

**IN
PARTNERSHIP
WITH
PLYMOUTH
UNIVERSITY**

Name: Sanjana Witharanage

Student Reference Number: 10707426

Module Code: CNET343SL	Module Name: Distributed Systems
Coursework Title: Final Report Submission	
Deadline Date: 09/05/2022	Member of staff responsible for coursework: Pramudya Thilakaratne
Programme: BSc. Software Engineering	
Please note that University Academic Regulations are available under Rules and Regulations on the University website www.plymouth.ac.uk/studenthandbook .	
Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for component parts.	
10707426 Sanjana Witharanage 10707370 Krishan Shanuka 10707273 Manuja Mallikarachchi 10707360 Kavindya Sandeepanie	

All members contributed equally

We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.

Signed on behalf of the group:

Sanjana Witharanage

Individual assignment: ***I confirm that I have read and understood the Plymouth University regulations relating to Assessment Offences and that I am aware of the possible penalties for any breach of these regulations. I confirm that this is my own independent work.***

Signed:

Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offence.

I have not used translation software.

If used, please state name of software.....

Overall mark _____ % Assessors Initials _____ Date _____

CONTENTS

CONTENTS	1
LIST OF FIGURES.....	3
LIST OF TABLES.....	5
1 INTRODUCTION.....	6
2 BACKGROUND	6
3 PROJECT SCOPE.....	6
4 PROJECT OBJECTIVES	7
5 PROBLEM STATEMENT.....	8
6 ASSUMPTIONS	8
7 SYSTEM REQUIREMENTS	9
7.1 FUNCTIONAL REQUIREMENTS	9
7.2 NON-FUNCTIONAL REQUIREMENTS	10
8 TECHNOLOGIES USED.....	10
8.1 FRONT-END	10
8.1.1 FLUTTER	10
8.1.2 ANGULAR	11
8.2 BACKEND.....	11
8.2.1 ASP.NET CORE WEB API.....	11
8.3 MIDDLEWARE JUSTIFICATION	12
8.4 3 RD PARTY LIBRARIES JUSTIFICATION	13
8.4.1 API.....	13
8.4.2 WEB APPLICATION.....	13
8.4.3 MOBILE APPLICATION	14
8.5 3 RD PARTY API USED	14
8.6 DEPLOYMENT	15
9 SYSTEM DESIGN.....	17
9.1 PROJECT IMPLEMENTATION DIAGRAM	17
9.2 ACTIVITY DIAGRAMS.....	18
9.3 USE-CASE DIAGRAM	21
10 APPLICATION IMPLEMENTATION.....	22
10.1 VMS MOBILE APPLICATION	22

10.1.1	LOGIN	22
10.1.2	REGISTRATION	23
10.1.3	HOME	24
10.1.4	APPOINTMENT	25
10.1.5	SERVICES	29
10.1.6	INFORMATION	30
10.1.7	PROFILE	33
10.2	VMS WEB APPLICATION	35
10.2.1	GENERAL	35
10.2.2	ADMIN	38
10.2.3	STAFF	65
11	CHALLENGERS/ISSUES AND RESOLVING	70
12	FUTURE IMPLEMENTATIONS	72
13	CONCLUSION	72
14	GROUP CONTRIBUTION	72
	REFERENCES	73
	APPENDIX	74
14.1	PROJECT PROPOAL	74

List of Figures

Figure 8.1: Middleware flow	12
Figure 8.2: Front-End CI/CD pipeline	15
Figure 8.3: Back-End CI/CD pipeline.....	15
Figure 8.4: Stored procedure scheduled automation with Azure Runbook.....	16
Figure 9.1: Project implementation diagram.....	17
Figure 9.2:User- Activity diagram	18
Figure 9.3: Admin- Activity diagram.....	19
Figure 9.4: Staff- Activity diagram	20
Figure 9.5: Use-Case diagram	21
Figure 10.1: Mobile login screen	22
Figure 10.2: Incorrect credentials error	22
Figure 10.3: Registration-step 1	23
Figure 10.4: Registration-step 2	23
Figure 10.5: Registration-step 3	24
Figure 10.6: Registration-step 4	24
Figure 10.7: Home screen.....	24
Figure 10.8: User appointments	25
Figure 10.9: User appointments	25
Figure 10.10: Add an appointment	26
Figure 10.11: Select a vaccination date	26
Figure 10.12: Select user district.....	26
Figure 10.13: Display vaccination centres for the selected district	27
Figure 10.14: Select a vaccination centre	27
Figure 10.15: Success message after the appointment booking	27
Figure 10.16: Email received.....	27
Figure 10.17: Confirm cancelling an appointment	28
Figure 10.18: Email for the cancelled appointment	28
Figure 10.19: Services	29
Figure 10.20: Report symptoms	29
Figure 10.21: Report symptoms - choose answers	29
Figure 10.22: QR code scanner	30
Figure 10.23: Information	30
Figure 10.24: Vaccination information	31
Figure 10.25: COVID-19 situation report.....	31
Figure 10.26: Pre-vaccine self-checklist.....	32
Figure 10.27: HPB contact numbers	32
Figure 10.28: Profile	33
Figure 10.29: Profile- Change Password.....	33
Figure 10.30: Validation when trying to update details without entering any data	33
Figure 10.31: Successfully updated account details	34
Figure 10.32: Successfully updated the password	34
Figure 10.33: Email received for password change	34

Figure 10.34: Web application login page	35
Figure 10.35: Forgot password	36
Figure 10.36: Email to reset the password	36
Figure 10.37: Reset password	37
Figure 10.38: Password reset success alert.....	37
Figure 10.39: Admin dashboard	38
Figure 10.40: Admin dashboard	39
Figure 10.41: Staff details	40
Figure 10.42: Create a staff account	41
Figure 10.43: Email received by the staff member to set up the password	41
Figure 10.44: Set up a new password	42
Figure 10.45: Staff details	42
Figure 10.46: View staff member's details.....	43
Figure 10.47: Edit staff member's details	43
Figure 10.48: Confirm before sending the reset password link	44
Figure 10.49: Alter for the email to reset the password is sent successfully	45
Figure 10.50: Confirm before deleting a staff account.....	45
Figure 10.51: Citizen details	46
Figure 10.52: Citizen details	47
Figure 10.53: View citizen's details	47
Figure 10.54: Edit citizen's details	48
Figure 10.55: Confirm before the reset password link is sent.....	49
Figure 10.56: Success alert to reset email sent	50
Figure 10.57: Confirm before deleting a citizen account	50
Figure 10.58: Vaccination centre details	51
Figure10.59: Add a new vaccination Centre.....	51
Figure 10.60: View details of the selected vaccination centre	52
Figure 10.61: Edit vaccination centre details.....	53
Figure10.62: Add a new staff member	54
Figure 10.63: Confirm before sending the reset password link	54
Figure 10.64: Admin details.....	55
Figure 10.65: Create a new admin	56
Figure 10.66: Admin details.....	56
Figure 10.67: View admin details	57
Figure 10.68: Edit admin details	58
Figure 10.69: Confirm before sending the reset password link	59
Figure 10.70: Alert for successfully sending reset password link	59
Figure 10.71: Confirm before deleting an admin account.....	60
Figure 10.72: Hospital emails	60
Figure 10.73: Hospital emails	61
Figure 10.74: View email address	61
Figure 10.75: Edit email address.....	62
Figure 10.76: Information logs.....	62

Figure 10.77: Warning logs	63
Figure 10.78: Error logs.....	63
Figure 10.79: Change password	64
Figure 10.80: Appointments	65
Figure 10.81: View appointments.....	66
Figure 10.82: Marking an appointment is complete.....	66
Figure 10.83: Alert after making a completed appointment	67
Figure 10.84: Centre settings	67
Figure 10.85: Edit vaccination centre details.....	68
Figure10.86: Add a new staff member	68
Figure 10.87: Confirm before removing a staff member	69
Figure 11.1: GIT flow.....	70
Figure 11.2: Pipeline releases.....	71
Figure 11.3: Discord notifications	71

List of Tables

Table 1: Contribution Table	72
-----------------------------------	----

Word count excluding Reference, Table of Contents, List of Figures and Tables: 5200

1 Introduction

The proposed system is a mobile application to manage COVID-19 vaccination in Sri Lanka. The application is built to improve the livelihood of Sri Lankan citizens. The details are accessible in a QR form, so places where vaccination certificates are required become a hassle-free experience. The application is built to provide each citizen with their vaccination details which contain information such as the location of where the citizen has taken the vaccine and information about the given vaccine. The app allows the citizen to search for the nearest vaccination centre and book an appointment and provides a way to manage the appointments as well. The app is fully equipped to provide the citizens with various information related to the COVID-19 pandemic to keep citizens up to date. The mobile application is managed by a web application where the admin and medical staff have various levels of authorization for their respective roles. The entire system is made as a distributed system to prevent system-wide downtime and to handle enormous amounts of traffic.

2 Background

Counteracting the COVID-19 pandemic and going back to normal lives is possible when a country reaches 70% immunization against COVID-19 (WHO, 2021). This made countries unite to create a vaccine against the virus. When the vaccine was tested and was safe for public consumption, every country started to vaccinate their citizens. First and Second world countries had the proper technology and resources to provide their citizens with modern vaccination infrastructure. Unfortunately, 3rd world countries like Sri Lanka, to this day, manages the vaccination data given to the citizen using paper. This raises issues in different sectors, such as the integrity of the data. The proof of vaccination given by the government using paper makes it easy to forge the document. Also, paper can be lost or damaged because citizens must carry it with them to gain access to some public and private places; therefore, it is not suitable for long-time use. To rectify these issues and to effectively manage the data given to the citizens, the idea of a vaccination management system was brought up.

3 Project Scope

- The vaccination management system has a web application and a mobile application.
- Citizens can register to the system after verifying the National Identity Card number with the Database.
- Citizens can only use the mobile application, and admins and staff can only use the web application.
- Registered users can book vaccination appointments at the nearest vaccination centre and manage vaccination appointments.

- Registered users can submit symptom reports.
- After vaccinating, the staff members can update the vaccination status of the citizen.
- Adding vaccination centres, assigning staff members to the vaccination centres, creating accounts, and viewing details of citizens, staff members, and admins are the functions related to admin.
- The owner of the account can reset admin created account.
- Sending emails to the user after creating an account, resetting the password, changing the password, forgetting the password, making an appointment, and finishing an appointment.
- Citizens can share their vaccination information through a QR code.

4 Project Objectives

Vaccine Management System (VMS) is mainly a mobile application for citizens and a website for the vaccination teams. VMS promises to deliver authentic proof that a person is vaccinated directly from the vaccination database through a QR code. The app is designed to show all the vaccination details of the user, including the place a person got the jab, vaccine type, batch, etc. These details are derived directly from the 'government vaccination' database.

The application promises to help the Sri Lankan citizen's by

- Eliminating the need to carry a physical document by introducing a digital vaccination certificate.
- Providing an authentic document with a QR code directly from the 'government vaccination' database.
- Reducing the hassle caused at the airport and improving travellers' quality of life abroad.
- Providing accurate information on covid vaccination centres.
- Providing a more trustable source to verify citizens' vaccination status.
- Providing easy access to the certificate through the mobile app.

Additionally, the app allows the user to book an appointment at the nearest vaccination centre effortlessly. When the QR code is scanned by our inbuilt or any QR scanner, a link to the VMS smart vaccination card is presented, which includes complete user details along with the full vaccination details. When it comes to the staff and admins, they are provided with a web application that displays a dashboard based on their authorization level. There are mainly three different types of users, staff, admin, and citizens. Each has different kinds of interaction with the application.

5 Problem Statement

COVID-19 has become something the world is accustomed to, and vaccines against that virus have also become a part of us. But in most countries, there's no proper management or better way to track or prove a way that we're vaccinated. For instance, let's take Sri Lanka. In Sri Lanka, only a single document is given to confirm whether a person is vaccinated. The standalone document is not adequate for international travel. Most countries require a presentable vaccination certificate that includes a readable QR code published by the Sri Lankan government. This arises certain complications for the citizens of Sri Lanka, as the people have to manually apply for the "Smart Certificate" offered by the government, which contains a readable QR that would take up to 21 days, and the "Smart Certificate" is issued only for citizens leaving Sri Lanka. To obtain the competent certificate, one has to upload personal information, vaccination certificate, and documents proving the departure from Sri Lanka.

This won't be fair because only a minority of the population would have a certificate approved for international travel or any purpose. Having an app that can automatically generate a vaccination certificate with an approved QR code reduces some significant issues for the people of Sri Lanka. One of the issues that are addressed is

- Provide authentic vaccination certificate with QR code.
- Reduces the hassle caused at the airport.
- Easy access to the certificate through the mobile app.

And add more, the majority of Sri Lanka people don't have the proper knowledge of where to book an appointment or where the nearest vaccination centre is located. To solve all these issues together and better serve the citizens of Sri Lanka for hassle-free travel and to provide an authentic vaccination certificate with a readable QR code document for everyone, VMS or Vaccine Management System comes to play.

6 Assumptions

To develop the VMS, the creators of this document had to come up with assumptions. Those assumptions are as follows,

- Developers of VMS have granted access to the Sri Lankan government's citizen identity database, which includes all 'National Identity Card' information. To implement that, we have created a mock NIC API to access citizens' data. For further information, refer: <https://sl-citizens.azurewebsites.net/swagger>

7 System Requirements

7.1 Functional Requirements

'Citizen' User:

- After checking the National Identity Card number with the government database, can continue the account setup.
- After confirming the email, the citizen can log in to the system and see the vaccination status.
- Changing passwords and resetting passwords also can do.
- Booking an appointment for vaccination, cancelling the appointment, and adding to the calendar are the functions related to the arrangements.
- The citizen can update the setup details of the profile.

'Staff' User:

- Any staff member can log in to the system by inputting their password and username.
- List appointments are listed and can be sorted by date, username, or email.
- A staff member can update the appointment status as completed by clicking the vaccination icon.
- One staff member can add another staff member to the vaccination centre by entering their email
- Can remove the staff members in the vaccination centre, which their works
- Can update information about vaccination centre, which their works
- Changing passwords and resetting passwords also can do.

'Admin' User:

- Adding vaccination centres, updating, and deleting details of vaccination centres.
- Creating, editing, and deleting accounts for admins and staff members.
- Viewing citizen, staff and vaccination centre details and getting analytics through charts.

7.2 Non-functional requirements

1. Usability
 - The user interfaces are straightforward to use and comprehend.
2. Availability
 - Officers can find job seekers at any time of the day.
3. Maintainability
 - There is a recovery method in place if the system crashes or any other issue happens.
4. Accuracy
 - The system provides immediate feedback.
5. Transparency
 - Users have no visibility into what happens inside the system or backend.
6. Sharing of resources
 - The built resources are divided into services so that they can be shared.
7. Extensibility
 - The services can be tailored to fit users' individual needs.
8. Tolerance to faults
 - There must be no downtime in a distributed system.
9. Interoperability
 - The architecture of the system is service-oriented.

8 Technologies Used

Technologies used in this project,

- For backend: ASP.NET core web API
- For mobile application front-end: Flutter
- For web application front-end: angular

8.1 Front-end

8.1.1 Flutter

Flutter, React Native, Swiftic, Xamarin, Ionic, Corona, JQuery Mobile, and Java are some of the current front-end technologies for mobile applications. Although there are many different technologies, there aren't as many resources to refer to them as there are in React Native, Flutter, and Java. The main reason for this is that, while designing the front-end, the author is likely to encounter a slew of mistakes, defects, and other issues resulting from differences in operating systems, IDEs, and versions. If there are previously established solutions available at that time, it will benefit the author. Flutter, React Native, and Java has a higher presence on YouTube, StackOverflow, and other popular websites than the other technologies mentioned above. The author's project

was chosen since the Flutter community is significant now and will continue to grow in the future. In addition, there is adequate documentation for this. Android Studio, Visual Studio Code, IntelliJ IDEA, and the Emacs IDE are good options for developing Flutter apps.

Flutter is open-source and a user interface software development kit produced by Google. It's used to create cross-platform apps from a single codebase for Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web. (Flutter-Wikipedia, no date)

8.1.2 Angular

The front-end of websites, like mobile apps, is built using a variety of technologies. Vue.js (JavaScript library), Npm, Vue.js (web framework), Ionic 4, Bootstrap, Chrome DevTools, HTML5 Boilerplate, Grunt, Angular, Blockchain Testnet, Grid Guide, Meteor, Git Extensions, Backbone, CodePen, and Foundation. It is straightforward to grasp because Angular is mostly used for web application development and is a well-developed language.

8.2 Backend

8.2.1 ASP.NET core Web API

Backend development for websites and mobile applications is being done in various languages. Some examples are JavaScript, Python, PHP, Java, Golang, C#, and Perl. Any of these languages can be used to create standard backends for developers. However, when linking it to the front-end, the author has two: one for the website and one for the mobile app. It is impossible to utilize the same code for both if the author traditionally writes the backend. The author chose web APIs as a solution, utilizing ASP.NET Core Web API as the backend technology. Asp.net Core web API 6.0 is the newest version. However, the author chose the 5.0 version because the resources accessible on online platforms are limited compared to other languages, and there are fewer courses in the most recent version. Because the author can utilize ASP.NET or ASP.NET core as the backend development technology, there is no need to use another platform like angular, Vue.js, React.js, or Flutter.

8.3 Middleware Justification

We can define Middleware as software that sits in the middle of an operating system and the programs that execute on it. Middleware allows communication and data management for distributed applications by acting as a covert translation layer.

(<https://azure.microsoft.com/en-in/overview/what-is-middleware/>, no date)

Middleware aids in the faster development of apps. It is the glue that holds programs, data, and users together. Middleware can help enterprises with multi-cloud and containerized environments develop and run large-scale applications at a lower cost.

(<https://www.redhat.com/en/topics/middleware/what-is-middleware>, no date)

So, in this vaccination management system API project uses,

1. UseDeveloperExceptionPage Middleware - used to report app runtime errors.
2. UseHttpsRedirection Middleware - used to redirects HTTP requests to HTTPS.
3. UseRouting - used to route requests.
4. UseCors - used to enable cross-origin resources sharing.
5. UseSwagger & UseSwaggerUI - used to generate Swagger API documentation.
(Configured to handle
6. UseAuthentication - used to authenticate users. (Configured to use JSON Web Tokens)
7. UseAuthorization - used to authorize users to access secure resources.
(Configured to use JSON Web Tokens)
8. UseEndpoints - used to map controllers to the request pipeline.

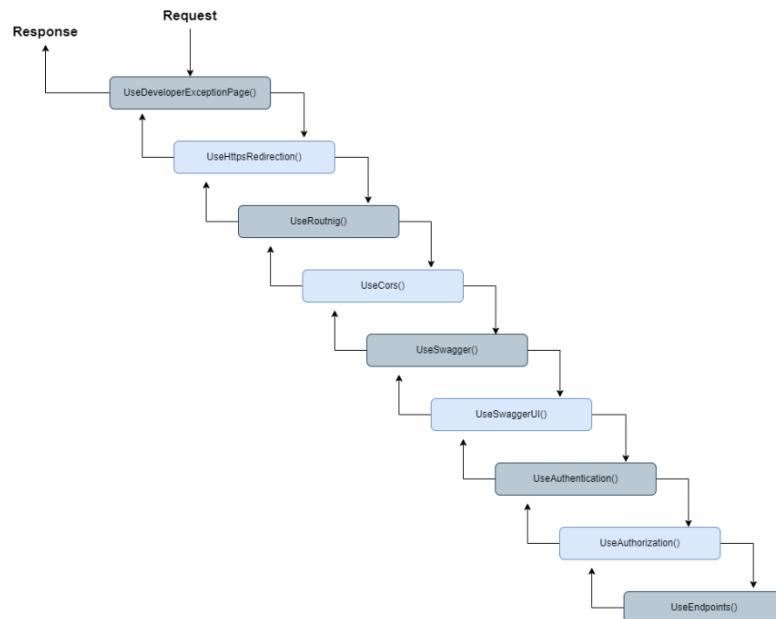


Figure 8.1: Middleware flow

8.4 3rd Party Libraries Justification

8.4.1 API

1. Mailkit

MailKit is an Open-Source, cross-platform .NET mail-client library based on MimeKit and optimized for mobile devices. Using this library system can send required emails to the user.

2. Serilog

Logging frameworks make sending logs to different places easy via simple configurations. Serilog uses sinks to send records to a text file, Database, log management solutions, or potentially dozens of other sites, all without changing code.

3. Swashbuckle.AspNetCore

This library can automatically generate interactive testable API documentation for ASP .NET core endpoints. This library will help developers to work with the front end.

8.4.2 Web Application

1. Data tables

JQuery DataTables is the plugin that powers the Angular DataTables framework. Angular 2+ is supported and set up to support TypeScript.

When using Angular DataTables,

- Developers have a vast dataset coming in from one or more API endpoints
- It would help if developers had customized data sorting/filtering

There are two primary categories of Angular DataTable features: basic and sophisticated. From there, Angular DataTables has a variety of extensions that may be added.

2. Moment

JavaScript developers can use Moment.js to parse, validate, manipulate, and display dates and times.

3. NGX-Avatar

If system have information about the user, system may use this universal avatar component to get or generate an avatar.

4. NGX-Spinner

For Angular 4+ versions, an animated loading spinner informs the user that an operation occurred.

5. NGX-Device-Detector

AOT-compatible device detector powered by Angular 6+ that helps identify the browser, operating system, and other important information about the device accessing the app. User-agent is used to automate the process of analyzing the data.

6. NGX Chart

Charting framework ngx-charts is open source and declarative for angular 2+. Swimlane is responsible for its maintenance. SVG elements are rendered and animated with the help of Angular's binding and speed, and d3's math functions, scales, axes, and form generators are used to make the animations even more impressive.

8.4.3 Mobile Application

1. HTTP

Http package makes it easy to send and process http request made by the mobile application to the server.

2. Shared Preferences

Shared preferences help to store data temporarily in the app which can later be used throughout the application.

3. QR flutter

QR flutter package makes it simple to paint a QR code with the provided the information.

4. Cool Alert

Allows to make custom mobile alerts to notify user.

8.5 3rd Party API Used

1. HPB Realtime API

Used Sri Lanka health promotion bureau API to display Sri Lanka's real-time COVID-19 statistics in the mobile application.

