecting children's health. The programme, with its user-friendly appearance and straightforward features, is a helpful tool for parents and carers to stay updated about forthcoming vaccination schedules and receive timely reminders. The programme enables parents to prioritize their child's immunization and guarantee timely vaccine administration by offering personalized notifications, educational resources, and easy access to vaccination records. The children vaccine reminder application helps to increase vaccination rates, reduce the risk of vaccine-preventable diseases, and ultimately improve children's overall well-being. With the potential to improve vaccine compliance and public health, the application is an important component of modern healthcare and well-being.

The children vaccination system can be improved by using geolocation technology to track and find doctors within a certain zone, allowing users and healthcare providers to communicate seamlessly. Furthermore, the system can be tailored to meet vaccine-specific demands in certain regions by providing accurate and timely information. Current information on vaccine needs, illness prevalence, and local immunization schedules. These innovations would increase accessibility, convenience, and personalized care for children, ultimately leading to increased immunization rates and improved health outcomes.

Chapter 7

References

- [1] Google Developer Training Team, "Android Developer Fundamentals Course-Concept Reference," 2017. [Online]. Available: <https://www.gitbook.com/book/googledeveloper-training/android-developerfundamentals-course-concepts/details>.
- [2] E. Hellman, "Android Programming - Pushing the limits," 1st Edition, Wiley India Pvt Ltd, 2014. ISBN-13: 978-8126547197.
- [3] D. Griffiths and D. Griffiths, "Head First Android Development," 1st SPD Publishers, 2015. ISBN-13: 978-9352131341.
- [4] B. Phillips, C. Stewart, and K. Marsicano, "Android Programming: The Big Nerd Ranch Guide," 3rd Edition, Big Nerd Ranch Guides, 2017. ISBN-13: 978-0134706054.