Link - <https://www.hpb.health.gov.lk>

8.6 Deployment

According to figure 9.1, this system contains Mobile Application, Web Application, Rest API, Database and Cloud Storage. Accordingly, most components of the system should be hosted in a cloud.

Developers of this project decided to select Azure related services to host this system because it provides unlimited support to host Microsoft's own ASP. Net APIs on their App Services.

- Web Application deployed to Azure static website container, with the help of GitHub Actions pipeline.

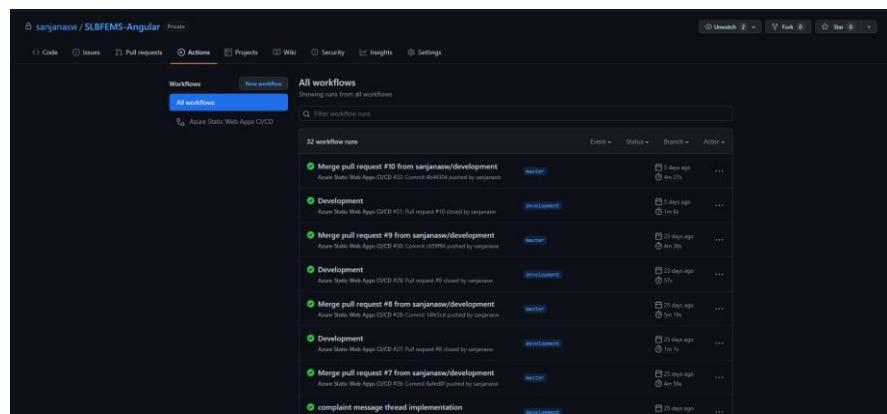


Figure 8.2: Front-End CI/CD pipeline

- Rest API deployed to Azure app service, with the help of GitHub Actions pipeline.

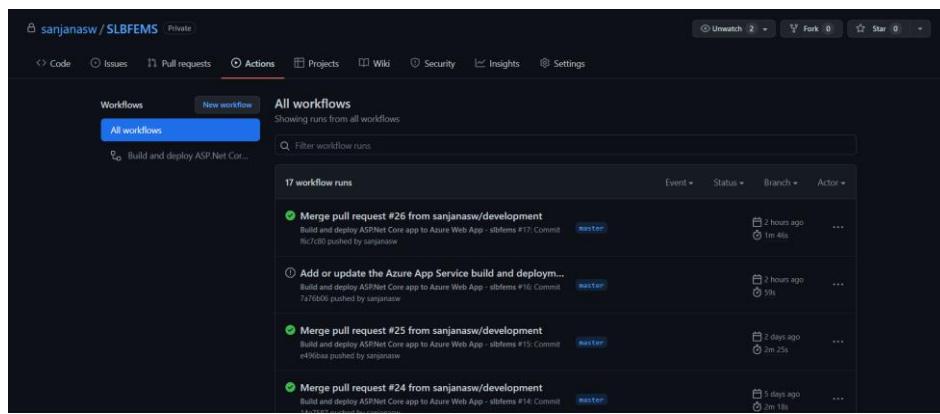


Figure 8.3: Back-End CI/CD pipeline

- This system has used the Azure MsSQL server along with Azure Runbook to store the data and run automated stored procedures. Runbook has been used to

change the status of appointments after those are expired.

The screenshot shows the Azure Runbook interface for the 'AppointmentExpire' runbook. On the left, the 'Overview' section displays basic information: Resource group: vmsauto, Account: vmsauto, Location: Australia Central, and Subscription: Azure for Students. The 'Recent Jobs' table lists six completed jobs, all of which failed. On the right, the 'View Published Source' pane shows the PowerShell script used for the automation:

```
5 $sqlConnection = new-object System.Data.SqlClient.SqlConnection
6
7
8 $sqlConnection.ConnectionString = "Data Source=tcp:vms-db-server1.database.windows.net,1433;Initial Catalog=master;User Id=vmsauto;Password=Pa$$w0rd;Encrypt=True;TrustServerCertificate=False"
9
10 $sqlConnection.Open()
11
12
13 Write-Output "Azure SQL database connection opened"
14
15
16
17 # Define the SQL command to run
18
19 $sqlCommand = new-object System.Data.SqlClient.SqlCommand
20
21 $sqlCommand.CommandTimeout = 120
22
23 $sqlCommand.Connection = $sqlConnection
24
25
26 Write-Output "Issuing command to run stored procedure"
27
28
29 # Execute the SQL command
30
31 $sqlCommand.CommandText= 'exec [dbo].[expireAppointments]'
32
33 $result = $sqlCommand.ExecuteNonQuery()
34
35
36 Write-Output "Stored procedure execution completed"
37
38
39 # Close the SQL connection
40
41 $sqlConnection.Close()
42
43
44 Write-Output "Run completed"
```

Figure 8.4: Stored procedure scheduled automation with Azure Runbook

9 System Design

9.1 Project Implementation Diagram

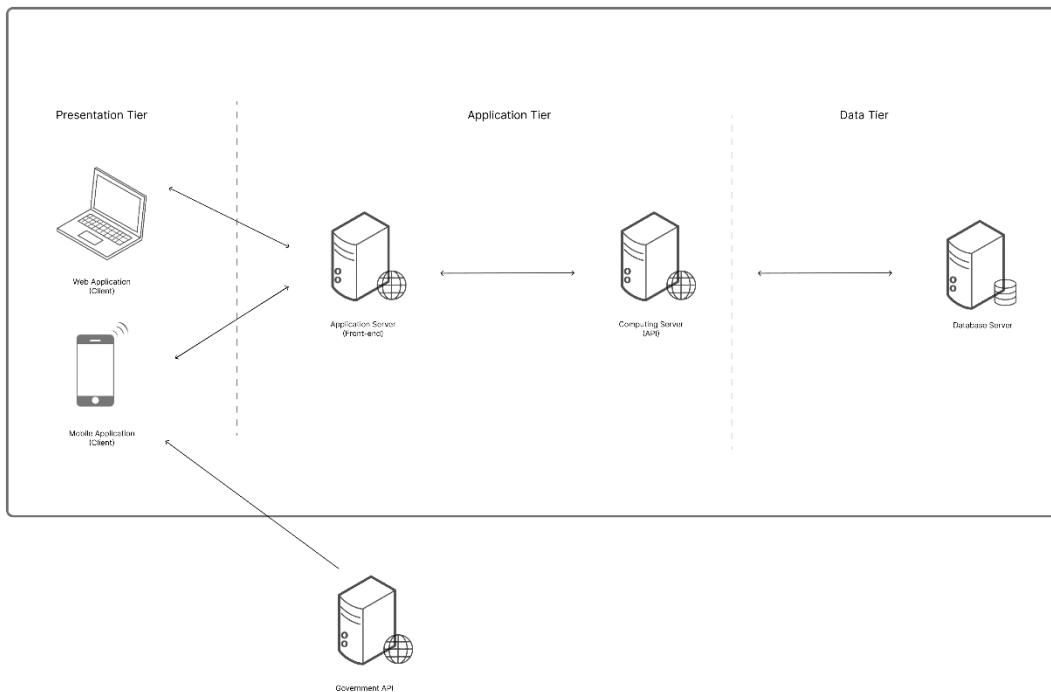


Figure 9.1: Project implementation diagram

9.2 Activity Diagrams

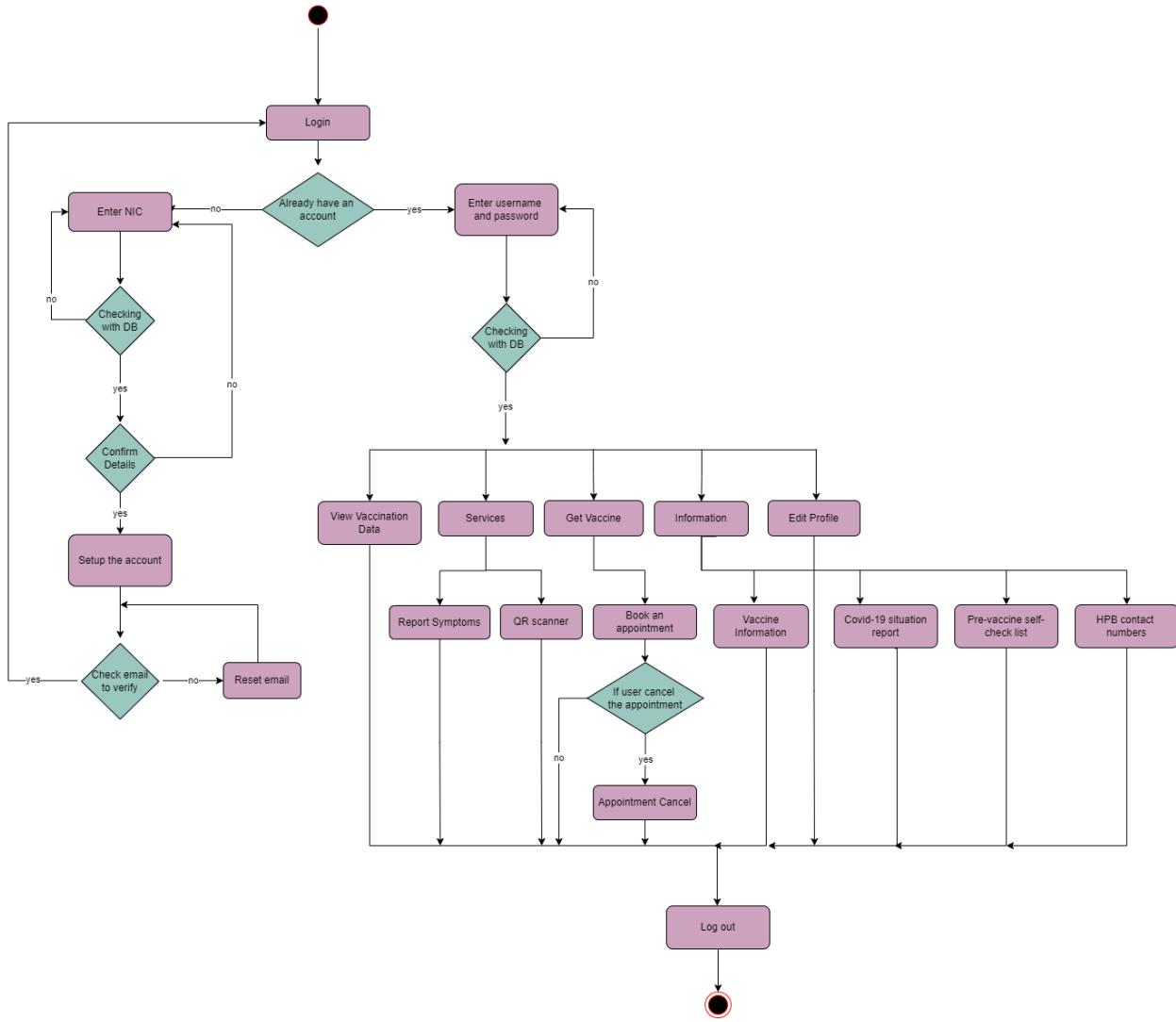


Figure 9.2:User- Activity diagram

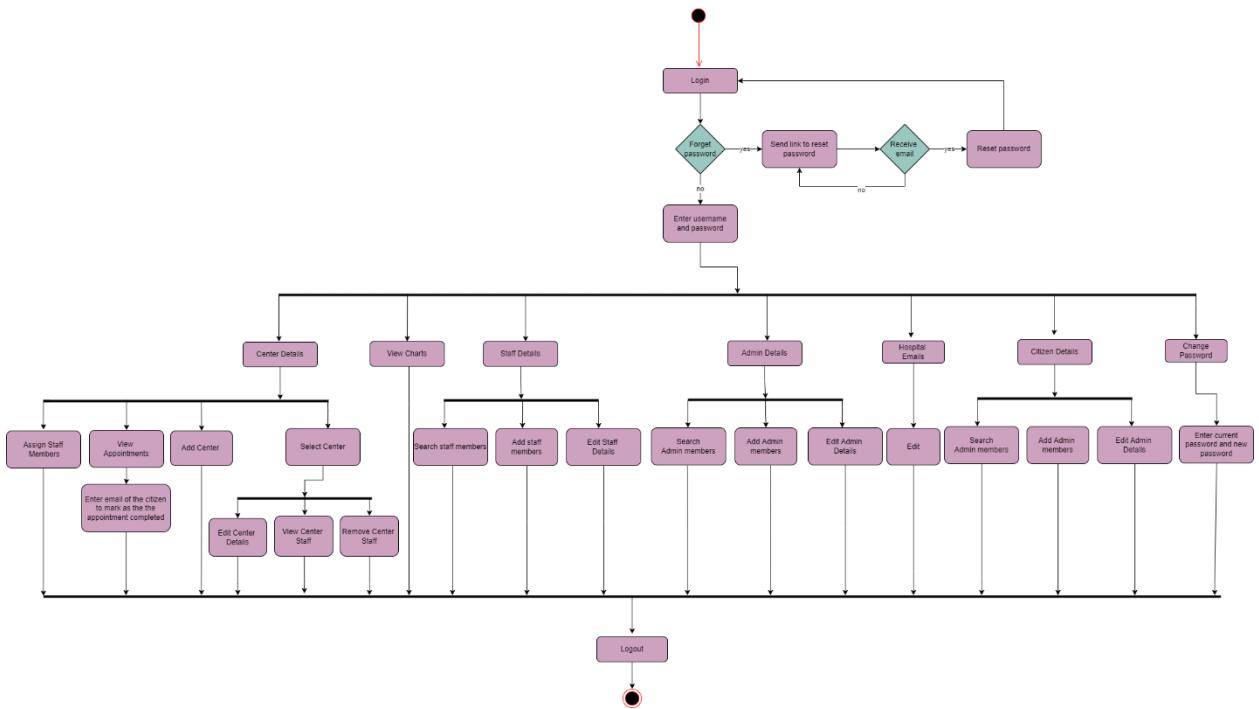


Figure 9.3: Admin- Activity diagram

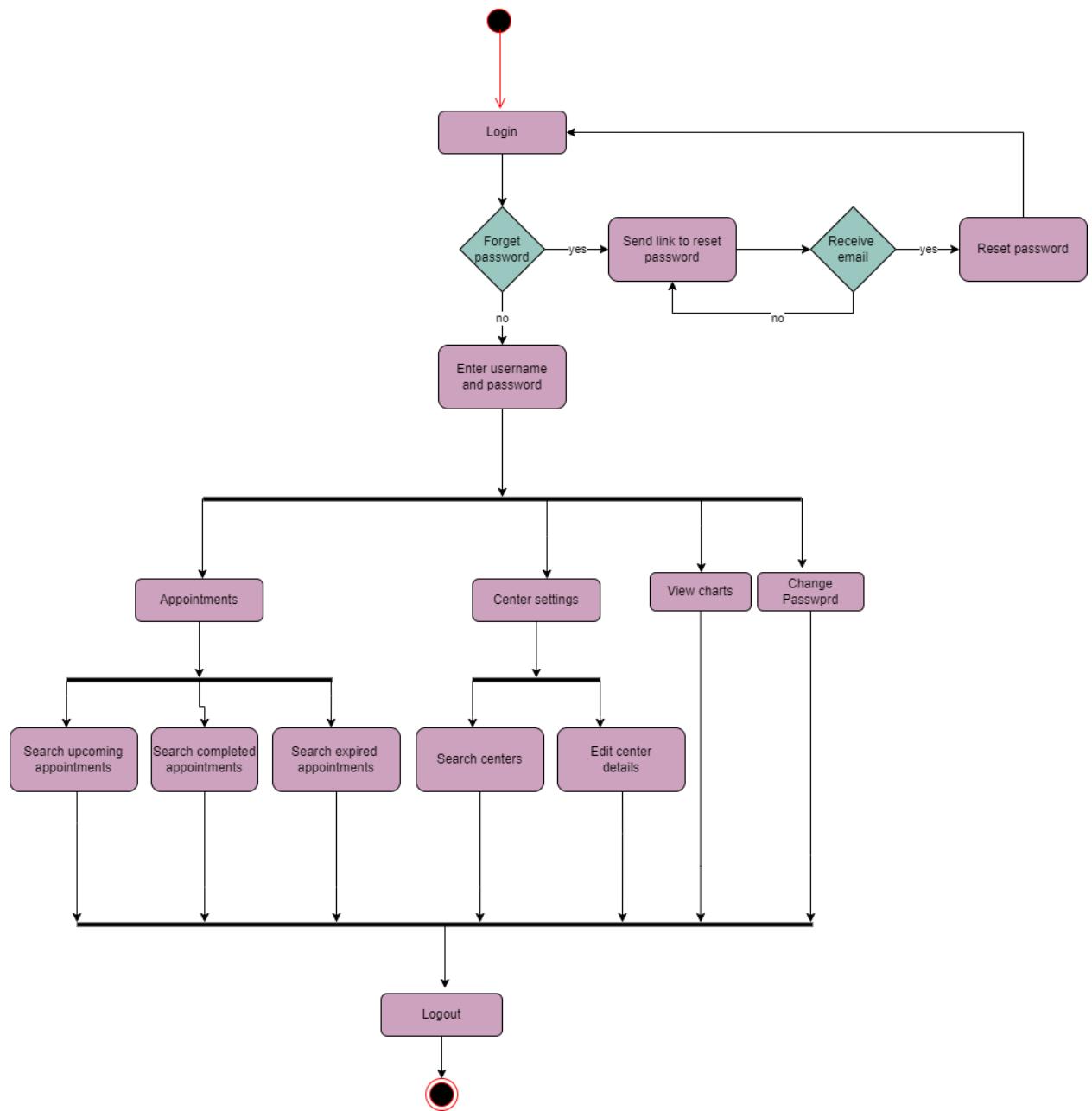


Figure 9.4: Staff- Activity diagram

9.3 Use-Case diagram

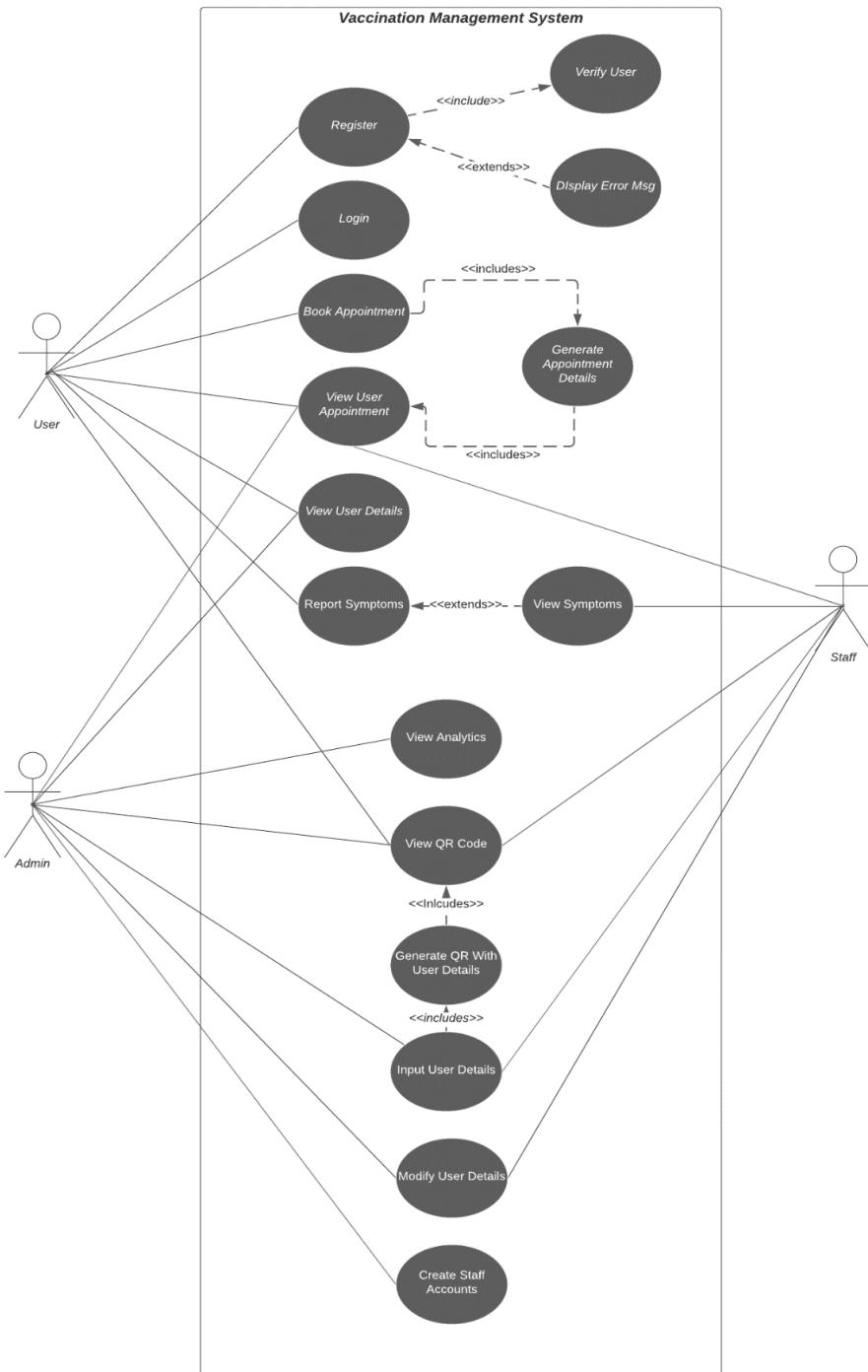


Figure 9.5: Use-Case diagram

10 Application Implementation

10.1 VMS Mobile application

Mobile application is focused on the citizen as most citizens use mobile phones. The flow of the application is designed very simple to make it more user-friendly.

10.1.1 Login

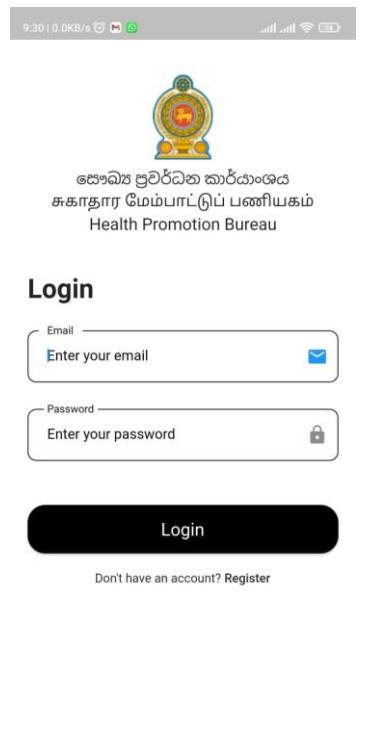


Figure 10.1: Mobile login screen

If the user already has an account, the user can log in to the system. Enter username and password correctly and click the login button. If not, the user must register to the system first by clicking the link below the login button.

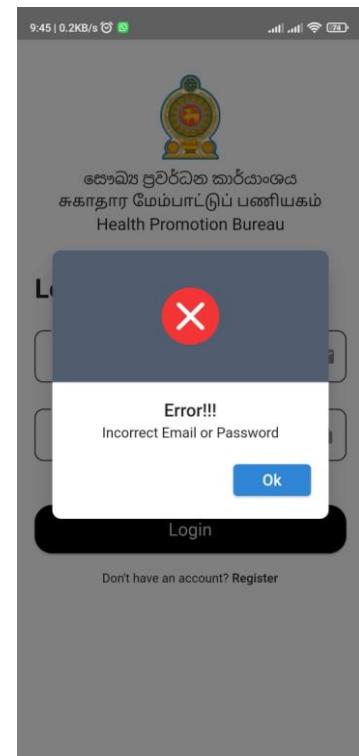


Figure 10.2: Incorrect credentials error

If the username or password user entered is incorrect, the error message will be displayed.

10.1.2 Registration

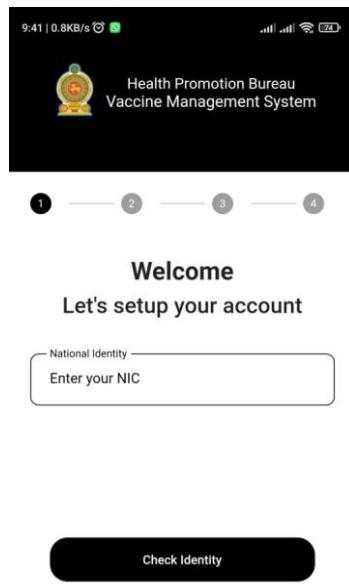


Figure 10.3: Registration-step 1

Users must enter users NIC number correctly in the first step. Then, click the check identity button to check if the entered value is in the government database.

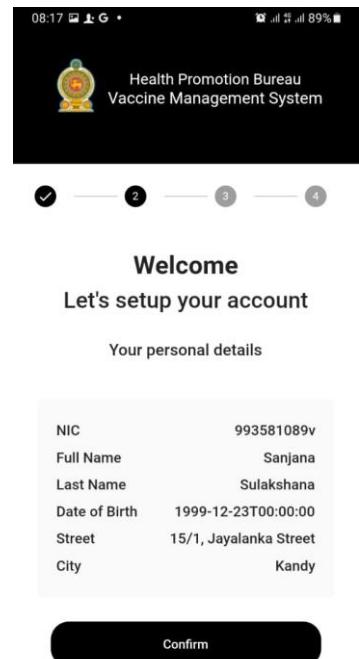


Figure 10.4: Registration-step 2

Then to continue the registration process, click the 'Confirm' button.



Figure 10.5: Registration-step 3

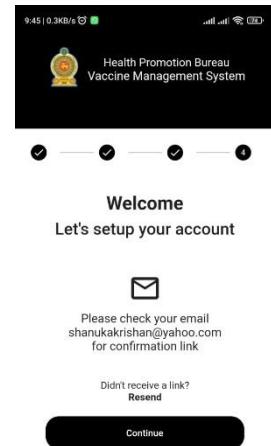


Figure 10.6: Registration-step 4

In the third phase, the user must enter a username, email, and password and retype the password and phone number. Then click create account button.

Finally, an email is sent to the users' email to confirm the email address. Check the user's email to see whether the user got the confirmation email, and click the continue button. If not, click resend link above the continue button.

10.1.3 Home



Figure 10.7: Home screen

10.1.4 Appointment

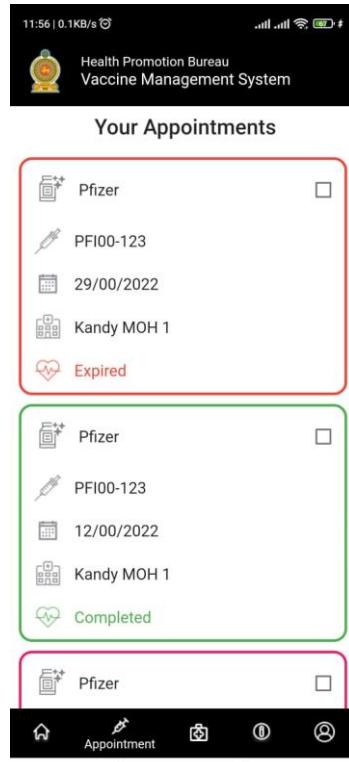


Figure 10.8: User appointments

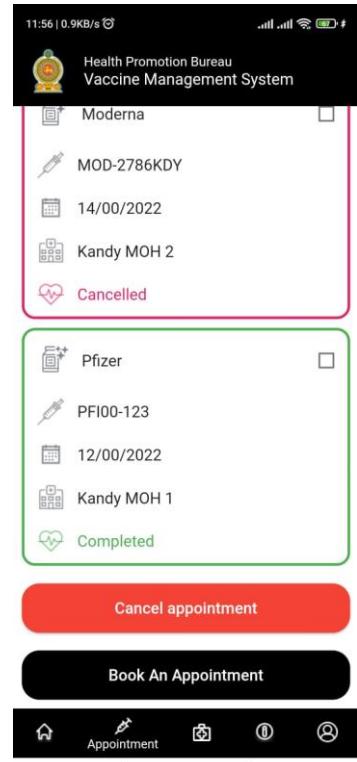


Figure 10.9: User appointments

- When the user goes to the appointment interface, the user can see the completed, ongoing, and expired appointments.
- Users can cancel the ongoing appointments by clicking the cancel appointment button after selecting the appointment user want to cancel.
- Also, user can book an appointment by clicking the book appointment button.



Figure 10.10: Add an appointment

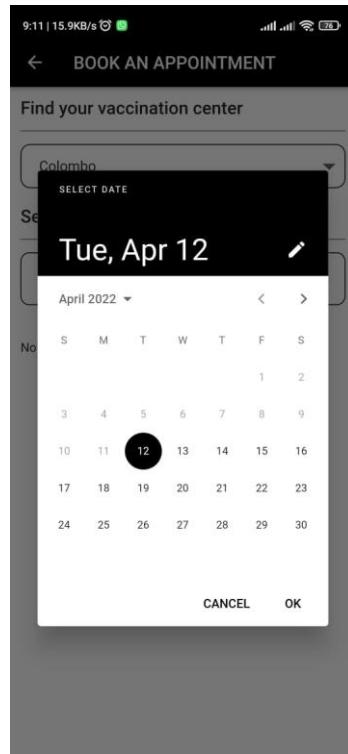


Figure 10.11: Select a vaccination date



Figure 10.12: Select user district

- When the user clicks the book an appointment button, the user will get an interface to select the vaccination centre
- When user click on the dropdown for selecting the vaccination centre, all the districts are listed here and select a date for the booking.

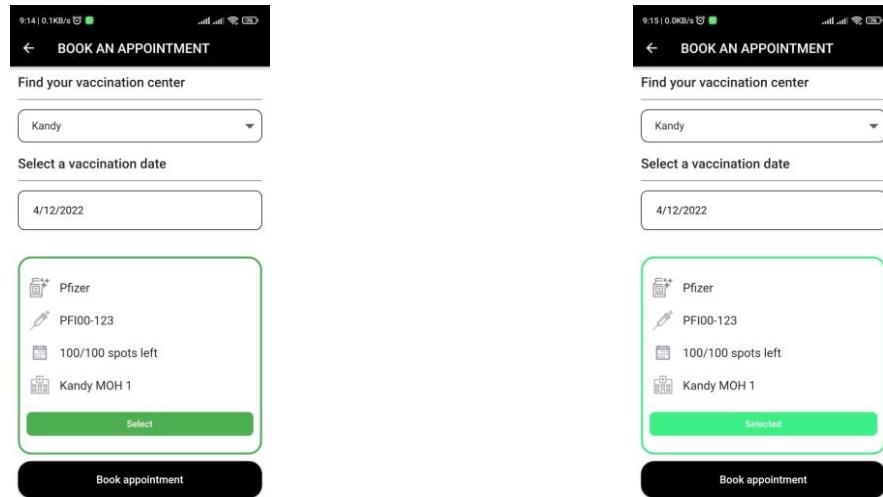


Figure 10.13: Display vaccination centres for the selected district

Figure 10.14: Select a vaccination centre

User can select any vaccination centre according to users preference and click on the book appointment button.

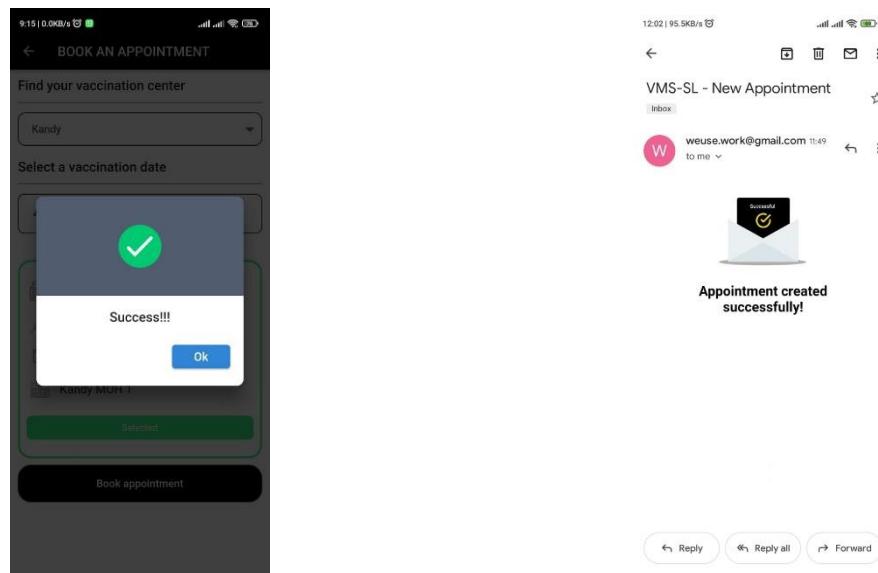


Figure 10.15: Success message after the appointment booking

Figure 10.16: Email received

After clicking the 'Book appointment' button, the user will get a success message.

Users will receive an email also for the newly created appointment.



Figure 10.17: Confirm cancelling an appointment

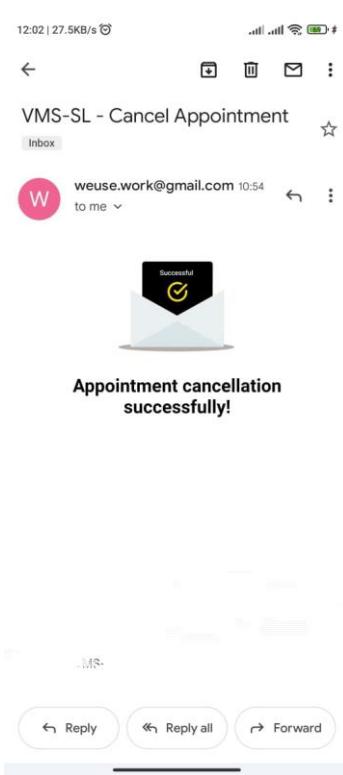


Figure 10.18: Email for the cancelled appointment

- If the user wants to cancel a booked appointment, the user must select the appointment card user wishes to cancel and click the cancel button.
- Click okay to confirm that the user wants to delete the appointment, or the user can click cancel to cancel the appointment cancel.
- An email is also sent to users email address to confirm that the booked appointment has been cancelled.

10.1.5 Services

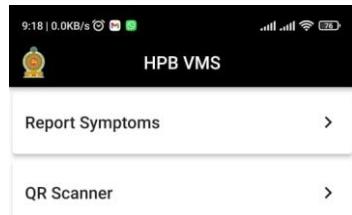
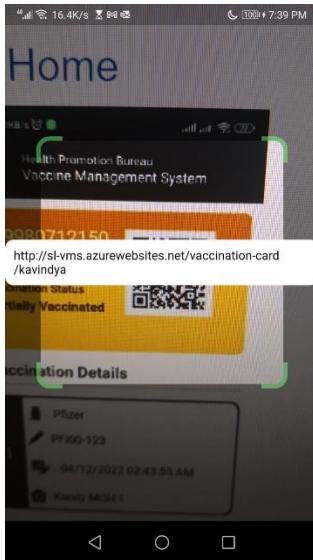
A screenshot of a "Report Symptoms" form. It starts with a dropdown menu for "Your Location" set to "Colombo". Below it, a question asks if the user has symptoms like "cough, fever, loss of smell, brain fog?". There are two radio button options: "Yes" and "No". Another question asks if the user took a COVID test, with similar "Yes" and "No" radio button options. At the bottom is a large "SEND REPORT" button.A screenshot of the same "Report Symptoms" form as the previous one, but with different user inputs. The "Your Location" dropdown is now set to "Kandy". For the symptom question, the "No" radio button is selected. For the COVID test question, the "No" radio button is also selected. The "SEND REPORT" button is at the bottom.

Figure 10.19: Services

Figure 10.20: Report symptoms

Figure 10.21: Report symptoms - choose answers

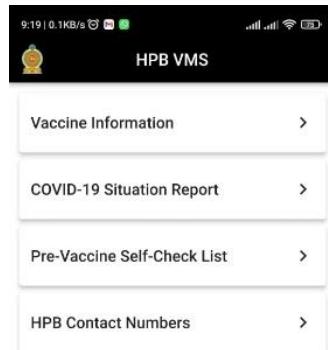
- Reporting symptoms and QR scanning are the services provided by the system.
- When the user clicks the option report symptoms, this is the interface user gets.
- Users can select users locations and the answers to the questions. Then click Send Report to send the form user filled out.



When the user clicks the QR scanner option, the user can scan the QR code like this.

Figure 10.22: QR code scanner

10.1.6 Information



- This is the interface user get when user clicks the information icon in the menu bar.
- Users can have information on the vaccines, COVID-19 situation report, pre-vaccine self-check list and Health Promotion Bureau contact numbers.



Figure 10.23: Information



Figure 10.24: Vaccination information



Figure 10.25: COVID-19 situation report

All the vaccine types are listed here with the basic information about the vaccine.

COVID-19 situation report display the total confirmed cases, active cases, daily cases, recovered, and death counts.

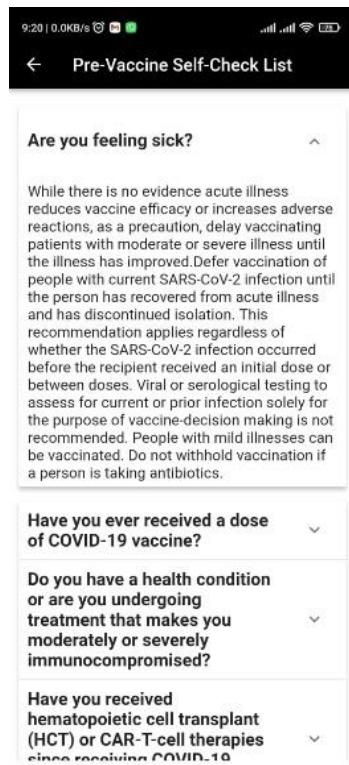


Figure 10.26: Pre-vaccine self-checklist

HPB Contact Numbers	
Health Promotion Bureau	011 2696606
Health Promotion Bureau	011 2696142
Suwasariya Hotline	1999
Epidemiology Unit	011 269511
Quarantine Unit	0112112705
Disaster Management Unit	0113071073

Figure 10.27: HPB contact numbers

- Users can have a look at the pre-vaccine self-check list to get the answers to the problems users have related to the vaccines.
- The questions and answers are listed, and the user can expand the question to view the answer.

Contact numbers of the Health Promotion Bureau and COVID-19 related units are given to user like this.

10.1.7 Profile

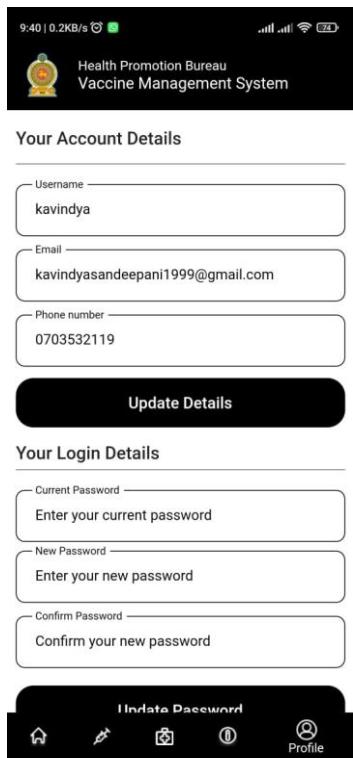


Figure 10.28: Profile

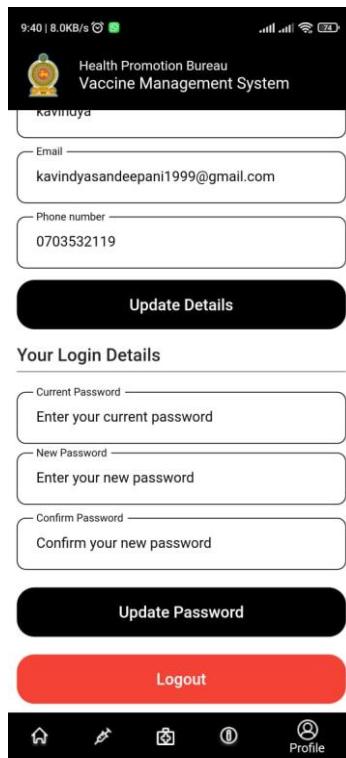


Figure 10.29: Profile- Change Password

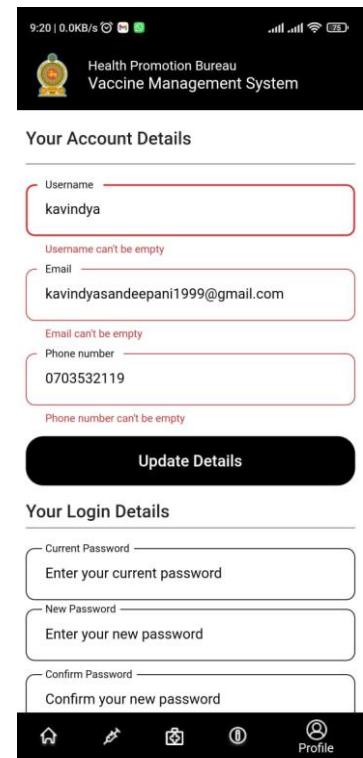


Figure 10.30: Validation when trying to update details without entering any data

- Users can click the update details button to update their username, email address and phone number.
- Users can update the password also by entering the current password and a new password. Users can log out from the system by clicking the logout button.
- If user click the button without editing the fields, validation messages are displayed like in the image.

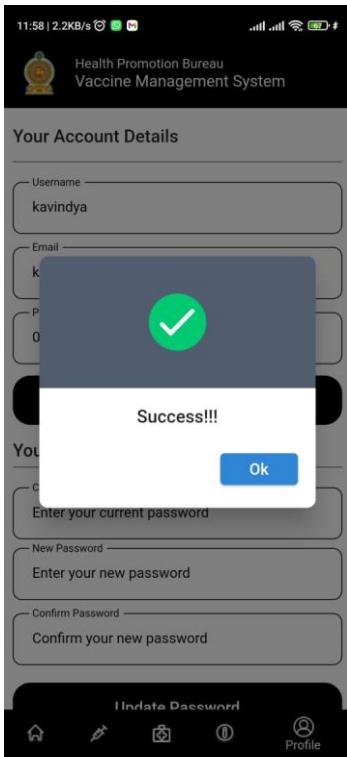


Figure 10.31: Successfully updated account details

When users update users account details successfully, the user will get a success alert like this.

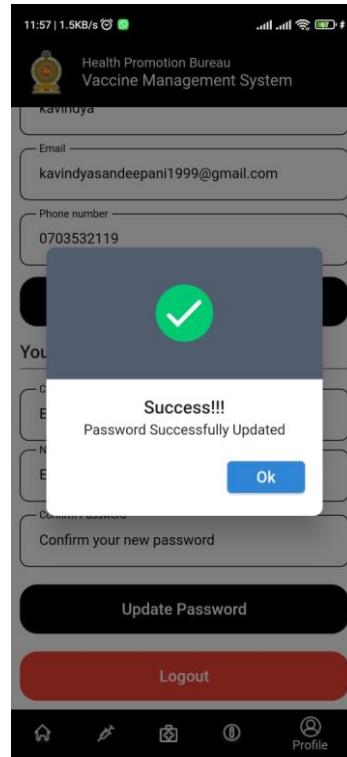


Figure 10.32: Successfully updated the password

When users update users login details successfully, the user will get a success alert like this.

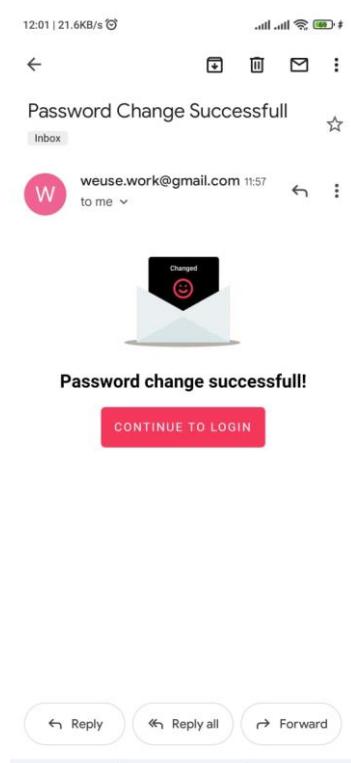


Figure 10.33: Email received for password change

To notify that the passwords are updated successfully, an email is also sent to the users' email addresses.

10.2 VMS Web application

VMS web application is for staff members and admins only. To manage the system and according to users' functions, team members decide to have both web and mobile applications.

10.2.1 General

10.2.1.1 Login

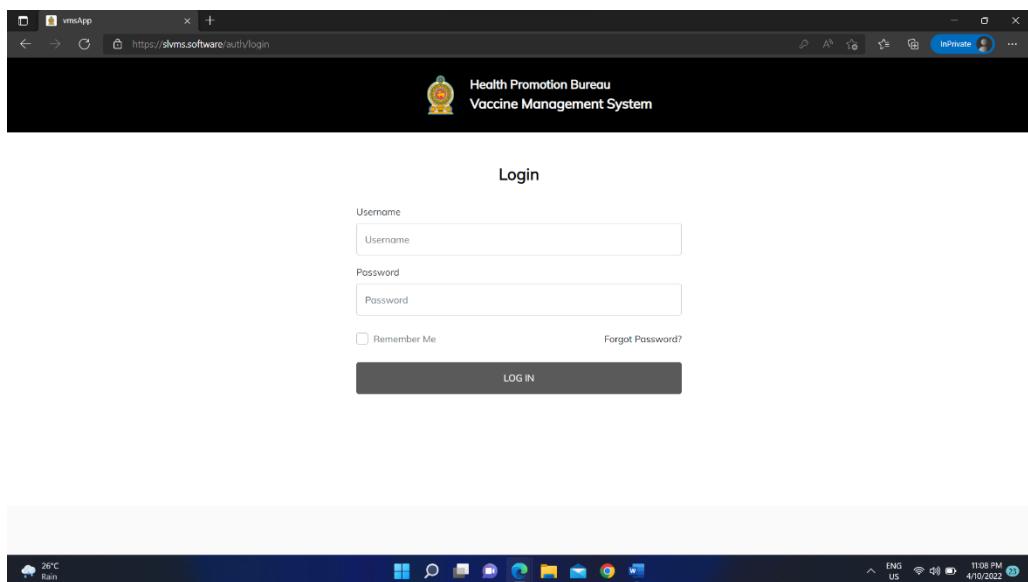


Figure 10.34: Web application login page

This is the first interface user meet when the user visits the web application of Health Promotion Bureau VMS. If the user is an admin or a staff member, the user must log in first to the system through this interface. The user must enter the correct username and password into the input fields. If the user entered invalid inputs, validation messages would be displayed below the input fields. If a user wants to keep themself as remembered, tick Remember Me checkbox. If the user forgot the user's password, the user could click the 'Forgot Password' link. Enter the username and password correctly, click the 'LOG IN' button and log in to the system.

10.2.1.2 Forgot password

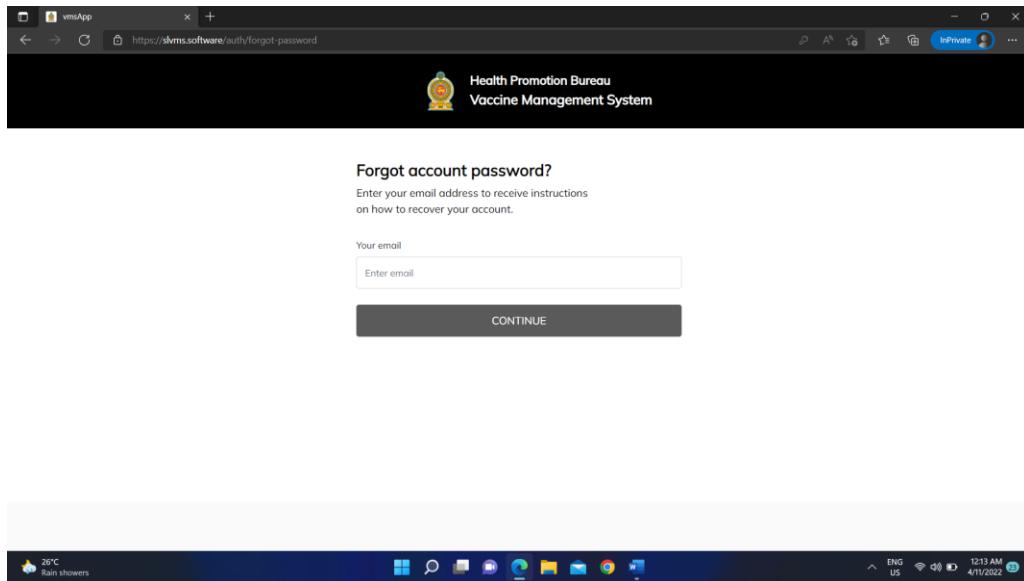


Figure 10.35: Forgot password

When the user clicks the 'Forgot Password' link, the user will redirect to this interface. Here, the user must enter the user's email address. Then, click the CONTINUE button. After that, the user must check the user's email as the user will receive an email to reset users password.

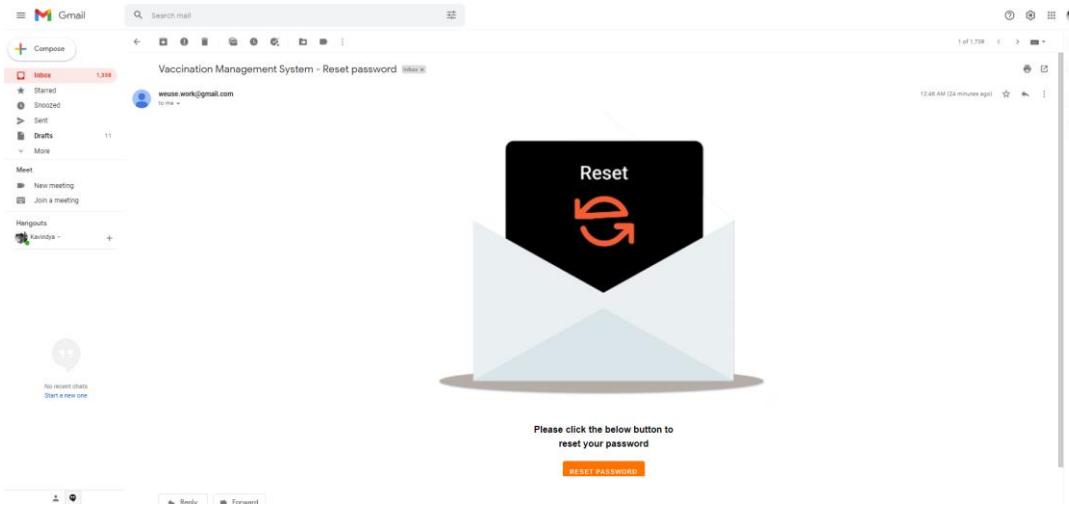


Figure 10.36: Email to reset the password

Users can see an email body like this when the user opens the email. Click the RESET PASSWORD button to reset users password.

10.2.1.3 Reset password

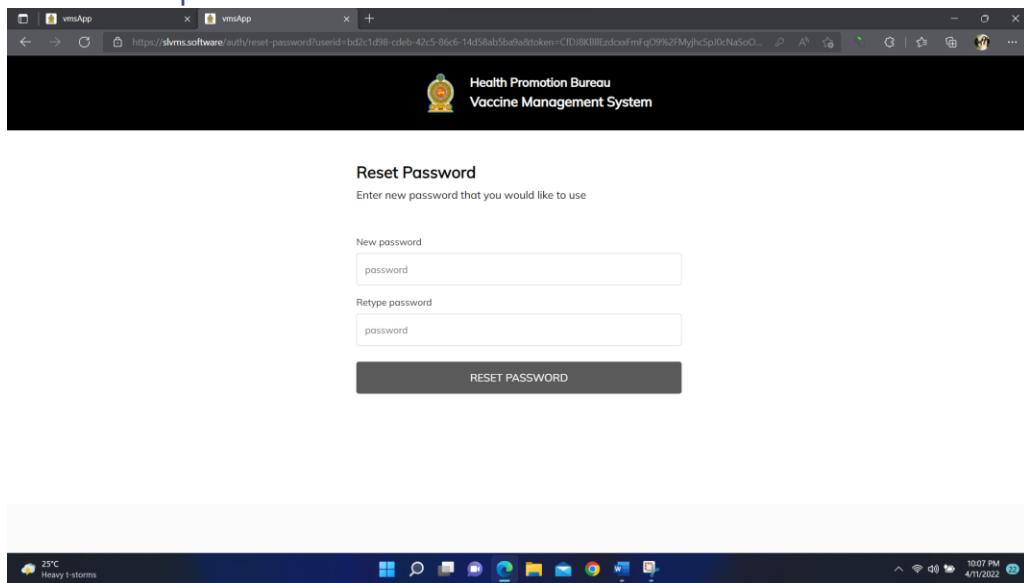


Figure 10.37: Reset password

Users will have this interface when user click the RESET PASSWORD button at the above interface. By entering the new password, the user can reset the user password. Retype the users' passwords to confirm the entered password. After that, the user will redirect to the login page and log in to the system by entering users username and new password.

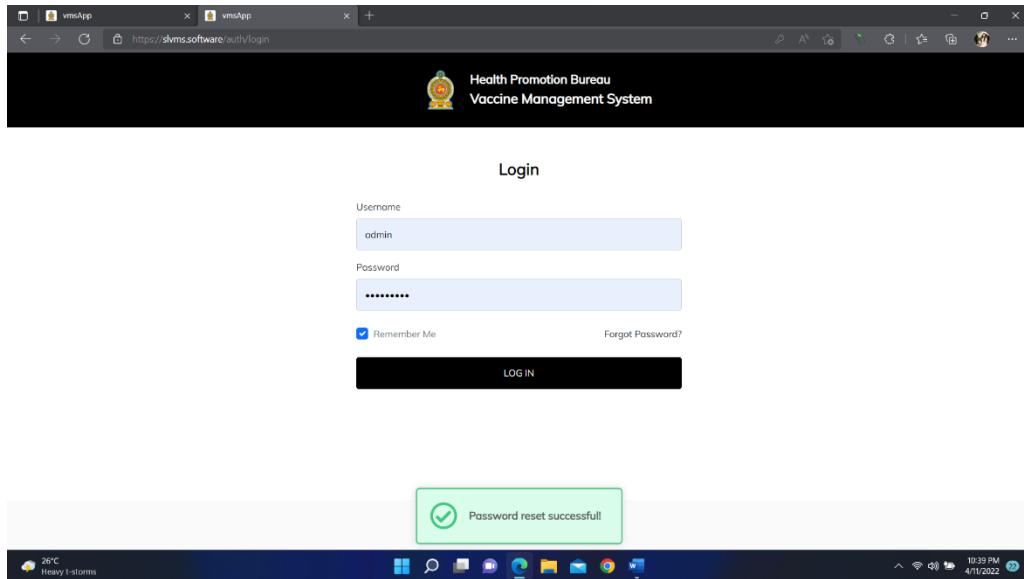


Figure 10.38: Password reset success alert

User will get a success alert and redirect to the login page.

10.2.2 Admin

10.2.2.1 Admin dashboard

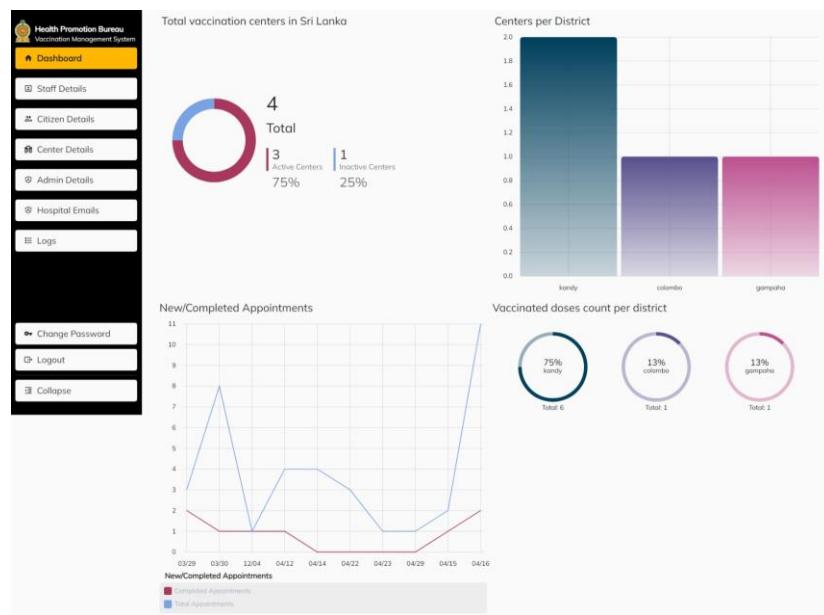


Figure 10.39: Admin dashboard

If the user is an admin, this is the user's dashboard.

The left side menu has seven pages, and users can navigate to those pages by clicking those buttons. Users can collapse users' navbar by clicking on the collapse button at the bottom of the side navigation. Users can log out from the system by clicking the log out button.

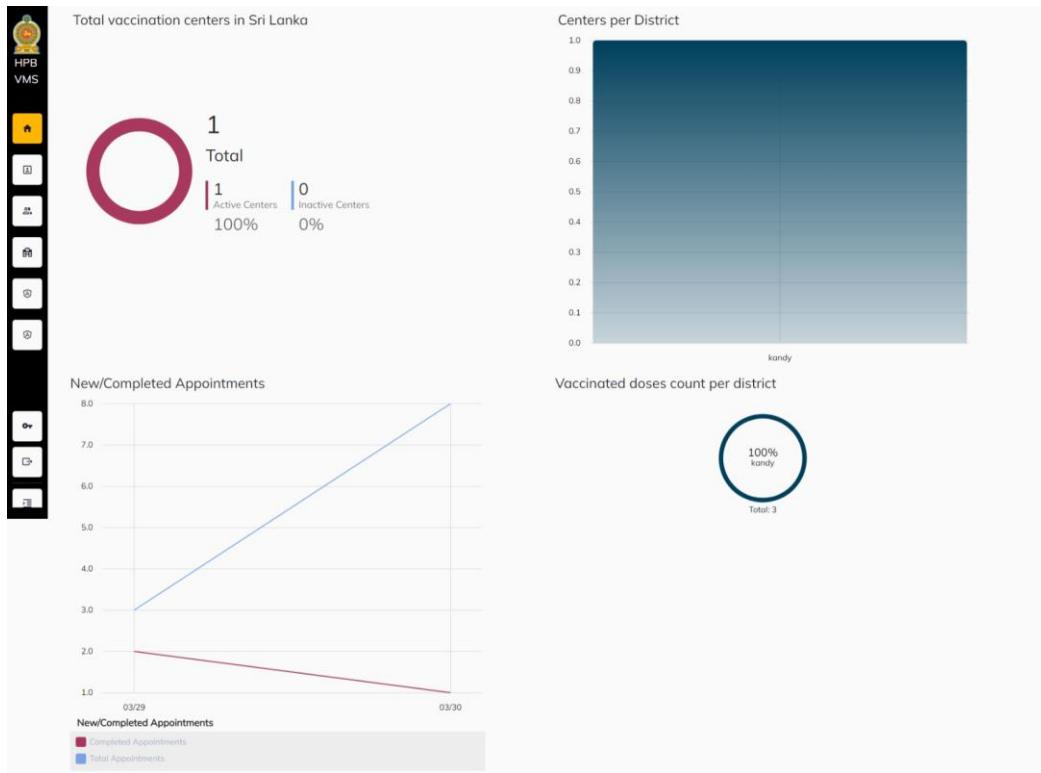


Figure 10.40: Admin dashboard

This is how the user can see the user's dashboard when the user collapses the side navigation. In this interface, the user can conclude the total vaccination centres in the country, completed and not completed appointments, no. of centres per district and vaccinated dose count per district.

10.2.2.2 Staff details

The screenshot shows a web application interface titled "vmsApp" at the top left. The URL is https://svlms.software/staff. The main content area is titled "Staff". On the left, there is a vertical sidebar with the following navigation options:

- Dashboard
- Staff Details** (highlighted in yellow)
- Citizen Details
- Center Details
- Admin Details
- Hospital Emails
- Change Password
- Logout
- Collapse

The main content area has a search bar at the top with the placeholder "search for username, email...". Below the search bar is a table with the following columns: NAME, NIC, VACCINATION CENTER, CONTACT NUMBER, and STATUS. One row is displayed in the table:

NAME	NIC	VACCINATION CENTER	CONTACT NUMBER	STATUS
Test Staff sonjanow99@gmail.com	993581089v	Kandy MOH 1	0771994147	ACTIVE

Below the table, there is a "Show" dropdown set to "10 rows" and a message "Showing 1-1 of 1 staff". At the bottom right of the content area, there is a small grid icon with the number "1".

Figure 10.41: Staff details

- When the user clicks staff details in the side navigation, the user will have this interface.
- Here, users can add new staff members and search for staff members by username, email, vaccination centre, contact number, status, or NIC.
- Users can select the number of rows to display per page.

10.2.2.3 Create staff account

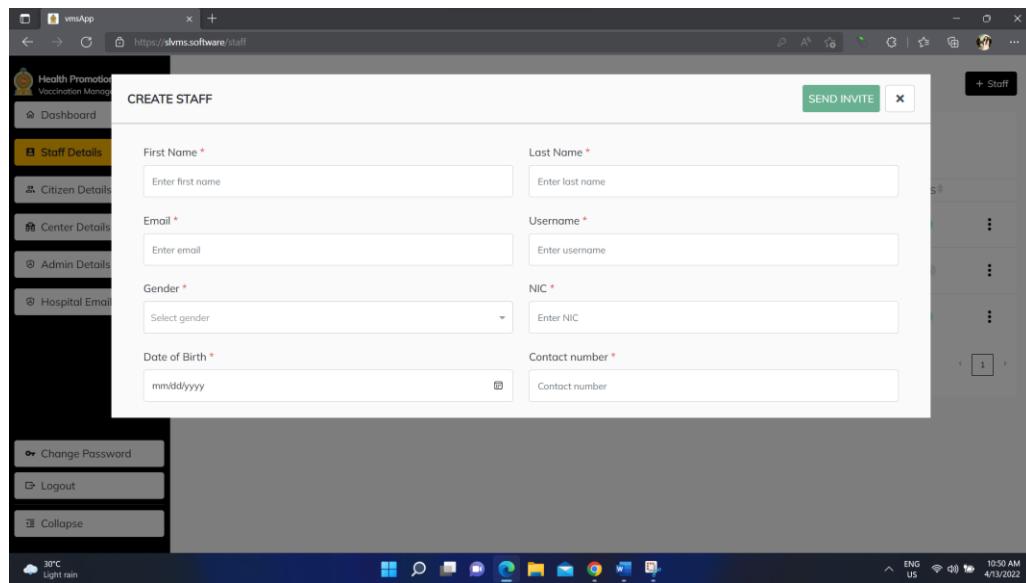


Figure 10.42: Create a staff account

When the user clicks the add staff button, the user will have this modal. Fill in all the fields as required and click the send invite button on the top corner of the modal.

10.2.2.4 Email received by the staff member

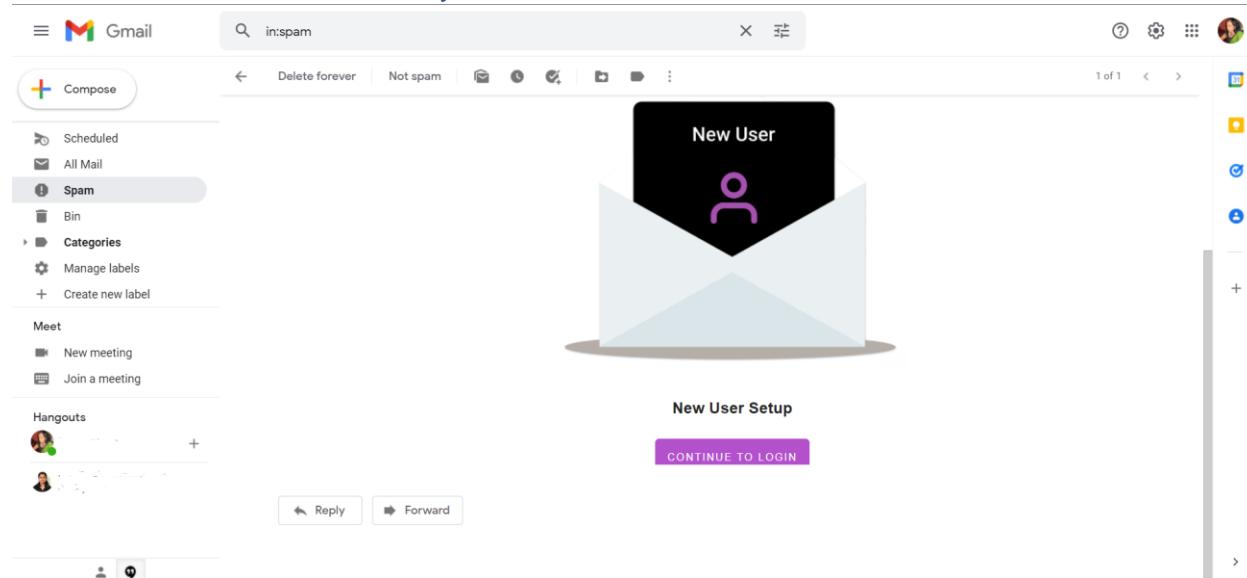


Figure 10.43: Email received by the staff member to set up the password

An email is sent to the email address of a new staff member. By clicking the button, user can reset the password.

10.2.2.5 Setting up a new password for the staff account

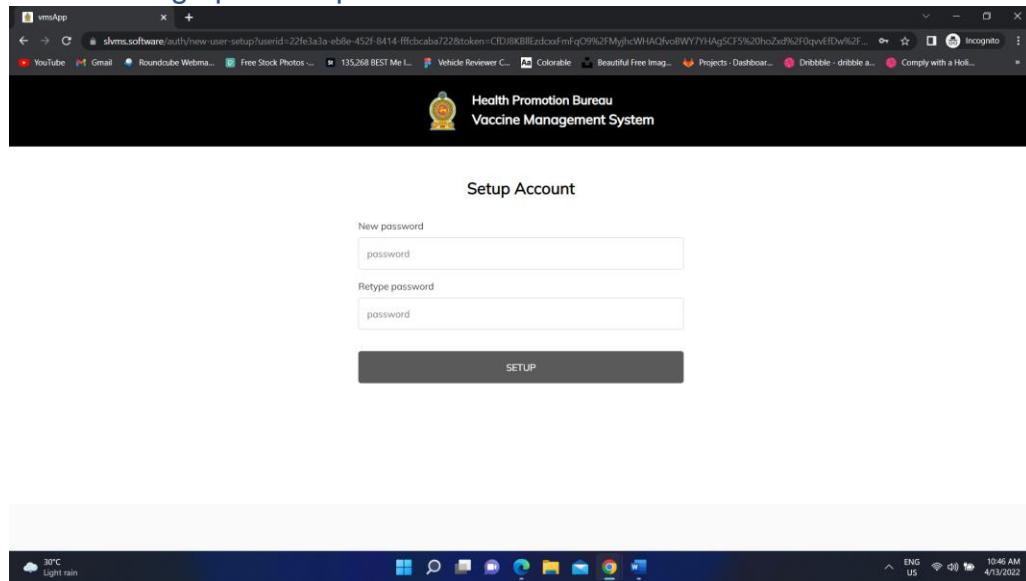


Figure 10.44: Set up a new password

The new staff member has to enter a new password, retype again to confirm the password and click the setup button.

10.2.2.6 Kebab menu in staff details data row

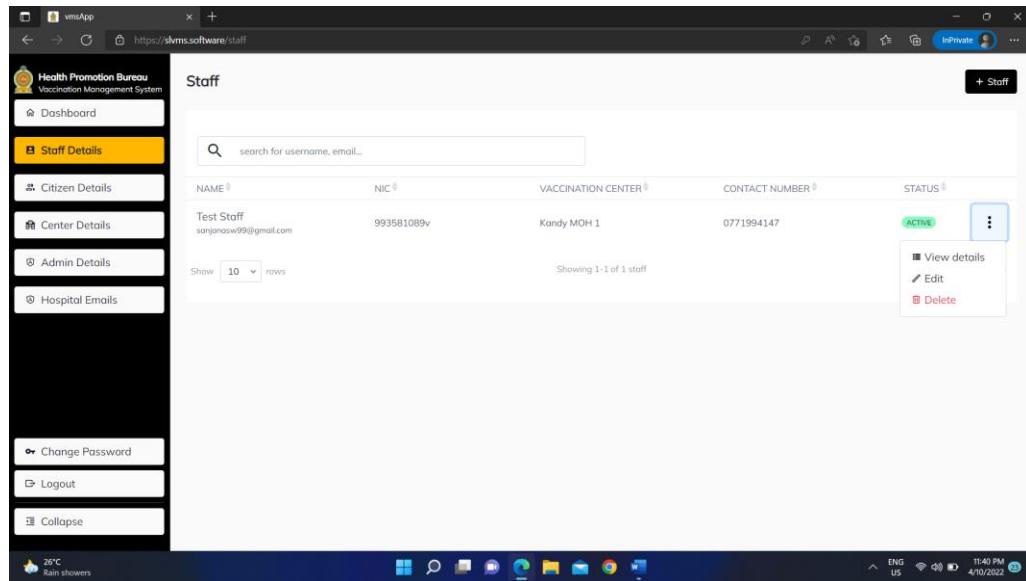


Figure 10.45: Staff details

When the user clicks the kebab menu icon on the last column of the data row, the user can get a dropdown with view details and edit and delete options.

10.2.2.7 View staff details

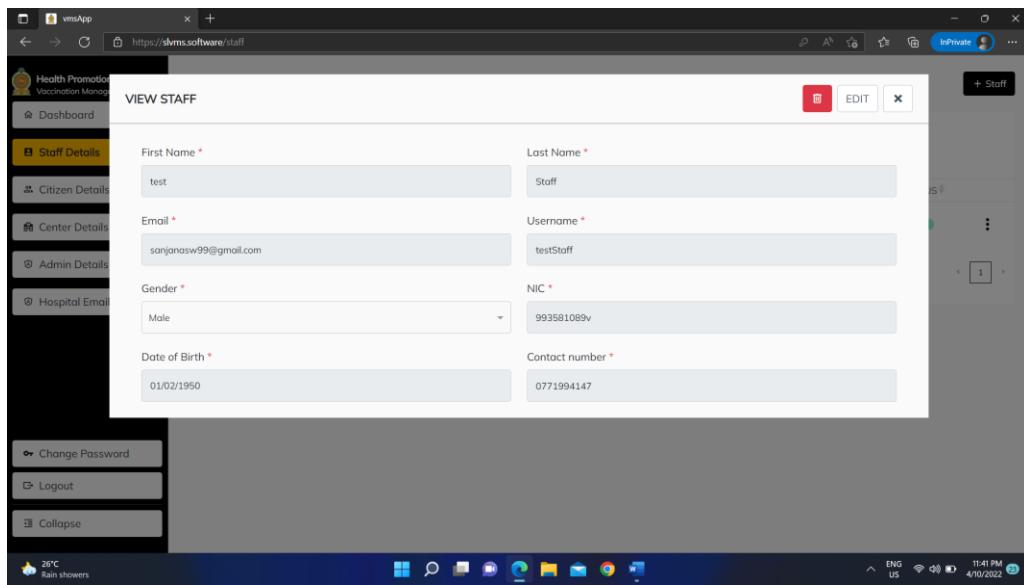


Figure 10.46: View staff member's details

When the user clicks the select edit option in the kebab menu, the user can have this interface and view the staff member's details.

If the user wants to edit or delete, the user can click the delete icon or edit button on the top of the modal.

10.2.2.8 Edit staff details

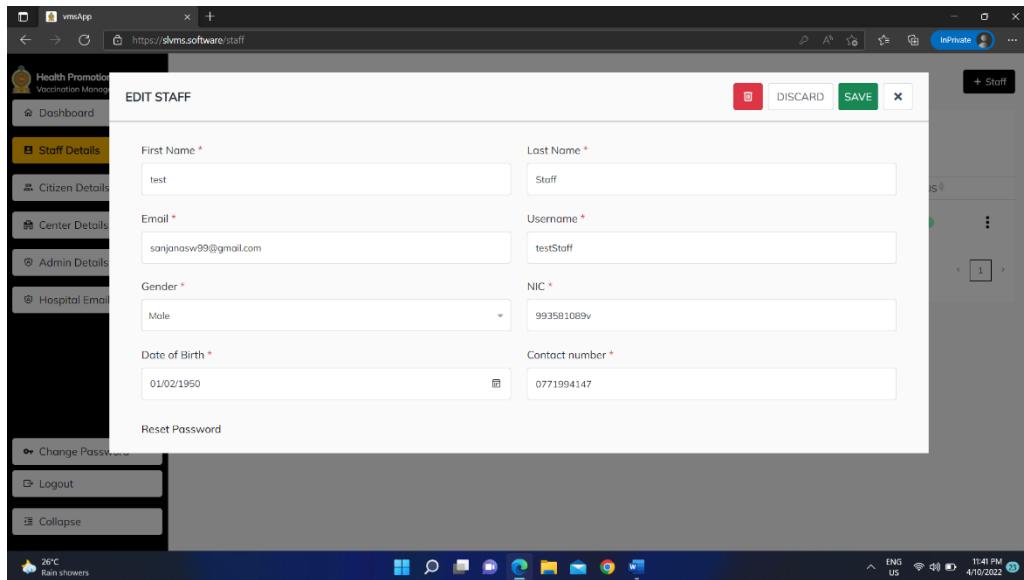


Figure 10.47: Edit staff member's details

Users can have this modal by clicking the edit button at the previous modal or the edit details option in the kebab menu.

If the user doesn't want the modifications to be saved, the user has done for the staff member details, and user can click the discard button on the top of the modal or if the user wants to keep those details, user can click the save button right from the discard button.

10.2.2.9 Reset password in staff account

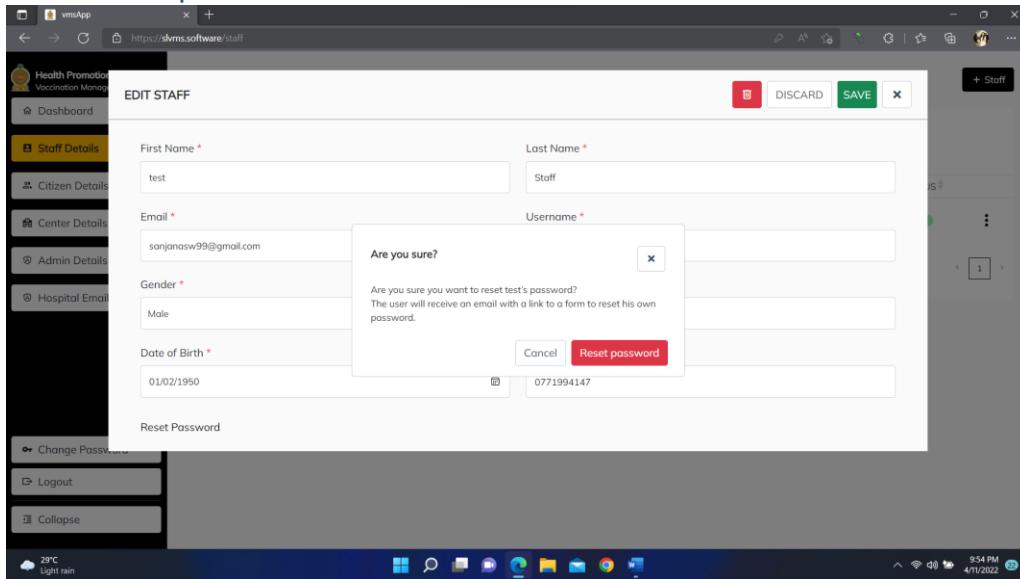


Figure 10.48: Confirm before sending the reset password link

- This is the interface user can see when the user clicks the reset password link.
- Users have to click the reset password button in the modal.
- Then the reset password link is sent to the staff member's email address.

10.2.2.10 Reset email sends successfully

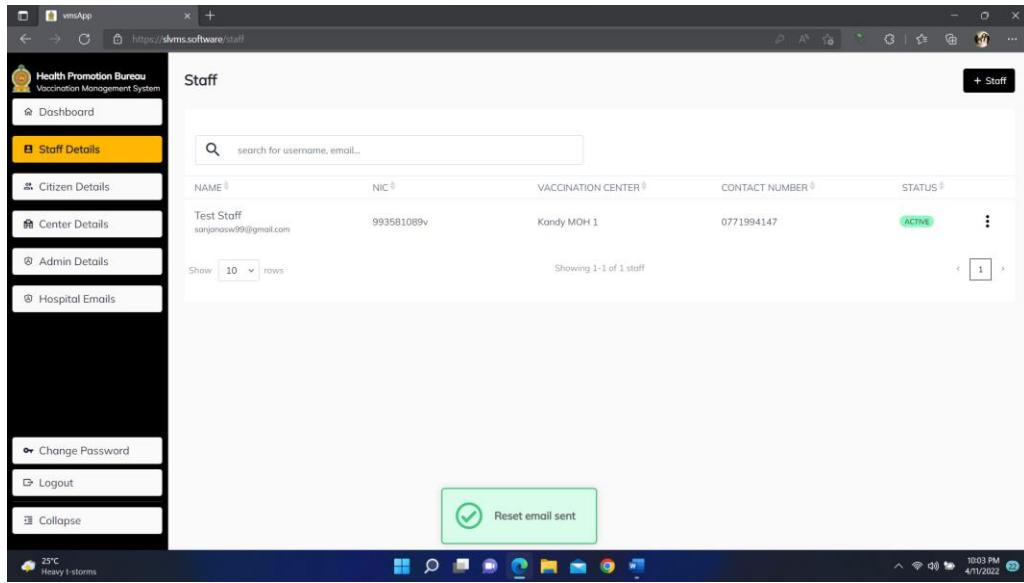


Figure 10.49: Alter for the email to reset the password is sent successfully

After sending the email successfully, the success alert will display like this.

10.2.2.11 Confirm deleting of staff account

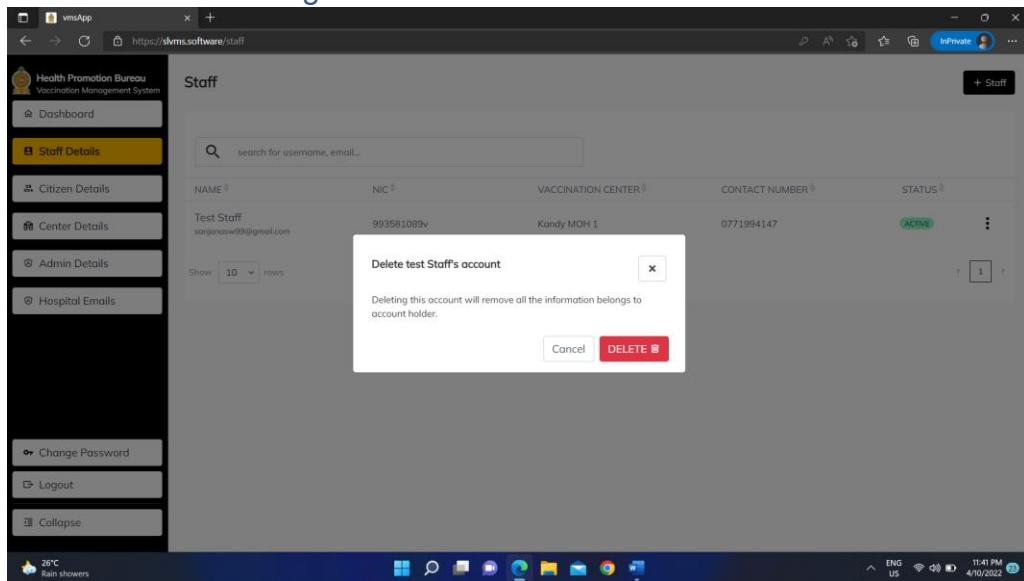


Figure 10.50: Confirm before deleting a staff account

Users have to confirm the delete by clicking the delete button in this modal, or else the user can cancel if the user doesn't want to delete the staff member.

10.2.2.12 Citizen details

NAME	NIC	DOB	VACCINATION STATUS	CONTACT NUMBER
Kavindya Sandeepanie kavindyasandeepani1999@gmail.com	199980712150	11/02/1999	Not Vaccinated	0703532119
Sanjana Witharage henavox412@flowminer.com	993581089v	12/23/1999	Received X	0761989993

Figure 10.51: Citizen details

Here, user can search citizens by username, email, NIC, DOB, vaccination status or contact number.

10.2.2.13 Kebab menu in citizen details

The screenshot shows a web application interface for managing citizens. On the left is a dark sidebar with various navigation options: Dashboard, Staff Details, Citizen Details (which is highlighted in yellow), Center Details, Admin Details, Hospital Emails, Change Password, Logout, and Collapse. The main content area has a header 'Citizens' and a search bar. Below is a table with two rows of citizen data:

NAME	NIC	DOB	VACCINATION STATUS	CONTACT NUMBER
Kavindya Sandeepanie kavindyasandeepani1999@gmail.com	199980712150	11/02/1999	Not Vaccinated	0703532119
Sanjana Witharage henavox412@flowminer.com	993581089v	12/23/1999	Received X	07619899

At the bottom right of the table, there is a kebab menu icon (three vertical dots) which is expanded to show options: View details, Edit, and Delete.

Figure 10.52: Citizen details

When the user clicks the kebab menu at the end of the data row, the user can see the dropdown list with view details and edit and delete options.

10.2.2.14 View citizen details

The screenshot shows a modal window titled 'VIEW CITIZEN'. It contains fields for First Name, Last Name, Email, Username, Gender, NIC, Date of Birth, and Contact number. The 'First Name' field is populated with 'kavindya' and 'Last Name' with 'sandeepanie'. The 'Email' field is 'kavindyasandeepani1999@gmail.com' and 'Username' with 'kavindya'. The 'Gender' dropdown is set to 'Female'. The 'NIC' field is '199980712150'. The 'Date of Birth' field is '11/02/1999' and 'Contact number' is '0703532119'. At the top right of the modal are 'EDIT' and 'X' buttons.

Figure 10.53: View citizen's details

When the user clicks the view details option, the user can see the citizen details.

10.2.2.15 Edit citizen details

The screenshot shows a web-based application window titled 'vmsApp' with the URL 'https://svms.software/citizen'. On the left, there is a vertical sidebar menu with options: Dashboard, Staff Details, Citizen Details (which is currently selected and highlighted in yellow), Center Details, Admin Details, and Hospital Details. Below the menu, it shows the weather as '26°C Rain showers'. The main content area is titled 'EDIT CITIZEN'. It contains several input fields: 'First Name *' with value 'kovindya', 'Last Name *' with value 'sandeepanie', 'Email *' with value 'kovindya.sandeepani1999@gmail.com', 'Username *' with value 'kovindyo', 'Gender *' with value 'Female' (selected from a dropdown), 'NIC *' with value '199980712150', 'Date of Birth *' with value '11/02/1999', and 'Contact number *' with value '0703532119'. At the top right of the form are three buttons: a red 'DISCARD' button, a green 'SAVE' button, and a close 'X' button. The bottom right of the screen shows system status icons for battery level, signal strength, and date/time.

Figure 10.54: Edit citizen's details

User can edit citizen details and save them.

10.2.2.16 Confirm reset password link

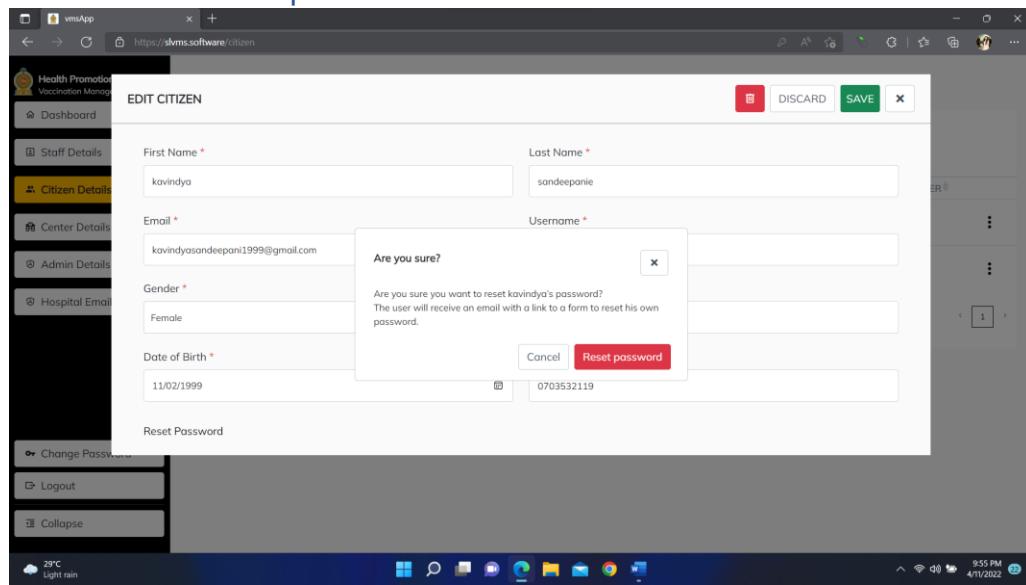


Figure 10.55: Confirm before the reset password link is sent

Click the reset password link and click the reset password button to send a reset password link to the citizen.

10.2.2.17 Successfully sent reset email

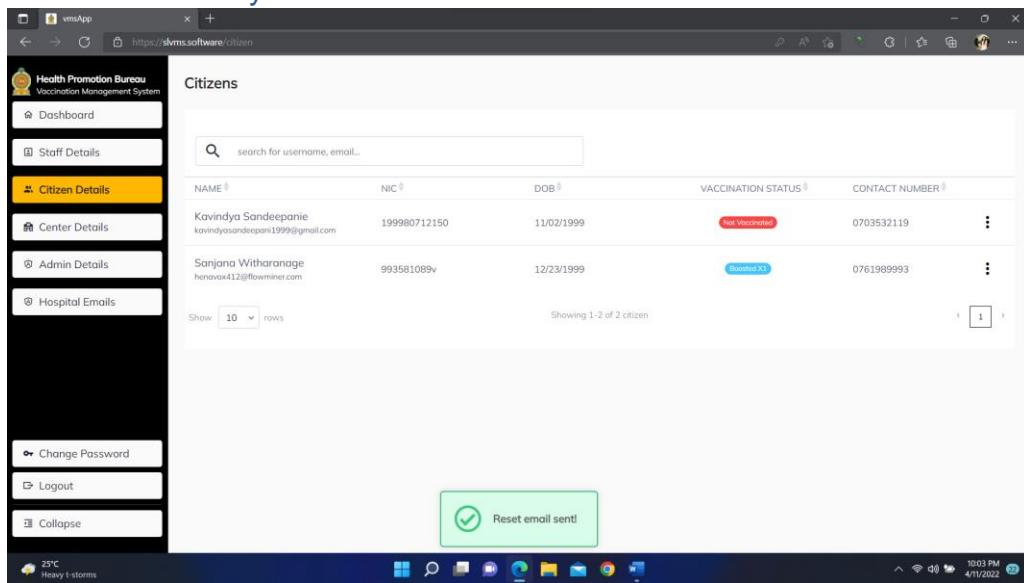


Figure 10.56: Success alert to reset email sent

After sending the email successfully, the success alert will display like this.

10.2.2.18 Confirm deleting a citizen account

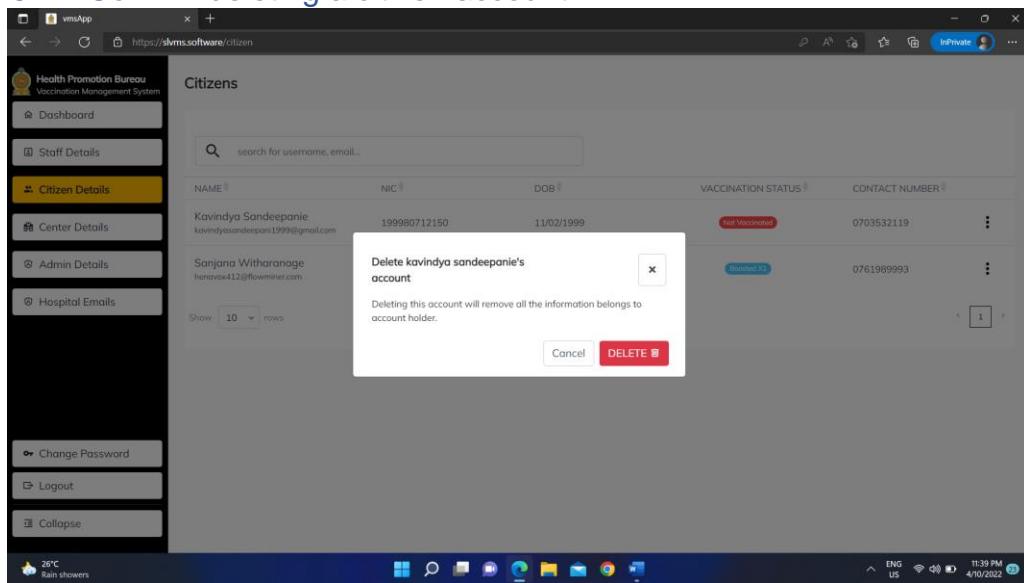


Figure 10.57: Confirm before deleting a citizen account

Click the delete button to delete the user.

10.2.2.19 Centre details

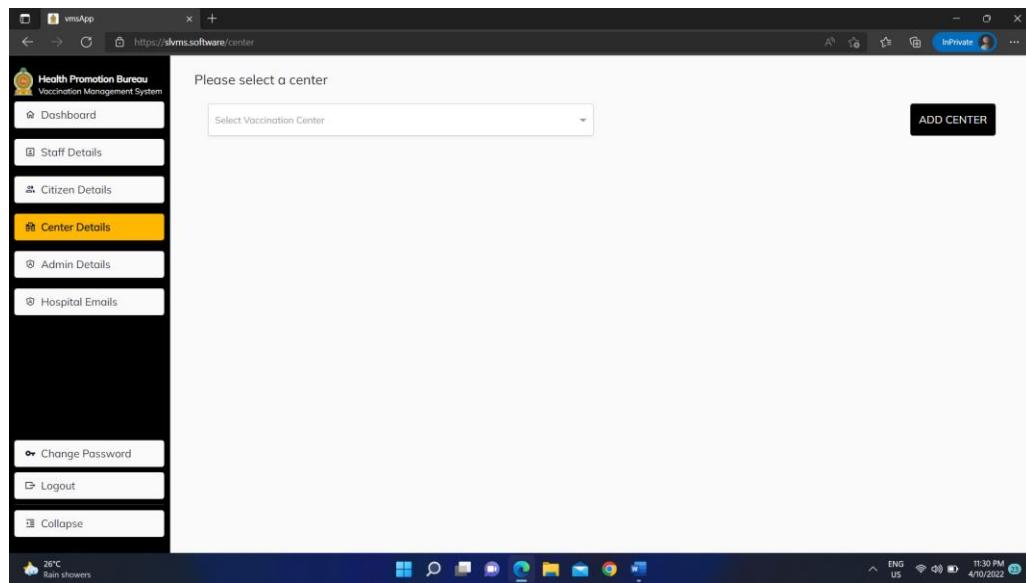


Figure 10.58: Vaccination centre details

User can select a vaccination centre from the dropdown list or add a vaccination centre to the system.

10.2.2.20 Add a new vaccination centre

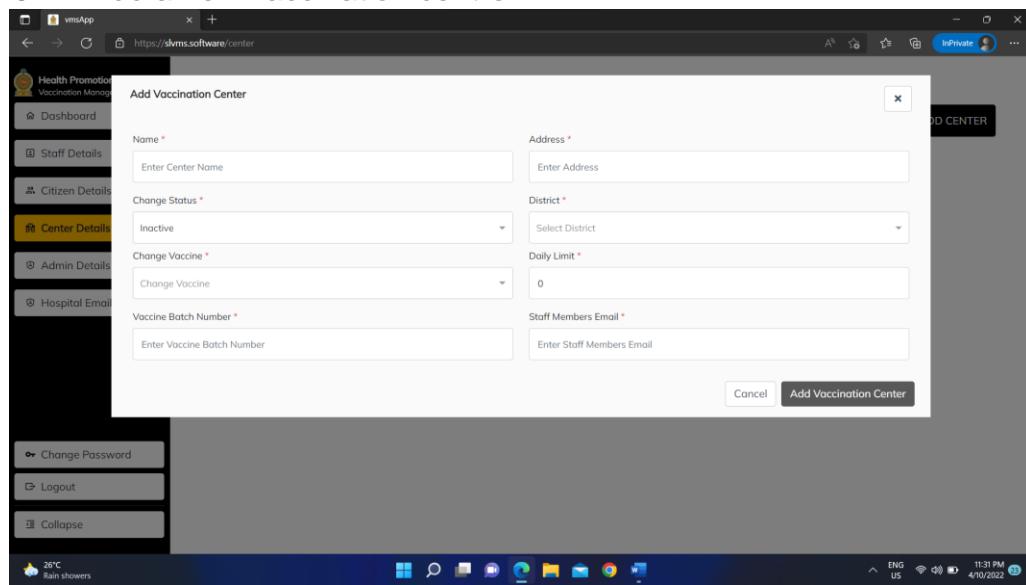


Figure 10.59: Add a new vaccination Centre

User have to fill all the fields and click add vaccination centre button.

10.2.2.21 View details of selected vaccination centre

The screenshot shows the 'Vaccination Center Information' page. On the left is a sidebar with navigation links: Dashboard, Staff Details, Citizen Details, Center Details (highlighted in yellow), Admin Details, Hospital Emails, Change Password, Logout, and Collapse. The main area has a dropdown menu set to 'Kandy MOH 1'. A large 'ADD CENTER' button is at the top right. Below it is an 'EDIT' button. The 'Vaccination Center Information' section contains fields: Name (Kandy MOH 1), Staff Count (1), Address (Dalada Maligawa, Kandy), District (Kandy), Status (Active). Another section shows Vaccine Brand (Pfizer), Daily Limit (100), Vaccine Batch Number (PFI00-123), and Daily Count (0). The 'Vaccination center staff' section lists one member: Test Staff (sanjanaw99@gmail.com), NIC (993581089v), Contact Number (0771994147), and Status (ACTIVE). A small red warning icon is next to the status. At the bottom, there's a pagination control showing page 1 of 1.

Figure 10.60: View details of the selected vaccination centre

When the user chooses a vaccination centre from the dropdown list, the user will get the details of the vaccination centre and the staff member details below.

10.2.2.22 Edit details of the selected vaccination centre

The screenshot shows the 'Vaccination Center Information' form. The 'Name' field contains 'Kandy MOH 1'. The 'Address' field contains 'Dokoda Moligowa, Kandy'. Under 'Change Status', 'Active' is selected. Under 'District', 'kandy' is selected. Under 'Change Vaccine', 'Pfizer' is selected. Under 'Daily Limit', '100' is entered. The 'Vaccine Batch Number' field contains 'PF100-123'. Below this, the 'Vaccination center staff' section lists one staff member: 'Test Staff' (NIC: 993581089v, Contact Number: 0771994147, Status: Active). A 'Save' button is visible at the top right of the info form, and an 'ADD STAFF' button is at the bottom right of the staff list.

Figure 10.61: Edit vaccination centre details

- User can edit any field and click the save button to save users changes.
- The vaccination centre staff list is displayed under the information, and click add staff to add a new staff member.

10.2.2.23 Add new staff member to the vaccination centre

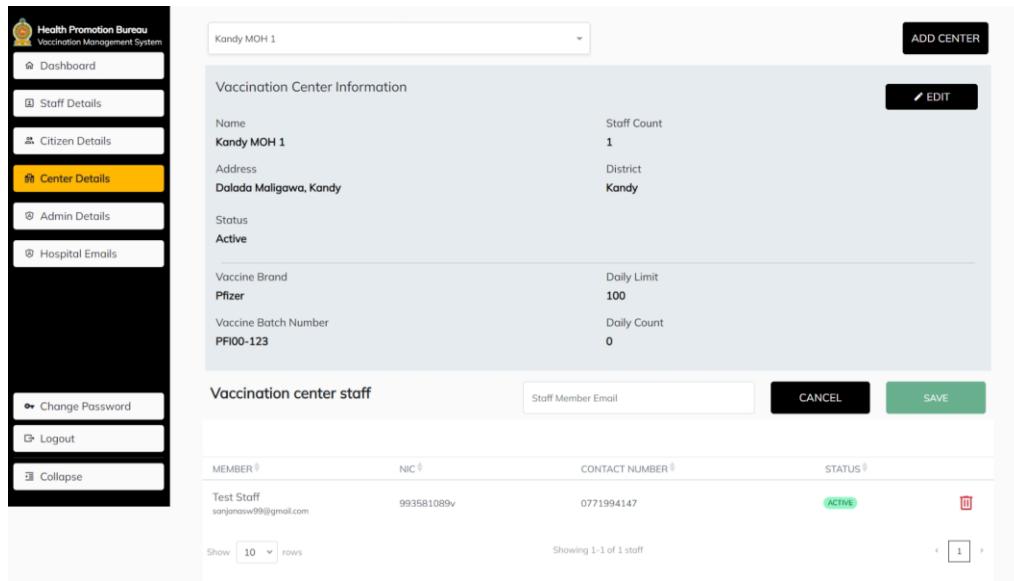


Figure10.62: Add a new staff member

When the user clicks the add staff button, the user can see an input field to enter the email of the new staff member and click the save button.

10.2.2.24 Confirm before deleting a staff member

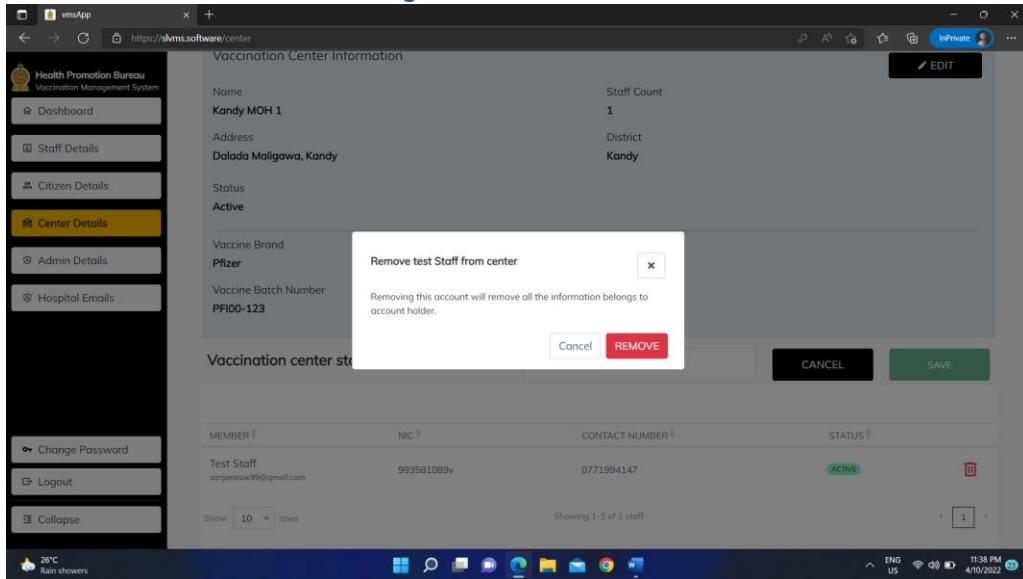


Figure 10.63: Confirm before sending the reset password link

Click the remove button if the user wants to remove the member, and if the user doesn't want to remove it, click the cancel button.

10.2.2.25 Admin details

NAME	NIC	USERNAME	CONTACT NUMBER	STATUS
Vms Admin sanjanasulakshana@thoronage@gmail.com	986281089v	admin	0771994147	ACTIVE

Show 10 rows Showing 1-1 of 1 admin

Figure 10.64: Admin details

Users can search admins by username, email NIC, contact number or status.

If the user wants to add a new admin, click the admin button.

10.2.2.26 Create a new admin

The screenshot shows the 'CREATE ADMIN' form. It includes fields for First Name, Last Name, Email, Username, Gender, NIC, Date of Birth, and Contact number. A 'SEND INVITE' button is at the top right. The sidebar on the left shows 'Admin Details' selected.

First Name *	Last Name *
Enter first name	Enter last name

Email *	Username *
Enter email	Enter username

Gender *	NIC *
Select gender	Enter NIC

Date of Birth *	Contact number *
mmddyyyy	Contact number

Figure 10.65: Create a new admin

When the user clicks the admin button, the user has to fill in all the fields and click the send invite button.

10.2.2.27 Kebab menu of the admin details data row

The screenshot shows the 'Admin' details page with a single record: 'Vms Admin'. The kebab menu (three dots) on the right side of the row provides options for 'View details', 'Edit', and 'Delete'.

NAME	NIC	USERNAME	CONTACT NUMBER	STATUS
Vms Admin sanjanasukuluhonwitharanage@gmail.com	986281089v	admin	0771994147	ACTIVE

Figure 10.66: Admin details

User have to click the kebab menu to get the dropdown with view details and edit and delete options.

10.2.2.28 View admin details

The screenshot shows a web application window titled 'vmApp' with the URL 'https://svms.software/admin'. On the left, there's a sidebar with a tree view containing 'Dashboard', 'Staff Details', 'Citizen Details', 'Center Details', 'Admin Details' (which is selected and highlighted in yellow), and 'Hospital Email'. The main content area is titled 'VIEW ADMIN' and contains the following form fields:

First Name *	Last Name *
vms	admin
Email *	Username *
sanjanasulakshanawithararage@gmail.com	admin
Gender *	NIC *
Male	9866281089v
Date of Birth *	Contact number *
01/02/1950	0771994147

At the bottom left of the content area, there are three buttons: 'Change Password', 'Logout', and 'Collapse'. The status bar at the bottom right shows 'ENG US' and the date '4/10/2022'.

Figure 10.67: View admin details

When the user clicks the view admin, the user can view the details of the admin.

10.2.2.29 Edit admin details

The screenshot shows a 'EDIT ADMIN' modal window. The left sidebar has a dark theme with icons for Dashboard, Staff Details, Citizen Details, Center Details, Admin Details (which is selected), and Hospital Email. The main form contains the following fields:

First Name *	Last Name *
vms	admin
Email *	Username *
sanjanasulakshanawitharage@gmail.com	admin
Gender *	NIC *
Male	986281089v
Date of Birth *	Contact number *
01/02/1950	0771994147

Below the form are 'Reset Password' and 'Change Password' links, followed by 'Logout' and 'Collapse' buttons. The top right of the modal has 'DISCARD' and 'SAVE' buttons. The status bar at the bottom shows weather (26°C Rain showers), system info (ENG US), and time (11:35 PM 4/10/2022).

Figure 10.68: Edit admin details

User can edit all the fields and click the save button on the top of the modal.

If user want to send a link to reset the password of the admin, click the reset password link.

10.2.2.30 Reset the password of the admin

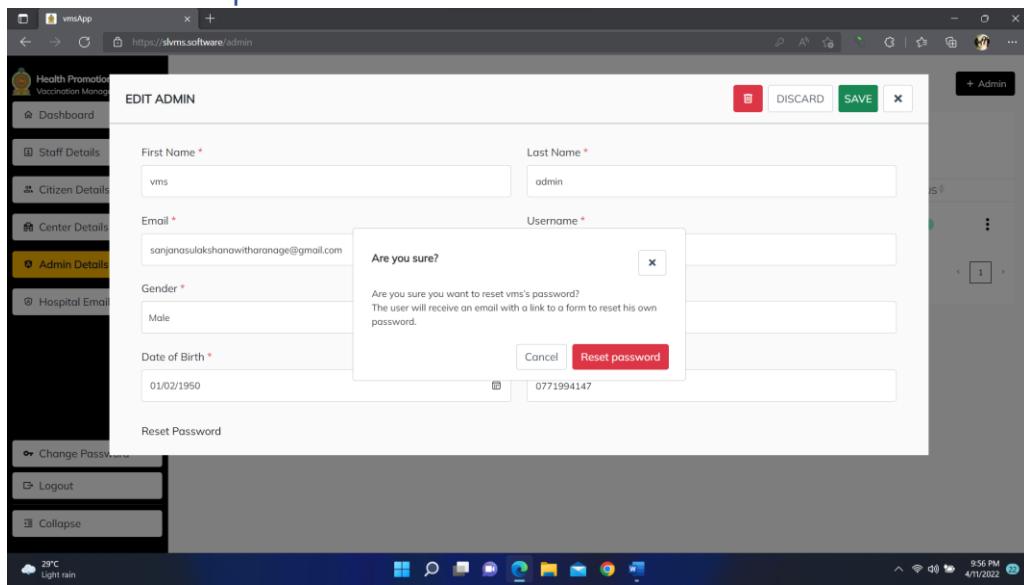


Figure 10.69: Confirm before sending the reset password link

Click the reset password link and confirm here by clicking the reset-password button.

10.2.2.31 Sending reset email successfully

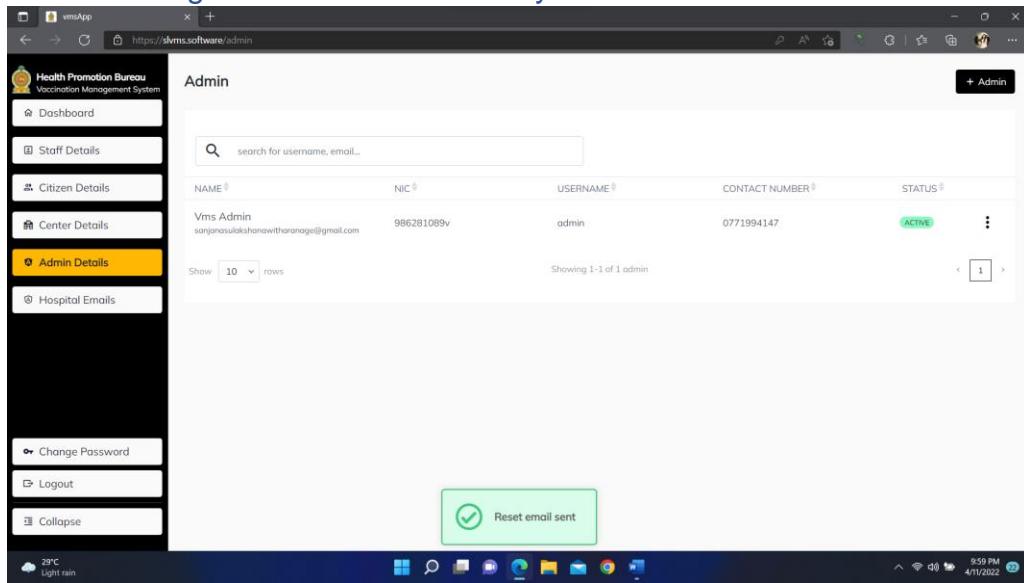


Figure 10.70: Alert for successfully sending reset password link

After sending the email successfully, the success alert will display like this.

10.2.2.32 Confirm before deleting an admin account

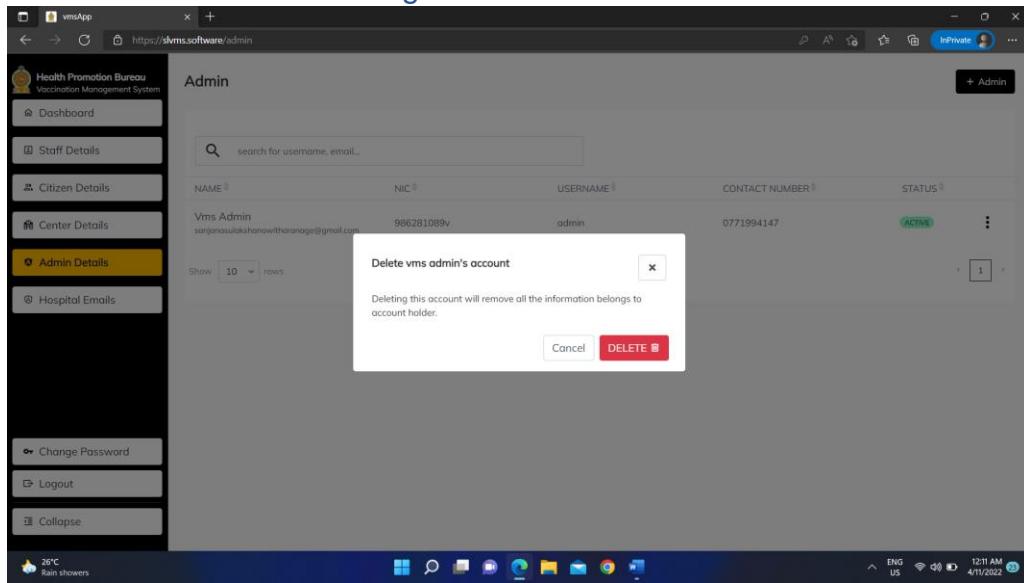


Figure 10.71: Confirm before deleting an admin account

The user must ensure that the user wants to delete the admin account. If no, click the cancel button.

10.2.2.33 Hospital emails

A screenshot of the 'Hospital Emails' section of the VMS Admin interface. The left sidebar has a yellow-highlighted 'Hospital Emails' option. The main area shows a table titled 'Hospital Emails' with columns: DISTRICT and EMAIL. The table lists 10 entries corresponding to districts: Colombo, Gampaha, Kalutara, Kandy, Matale, Nuwara Eliya, Galle, Mora, Hambantota, and Jaffna, each with an associated email address like 'alerts@vmslk.lk'. A search bar at the top allows searching by district or email. A pagination control at the bottom shows 'Showing 1-10 of 50 emails' and a page number '1'.

Figure 10.72: Hospital emails

- Users can search by district or email address of each district.
- When the user clicks the kebab menu, the user will get two options.

10.2.2.34 Kebab menu of hospital emails

The screenshot shows a web application interface for managing hospital emails. On the left is a sidebar with navigation links: Dashboard, Staff Details, Citizen Details, Center Details, Admin Details, and Hospital Emails (which is highlighted). Below the sidebar are buttons for Change Password, Logout, and Collapse. The main area is titled 'Hospital Emails' and contains a table with columns 'DISTRICT' and 'EMAIL'. The table lists the following data:

DISTRICT	EMAIL
Colombo	alerts@vmssl.lk
Gampaha	alerts@vmssl.lk
Kalutara	alerts@vmssl.lk
Kandy	alerts@vmssl.lk
Matale	alerts@vmssl.lk
Nuwara Eliya	alerts@vmssl.lk
Galle	alerts@vmssl.lk
Matara	alerts@vmssl.lk

A context menu is open over the row for Gampaha, showing options 'View details' and 'Edit'. The browser's address bar shows the URL <https://slvms.software/hospital-emails>. The system status bar at the bottom right shows the date as 4/1/2022 and the time as 10:42 PM.

Figure 10.73: Hospital emails

User can click the view details or edit option from the kebab menu.

10.2.2.35 View email address

The screenshot shows a 'VIEW EMAIL' modal window. The sidebar on the left is identical to Figure 10.73. The modal window has a header 'VIEW EMAIL' with 'EDIT' and close buttons. Inside, it displays the following information:

District : Colombo

Email *
alerts@vmssl.lk

The background list of hospitals is partially visible behind the modal.

Figure 10.74: View email address

When user click the view details, it will open this modal and display the district of the hospital and the email address.

10.2.2.36 Edit email address

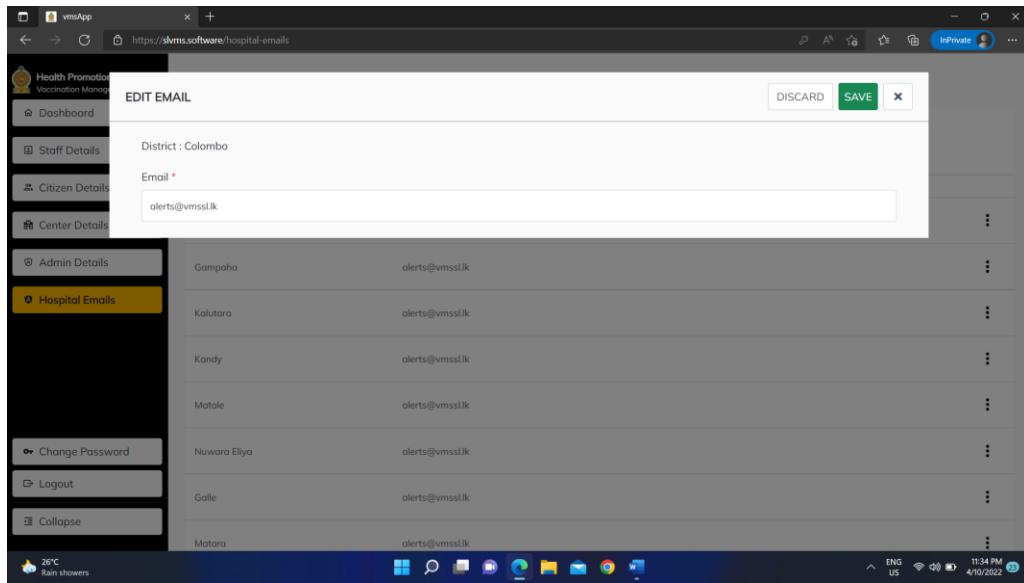


Figure 10.75: Edit email address

Here, user can edit the email address and click the save button.

10.2.2.37 Information Logs

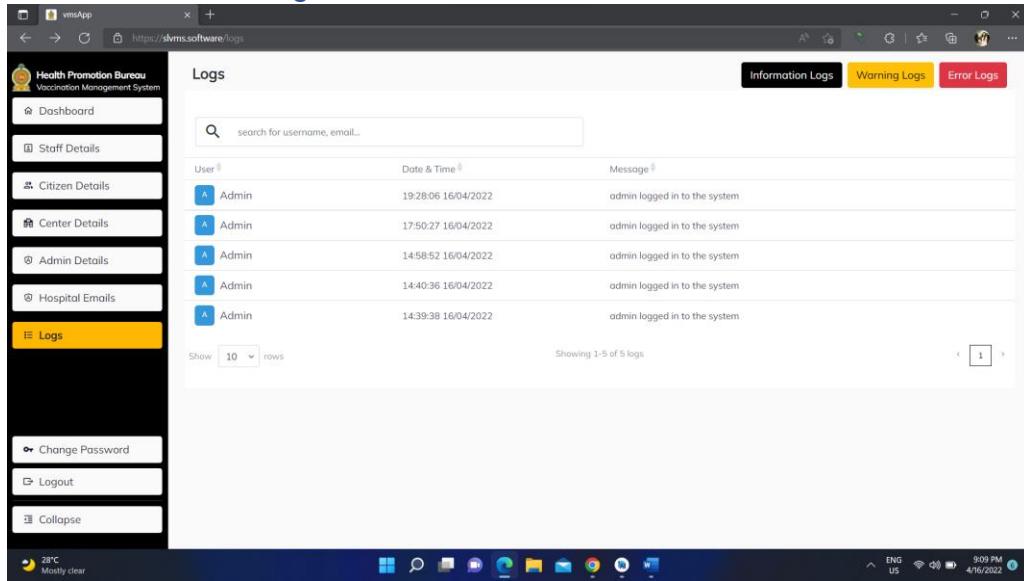


Figure 10.76: Information logs

Information logs are displayed here.

10.2.2.38 Warning Logs

The screenshot shows a web application interface for the 'Health Promotion Bureau Vaccination Management System'. The left sidebar contains navigation links: Dashboard, Staff Details, Citizen Details, Center Details, Admin Details, Hospital Emails, and Logs (which is highlighted). The main content area is titled 'Logs' and has tabs for 'Information Logs', 'Warning Logs' (which is selected), and 'Error Logs'. A search bar at the top of the logs section allows searching by 'User', 'Date & Time', and 'Message'. Below the search bar, there is a table with two entries, both labeled 'Failed':

User	Date & Time	Message
Failed	13:57:58 16/04/2022	Failed to determine the https port for redirect.
Failed	13:57:21 16/04/2022	Failed to determine the https port for redirect.

At the bottom of the logs table, it says 'Showing 1-2 of 2 logs'. The system status bar at the bottom indicates the date and time as 4/16/2022 9:18 PM, and the weather as 28°C Mostly clear.

Figure 10.77: Warning logs

Logins that are failed are displayed here.

10.2.2.39 Error Logs

The screenshot shows the same web application interface as Figure 10.77. The 'Logs' section is displayed, with the 'Error Logs' tab selected. The main content area is titled 'Logs' and has tabs for 'Information Logs', 'Warning Logs', and 'Error Logs' (which is selected). A search bar at the top of the logs section allows searching by 'User', 'Date & Time', and 'Message'. Below the search bar, there is a table with six entries, all labeled 'An unhandled exception has occurred while executing the request.':

Message
An unhandled exception has occurred while executing the request.
An exception occurred while iterating over the results of a query for context type "VMS_API.Models.ApplicationDbContext". "Microsoft.Data.SqlClient.SqlException (0x80131904): A connecti
An error occurred using the connection to database "VMS-API_db" on server "tcp:vms-db-server1.database.windows.net,1433".
An unhandled exception has occurred while executing the request.
An exception occurred while iterating over the results of a query for context type "VMS_API.Models.ApplicationDbContext". "Microsoft.Data.SqlClient.SqlException (0x80131904): A connecti
An error occurred using the connection to database "VMS-API_db" on server "tcp:vms-db-server1.database.windows.net,1433".

At the bottom of the logs table, it says 'Showing 1-6 of 6 logs'. The system status bar at the bottom indicates the date and time as 4/16/2022 9:21 PM, and the weather as 28°C Mostly clear.

Figure 10.78: Error logs

Errors when trying to log in are displayed here, with the exceptions.

10.2.2.40 Change password

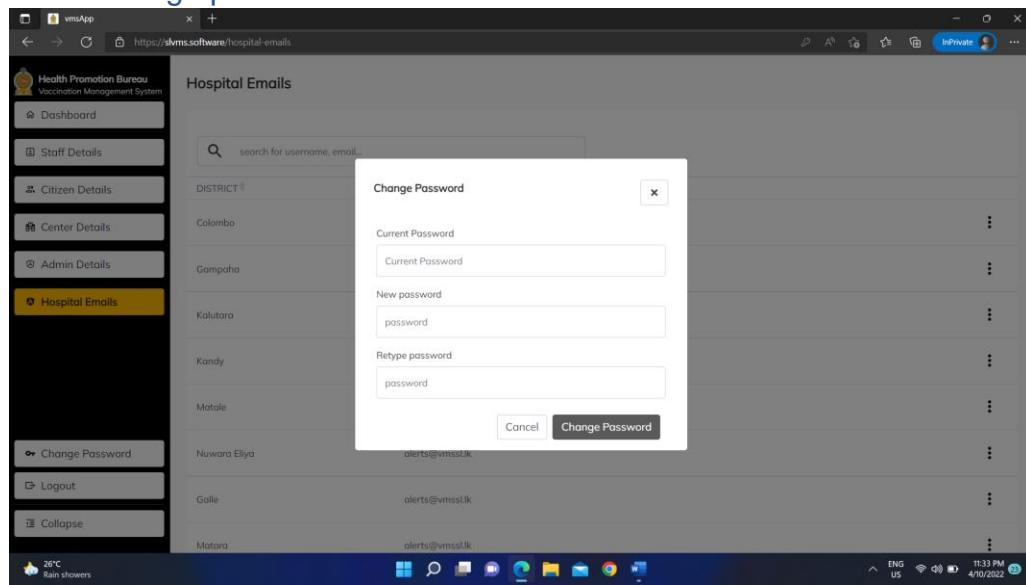


Figure 10.79: Change password

If a user wants to modify the user's password, the user must enter the current password new password and retype the password to confirm and click the button to change the password.

10.2.3 Staff

10.2.3.1 Appointments

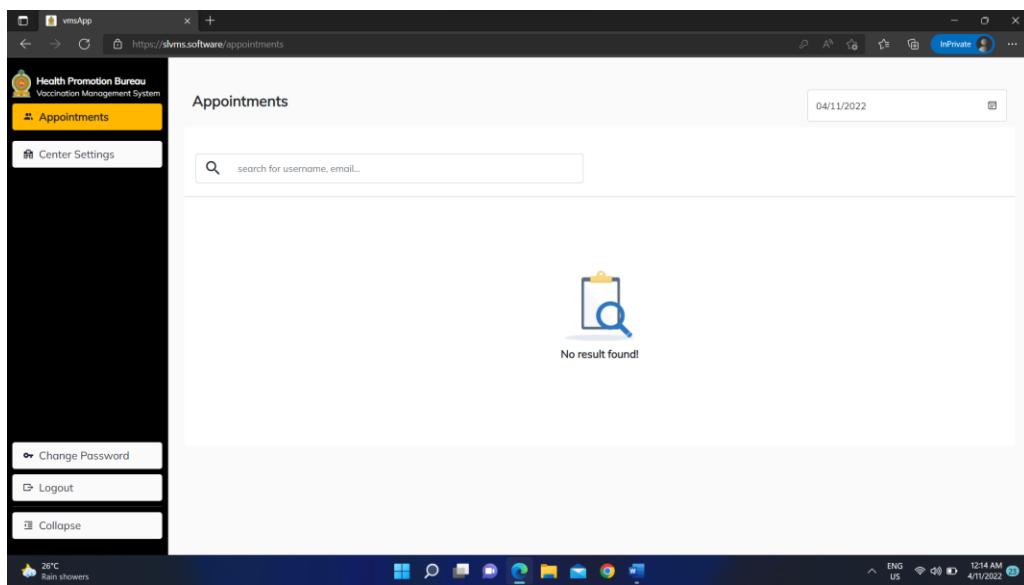


Figure 10.80: Appointments

All the appointments appointed for the hospital which the staff member related to.

User can search by username or email. Also, users can filter by the date.

10.2.3.2 View Appointments

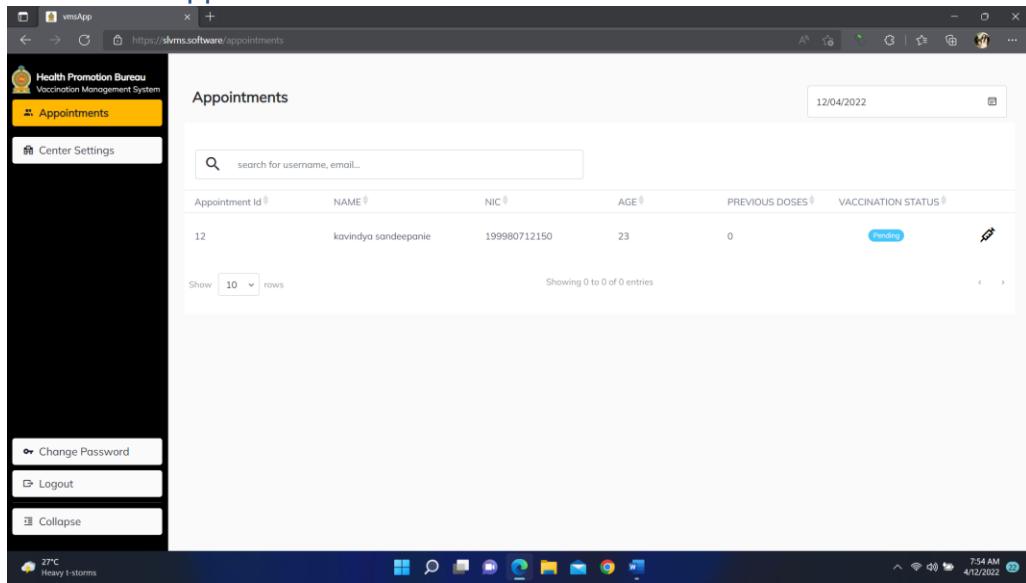


Figure 10.81: View appointments

The appointments are displayed like this.

10.2.3.3 Marking an appointment is completed

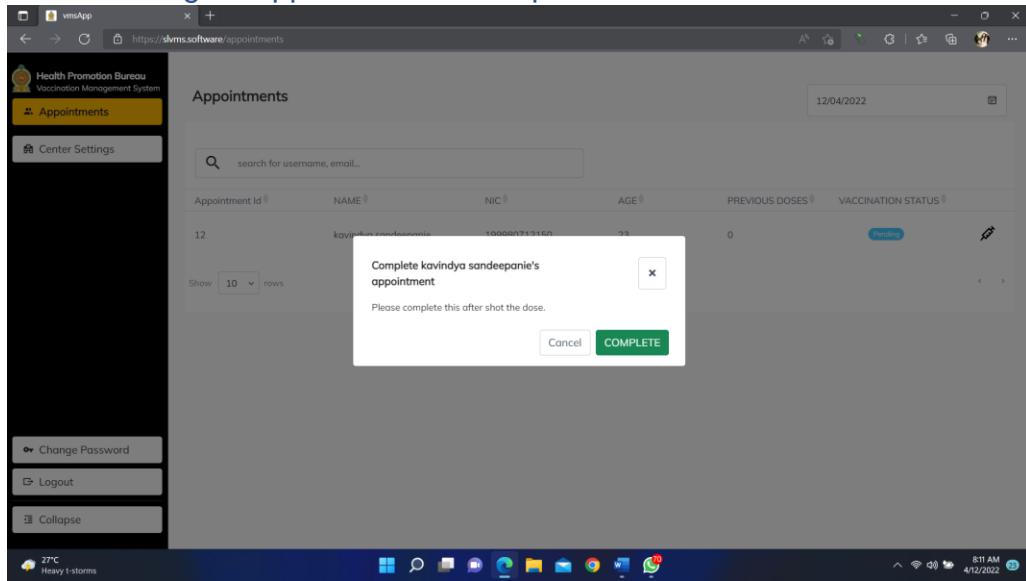


Figure 10.82: Marking an appointment is complete

Click the vaccination icon and click the complete button to mark that the appointment is completed.

Else, click the cancel button.

10.2.3.4 Alert after marking an appointment is done

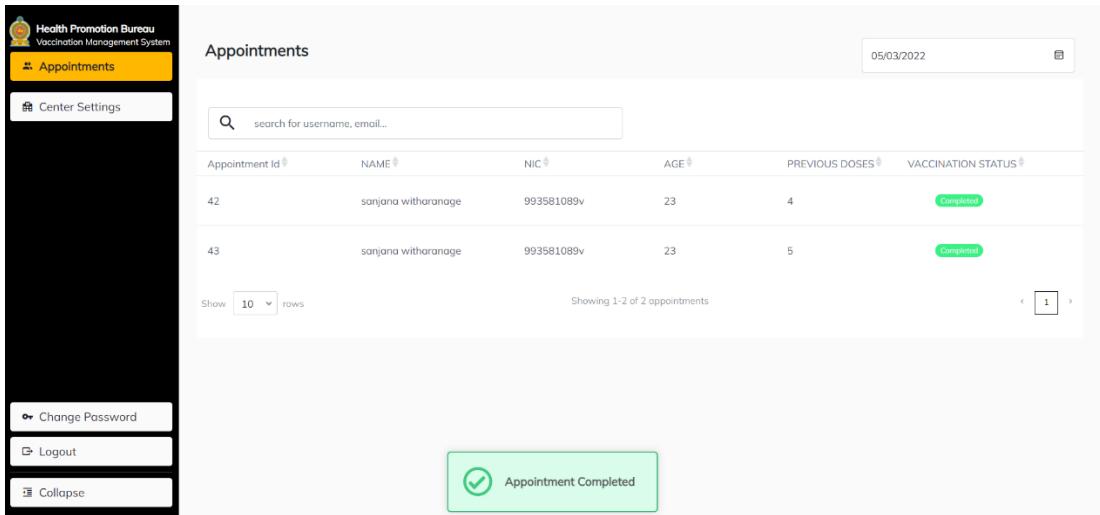


Figure 10.83: Alert after making a completed appointment

The success alert will display like this, and the status also changed to completed.

10.2.3.5 Centre settings

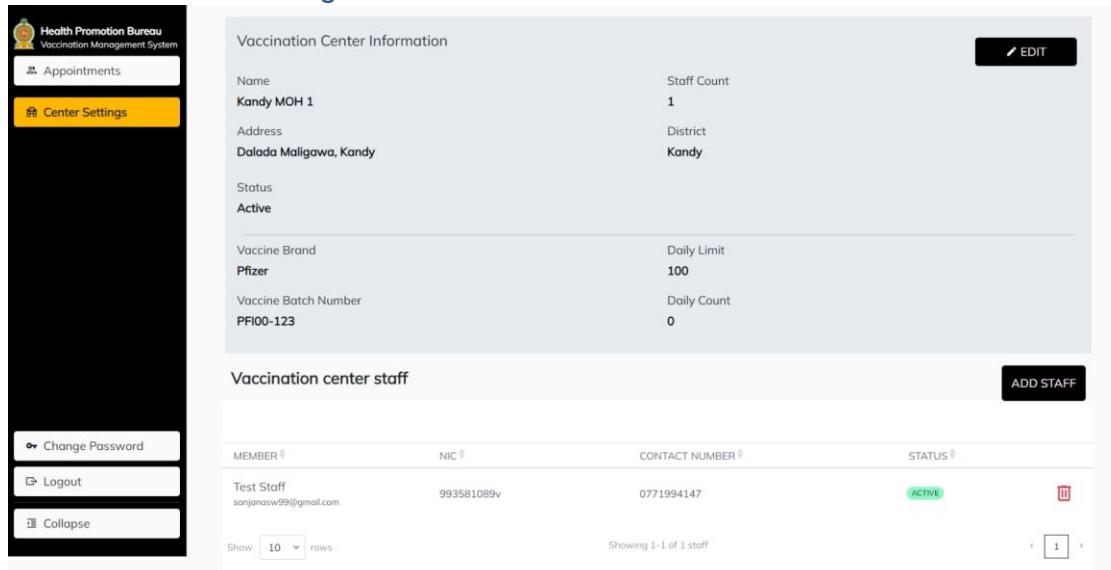


Figure 10.84: Centre settings

When user click the centre settings, user will get the details of the vaccination centre to which user belong and the other Staff members of the vaccination centre. Editing the centre details and adding staff members are also provided.

10.2.3.6 Edit vaccination centre information

The screenshot shows the 'Vaccination Center Information' section with the following fields:

- Name: Kandy MOH 1
- Address: Dalada Maligawa, Kandy
- Change Status: Active
- Change Vaccine: Pfizer
- District: Kandy
- Daily Limit: 100
- Vaccine Batch Number: PF100-123

Below this is a table titled 'Vaccination center staff' showing one staff member:

MEMBER	NIC	CONTACT NUMBER	STATUS
Test Staff sanjanaw99@gmail.com	993581089v	0771994147	ACTIVE

At the bottom, there are buttons for 'ADD STAFF' and 'Save'.

Figure 10.85: Edit vaccination centre details

Here, user can edit all the details of the vaccination centre and click the save button to save the changes user made.

10.2.3.7 Add new staff members to the vaccination centre

The screenshot shows the 'Vaccination Center Information' section with the following fields:

- Name: Kandy MOH 1
- Address: Dalada Maligawa, Kandy
- Status: Active
- Vaccine Brand: Pfizer
- Daily Limit: 100
- Vaccine Batch Number: PF100-123
- Staff Count: 1
- District: Kandy

Below this is a table titled 'Vaccination center staff' showing one staff member:

MEMBER	NIC	CONTACT NUMBER	STATUS
Test Staff sanjanaw99@gmail.com	993581089v	0771994147	ACTIVE

At the top right of the main section is an 'EDIT' button. At the bottom, there are buttons for 'Staff Member Email', 'CANCEL', and 'SAVE'.

Figure 10.86: Add a new staff member

When user click the add staff button, user can enter the user's email address and click the button 'save' to add the user as a staff member. Also, user can remove the staff members by clicking the delete button at the corner of the data row.

10.2.3.8 Confirm before removing a staff member

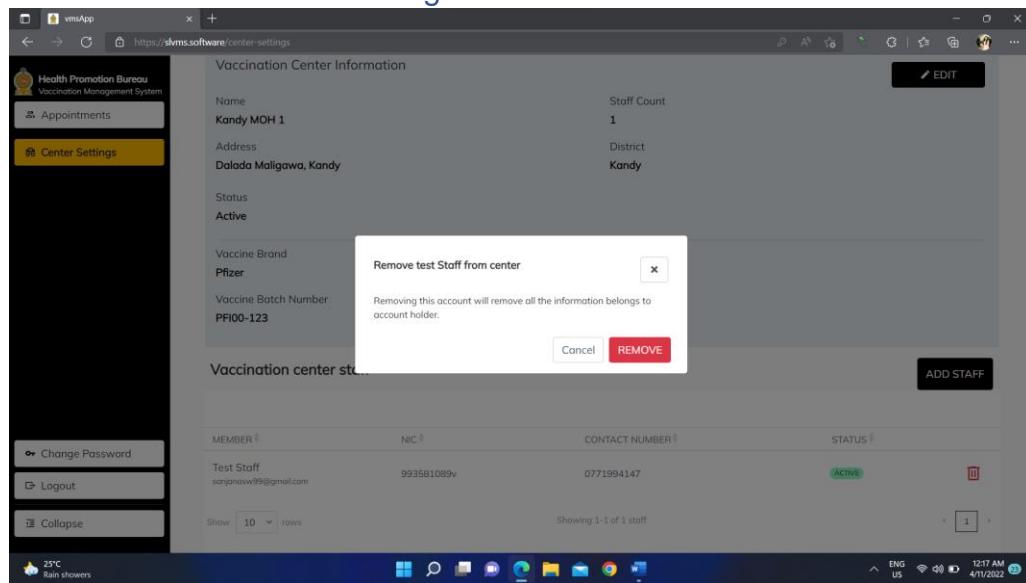


Figure 10.87: Confirm before removing a staff member

User have to confirm whether user need to remove this member from the vaccination centre and from the system.

11 Challengers/Issues and Resolving

1. Git conflicts

Previously, we have used flocked repositories for development. Then we had to resolve conflicts repeatedly, and duplicate case files changed several times during development.

We moved to feature branches to overcome that instead of using flocked repositories. We maintained a proper flow of git to minimize code conflicts.

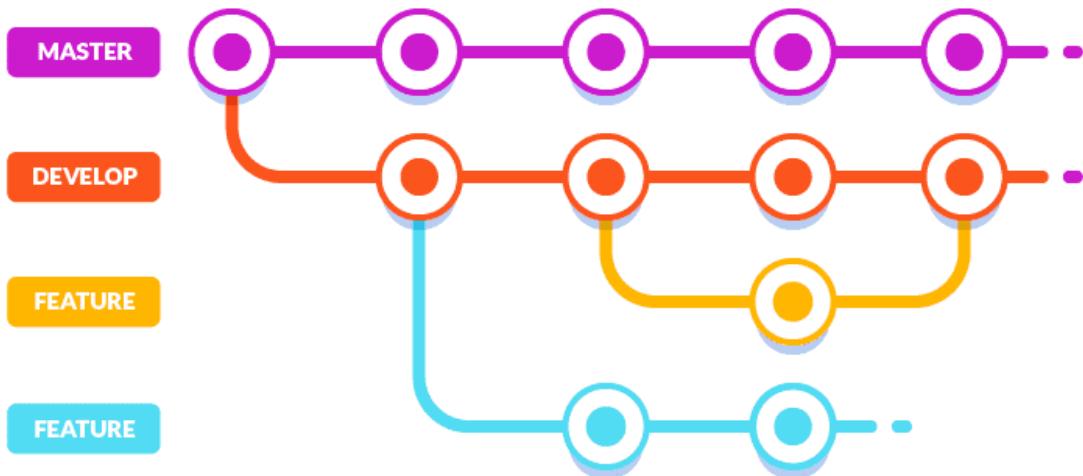


Figure 11.1: GIT flow

2. Database syncing issues

When we started the development with our laptops without hosting the system on the cloud, there was an issue with syncing data in different local Databases into a single database. To overcome that, we hosted our application in Azure app service and SQL server from the initial step and configured CI/CD pipelines with GitHub action to do quick deployments. And configured Discord notification channel to notify front-end developers about releases.

The screenshot shows the GitHub Actions pipeline for the 'Predictly-Api' repository. The 'All workflows' tab is selected, displaying 28 workflow runs. The runs are listed in descending order of age, starting from the most recent at the top. Each run is represented by a card showing the workflow name, a green checkmark icon indicating success, and details such as the event (e.g., 'Build and deploy ASP.NET Core app to Azure Web App'), status (e.g., 'Build and deploy ASP.NET Core app to Azure Web App - Predictly #23'), branch ('master'), and the time it was run (e.g., '6 days ago').

Figure 11.2: Pipeline releases

The screenshot shows a Discord channel named '#github-updates'. The channel interface includes a sidebar with various project categories like 'personal-details', 'GENERAL', 'voice', 'DISTRIBUTED SYSTEMS', 'resources', 'links-research', and '# github-updates'. The main window displays a series of messages from GitHub. The first message is a general update: '[vms-client] GitHub Actions checks cancelled on development'. Subsequent messages are specific to the 'vms-client' repository: '[vms-client] Build and Deploy Job success on master', '[vms-client] GitHub Actions checks success on master', and '[vms-client] 1 new commit' (with a link to a commit message). On April 12, 2022, there is a message from GitHub '[vms-api:development] 1 new commit' with a link to a commit message. On April 13, 2022, another message from GitHub '[vms-api:development] 1 new commit' is shown. On April 16, 2022, a third message from GitHub '[vms-api:development] 1 new commit' is shown. The right side of the screen shows a list of online users: Sanjana (ONLINE), Kavindya (OFFLINE), Krishan (OFFLINE), and Manaja (OFFLINE).

Figure 11.3: Discord notifications

12 Future Implementations

- SMS – Twilio messaging service / AWS messaging service.
- For Foreigners using passport numbers instead of NIC.
- The app has a lot more potential to provide to the public in the future. The application can be built to provide each citizen's PCR history.

13 Conclusion

- The critical characteristics of a distributed system are fault tolerance, concurrency, scalability, and transparency.
- The ACID qualities are critical for a system's longevity, as they allow it to run reliably with no downtime or errors.
- Project management abilities appeared realistic and valuable to keep up with the time frame when it was difficult to get a team together to handle projects.

14 Group Contribution

ID Number	Name	Contribution
21003856	S S Witharanage	Equal Contribution
21004588	M C Mallikarachchi	Equal Contribution
23015875	K Shanuka	Equal Contribution
21004343	A D K Sandeepanie	Equal Contribution

Table 1: Contribution Table

References

- asp.net web api 5.0 (no date). Available at: <https://www.c-sharpcorner.com/article/asp-net-core-5-0-web-api/#:~:text=In%20simple%20words%2C%20we%20can,a%20cross%2Dplatform%20web%20API>. (Accessed: February 26, 2022).
- Flutter-wikipedia (no date).
- <https://azure.microsoft.com/en-in/overview/what-is-middleware/> (no date). Available at: <https://azure.microsoft.com/en-in/overview/what-is-middleware/> (Accessed: March 7, 2022).
- <https://code.visualstudio.com/> (no date). Available at: <https://code.visualstudio.com/> (Accessed: March 30, 2022).
- <https://developer.android.com/studio/intro> (no date). Available at: <https://developer.android.com/studio/intro> (Accessed: March 30, 2022).
- <https://www.redhat.com/en/topics/middleware/what-is-middleware> (no date). Available at: <https://www.redhat.com/en/topics/middleware/what-is-middleware> (Accessed: March 7, 2022).
- ben Kopf (no date) <https://www.toptal.com/designers/ui/figma-design-tool>. Available at: <https://www.toptal.com/designers/ui/figma-design-tool> (Accessed: March 30, 2022).
- Web APIs - W3Schools. https://www.w3schools.com/js/js_api_intro.asp (no date). Available at: Web APIs - W3Schools. https://www.w3schools.com/js/js_api_intro.asp (Accessed: February 27, 2022).
- WHO (2021). Achieving 70% COVID-19 Immunization Coverage by Mid-2022. [online] www.who.int. Available at: <https://www.who.int/news/item/23-12-2021-achieving-70-covid-19-immunization-coverage-by-mid-2022>.

Appendix

14.1 Project Proposal

1. Problem statement

Covid has become a part of our world, and vaccines have also become part of us. But in most countries, there's no proper management or better way to track or prove a way that we're vaccinated. For instance, let's take Sri Lanka. In Sri Lanka, only a single document is given to prove whether a person is vaccinated. The standalone document is not adequate for international travel. Most countries require a presentable covid certificate that includes a readable QR code published by the Sri-Lankan government (gov.uk, n.d.). This arises certain complications for the citizens of Sri - Lanka, as the people have to manually apply for the "Smart Certificate" offered by the government, which contains a readable QR that would take up to 21 days, and the "Smart Certificate" is issued only for citizens leaving Sri-Lanka. To obtain the competent certificate, one has to upload their personal information, vaccination certificate, and documents proving the departure from Sri Lanka (news.lk, 2021).

This won't be fair because only a minority of the population would have a certificate approved for international travel or any purpose. Having an app that can automatically generate a covid certificate with an approved QR code reduces some significant issues for the people of Sri Lanka. One of the issues that are addressed is

- Provide authentic vaccination certificate with QR code.
- Reduces the hassle caused at the airport.
- Easy access to the certificate through the mobile app.

And to add more, the majority of Sri Lanka people don't have the proper knowledge on where to book an appointment or where the nearest vaccination centre is located. To solve all these issues together and better serve the citizens of Sri Lanka for hassles travel and to provide an authentic covid certificate with a readable QR code document for everyone, VMS or Vaccine Management System comes to play.

2. Project Objective

VMS or Vaccine Management System is mainly a Mobile application with a website for the admins/staff and redirection pages. VMS promises to deliver authentic proof that you're vaccinated directly from the government vaccination database in the form of an inbuilt app QR code. The app is designed to show all the user details and the vaccination details, including from what country and place the person got the jab, vaccine type, batch, etc. All these details are taken directly from the "demo government vaccination database".

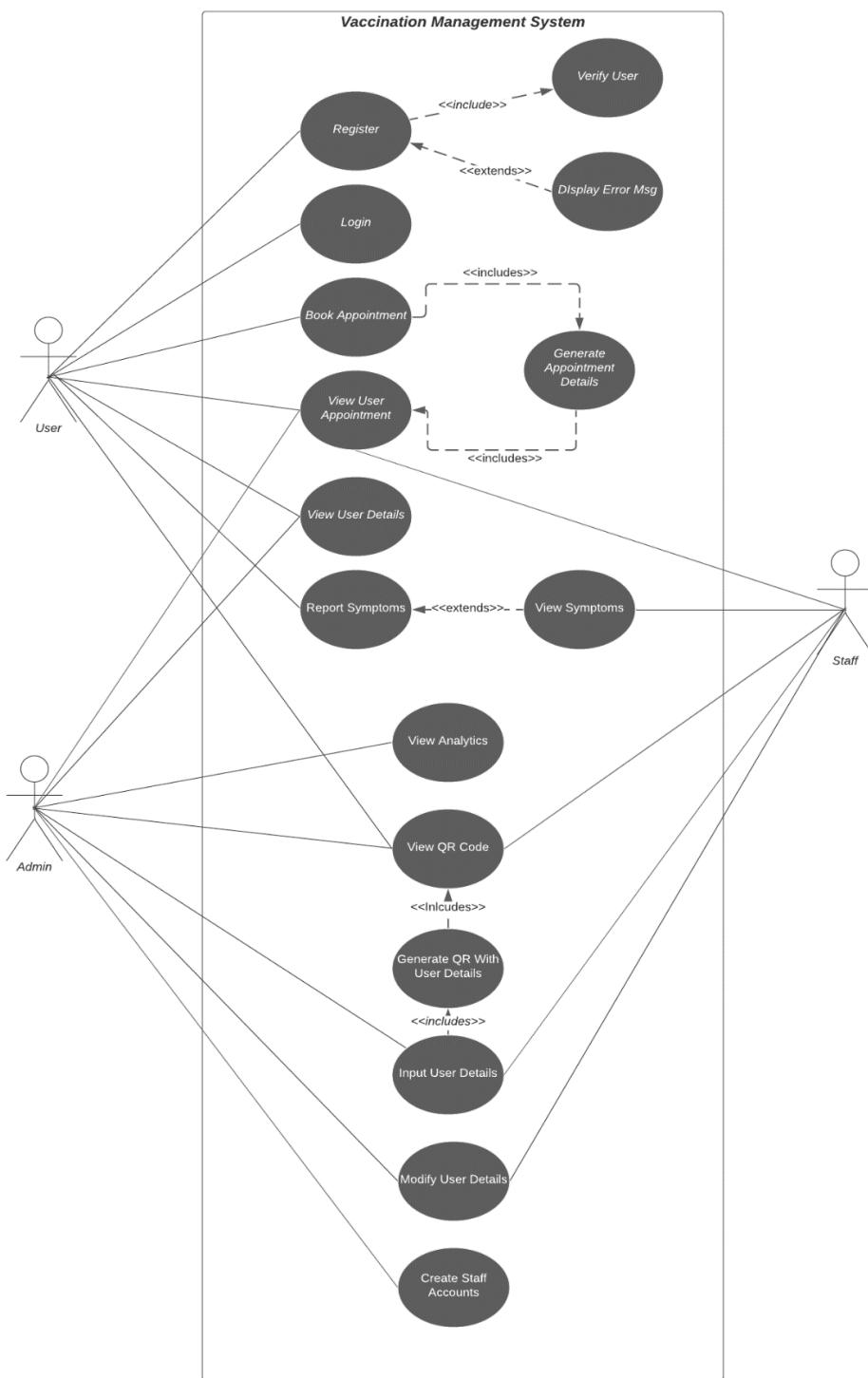
The application promises to help the Sri Lankan citizen's by

- Eliminating the need to carry the physical covid document.
- Provide an authentic document with a QR code directly from the "Government vaccination database".
- Give citizens the to travel abroad with ease.
- No need of applying for a "smart covid certificate" (news.lk, 2021).
- Reduces the hassle caused at the airport.
- Provide accurate information on covid vaccination centres.
- An easier way to verify whether a citizen is vaccinated.
- Easy access to the certificate through the mobile app.

Additionally, the app allows the user to book an appointment at the nearest vaccination centre effortlessly. When the QR code is scanned by our inbuild or any QR scanner, the link to the "demo government" vaccination website is presented.

This link redirects the user to the complete user details along with the full vaccination details. When it comes to the Staff and admins, they are provided with a web application that displays a dashboard based on their authorization level. There are mainly three different types of individuals staff, admin, citizen. And each has different kinds of interaction with the application.

Figure
4:



Usecase Diagram

3. Target Users

VMS is built in a way where the entire population of Sri Lanka can use the application.

Each and every citizen can log in directly using their NIC or register an account.

The Citizens' vaccination information is managed by the Staff and the admin with proper privacy rights.

4. Application Features and Descriptions

Vaccination Management System has three different user categories that are Staff, Admin, and Citizen. All three of these users have different kinds of features unique to them. The features VMS would consist of are

4.1 Citizen

- Onboarding: citizens are briefed on how to use the application.
- Citizen Login & Registration: Citizens can create an account and log in to use the application.
- QR code: Every citizen will have an authentic QR code showing the status of vaccination along with personal details.
- Vaccine Center Locator: Citizens can locate the nearest vaccination centre.
- Vaccine Booking: Citizens can book the preferred vaccine at the nearest vaccination centre.
- Bookings: Show upcoming appointments.
- QR scanner: Inbuilt QR scanner is provided.
- General Information: Provide prerequisites to the citizens before coming to a vaccination centre.
- Symptoms: Citizens can report signs of covid or side effects from the vaccine.

4.2 Staff

- Staff Login: Staff has an ID and password created for them, which they can log in with.
- Staff Dashboard: Staff dashboard will have a place to enter a citizen's NIC to check if he is eligible for the vaccine.
- Vaccination Details: If the citizen is eligible for the vaccination, The citizens' information is displayed to the Staff.
- Vaccination Status Update: Staff can update the vaccination status of the citizen

4.3 Admin

1. Admin Login: Admin can log in with his password and username.
2. Admin Dashboard: Admin dashboard is greeted with the choice of editing staff details or citizens.
3. Staff Update: Admin can add, delete, edit staff information.
4. Citizen Update: Admin can add, delete, edit Citizen information.

5. Application Technologies and Justification

Vaccination Management System is a complicated system with a mobile application for the citizens and a web application for the Staff and admins. Having two different types of platforms require various types of technology. The list of technologies is broken down into three categories

1. Frontend.
2. Backend.
3. Project management.

5.1 Frontend

Frontend development means creating a web application or mobile application graphical user interface to interact with them directly. The technologies used to code VMS front end are

- Angular.
- Flutter.

5.1.1 Angular

Our web application graphical user interaction is coded with the angular framework because

Angular is an open-source framework and is made by google. The angular framework is based on JavaScript and TypeScript, which allows to implementation of many core functionalities. The angular framework satisfies the user experience and provides good performance (Natalia Juhasz, 2020). VMS web application is mainly built on angular because of these core functionalities

- Not complicated to learn: Angular framework looks imitating, but it's relatively easy to understand the framework.
- Data binding: Data binding in the Angular framework is automatic and synchronized between the two model and view components. Because Angular handles data binding, developers can use the model as one source of truth in the project. Thus, when the model changes, it changes the look of the view and vice versa (AngularJS, n.d.).
- Incremental Faster Document Object Model: DOM is an API that provides logical documents and how they're accessed. The main goal is to increase performance.
- MVVM Architecture: Angular follows MVC, but it's more of an MVVM architecture because angular doesn't need the developers to split the code and later unite them. MVVM supports data binding between the View and View Model.

5.1.2 Flutter

Flutter is the technology that is used for the mobile application development for VMS. Flutter was only released a few years ago by Google, but it had already won the technology market. Its capability for developing apps natively and cross-platform have piqued interest in many companies and becoming more popular. Here are some of the reasons why flutter is the technology that is used for VMS

- Cross-platform: When using a native framework like Kotlin or Swift, the code written for an application in android can't be run on IOS. Companies pour more capital into developing two different native applications. But flutter doesn't require any of that as the code is written for android and be run on IOS and vice-versa. This has caused flutter to gain a lot of attention.
- Support & Community: Flutter has a large community supporting the framework. A large gathering is essential when choosing a framework because debugging and learning become easier (Cherednichenko, 2020).
- Performance: Performance is a huge factor as everyone wants the application to run smoothly and fast. Flutter framework is one of the best performing languages. Flutter even outperforms the native framework when tested (Cherednichenko, 2020).
- Code Structure: Flutter code structure is similar to native code structure because the google developers were inspired by it. The Google flutter team has given the best features of native languages to flutter.

5.2 Backend

Backend Frameworks are libraries that help to configure the server-side of a web application or mobile application. VMS is coded with two different frameworks

- ASP.net Core
- Microsoft SQL Server

5.2.1 ASP.Net Core

ASP.NET CORE is one of the leading technologies in the Information & Technology industry. Today, the high-performance, cross-platform framework is frequently utilized to develop cloud-based and modern apps. VMS uses ASP.NET CORE cause of some core functions provided by .net, and these are

- Cross-Platform: .Net Core framework is used for creating cross-platform applications for Windows, macOS, Linux. This reduces the code written and increases time efficiency.
- Microsoft Support: .Net Core is a product of Microsoft. It supports Windows, macOS, Linux. It is updated multiple times a year to keep up with modern technologies. Moreover, .Net is tested and managed by Microsoft.
- Web API Support: ASP.Net Core allows the creation of Web Application Programming Interface effortlessly.
- Open Source: The MIT and Apache 2 licenses are used by .NET Core, which is open source. It is a part of the .NET Foundation, an independent non-profit that supports the .NET ecosystem, innovative, commercial, and open-source friendly. Over 25,000 developers are now contributing to The .NET open-source code base from over 1700 companies outside of Microsoft. The .NET community is rapidly expanding, and there are currently a lot of free community projects and libraries. Aside from the community and Microsoft, members of the Technical Steering Group such as Google, JetBrains, Red Hat, Samsung, and Unity are directing the .NET platform's development (Rajarathinavel, 2019).

5.3 Project Management

Planning and time management play a crucial role in the success of a project. The technologies used for planning and managing VMS are

- Figma
- Jira

5.3.1 Figma

Figma is a web-based development tool used for creating a graphical user interface for websites or mobile applications. Figma is useful for creating a website prototype or mobile application developers have in mind based upon the client's requirements.

The work is transparent because all the members can see the things added newly and the items removed. Comparingly to the other design platforms, Figma is a collaborative, borderless, transparent, community-driven and open-sourced software.

5.3.2 Jira

Jira is a project management and software bug tracking web-based application. Jira software was developed by Atlassian cooperation Plc in Australia, which has become widely used for agile development and for teams to track bugs, stories, epics, and other activities in a project. Jira Software includes scrum and kanban boards for teams using agile approaches. Tasks are connected to customized workflows on boards, which are task management centres. Boards provide visibility into the status of each work item and provide transparency across teamwork. (Atlassian, 2021)

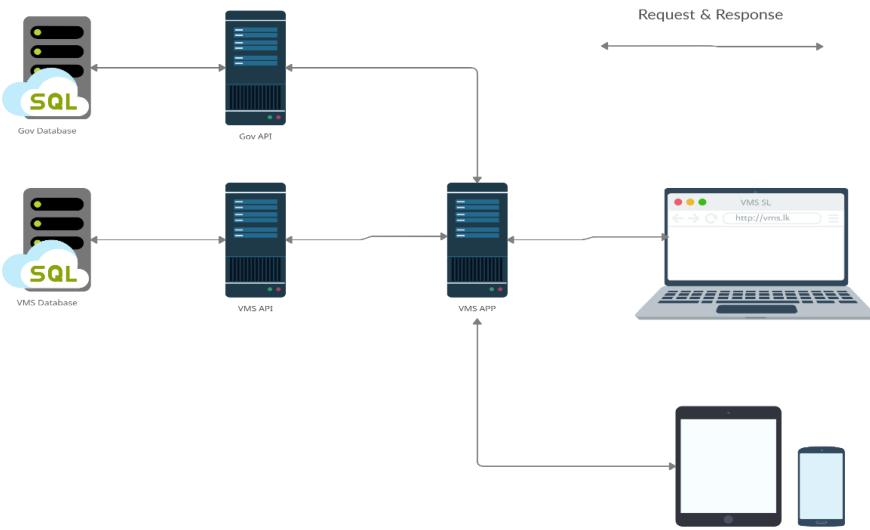


Figure 5: Project Explanation Diagram

6. Time frame

Time management is crucial as it keeps the project within the given deadlines. The roadmap given below helps the team to manage the given deadlines for each person

