



COMSATS University Islamabad (CUI)

Project Name
Child Immunization and Growth Tracking
System

By

Student Name 1 CIIT/SP18-BSC-151/ISB

Student Name 2 CIIT/SP18-BSC-164/ISB

Supervisor

Dr. Ashfaq Farooqi

Bachelor of Science in Computer Science (2018-2022)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.



COMSATS University Islamabad (CUI)

Projechke
Child Immunization and Growth Tracking
System

A project presented to
COMSATS University Islamabad

In partial fulfillment
of the requirement for the degree of

Bachelor of Science in Computer Science (2018-2022)

By

Student Name 1 CIIT/SP18-BSC-151/ISB

Student Name 2 CIIT/SP18-BSC-164/ISB

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Student Name1

Shariq Ahmed

Student Name2

Umar Khalid

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (CS) “Project title” was developed by **STUDENT 1 NAME (CIIT/FAXX-BSE-000)** and **STUDENT 2 NAME (CIIT/FAXX-BSE-000)** under the supervision of “SUPERVISOR NAME” and co supervisor “CO-SUPERVISOR NAME” and that in (their/his/her) opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Sciences.

Dr. Ashfaq Farooqi

Supervisor

External Examiner

Head of Department

(Department of Computer Science)

Executive Summary

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Dr. Majid Iqbal Khan” and our Co-Supervisor “Mr. Mukhtar Azeem”. Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are deeply indebted to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

Student Name1

Shariq Ahmed

Student Name2

Umar Khalid

Abbreviations

SRS	Software Requirement Specification
PC	Personal Computer
SDD	Software Design Description
ML	Machine Learning
MERN	Mongo DB, Express.js, React, Node.js
JS	JavaScript
SDLC	Software Development Lifecycle

Table of Contents

1	Introduction	1
1.1	Brief Overview	1
1.2	Relevance to Course Modules	1
1.3	Project Background	1
1.4	Literature Review	1
1.5	Analysis from Literature Review	1
1.6	Methodology and Software Lifecycle for this Project	1
1.6.1	Rationale behind the Selected Methodology	1
2	Problem Definition	2
2.1	Problem Statement	2
2.2	Deliverables and Development Requirements	2
2.3	Current Systems	2
3	Requirement Analysis	4
3.1	Use Cases Diagram(s)	4
3.2	Detailed Use Cases	4
3.3	Functional Requirements	4
3.4	Non-Functional Requirements	4
4	Design and Architecture	5
4.1	System Architecture	5
4.2	Data Representation	5
4.3	Process Flow/Representation	5
4.4	Design Models	5
5	Implementation	7
5.1	Algorithm	7
5.2	External APIs	7
5.3	User Interface	7
6	Testing and Evaluation	8
6.1	Manual Testing	8
6.1.1	Unit testing	8
6.1.2	Integration Testing	9
6.1.3	Functional Testing	10
6.1.4	System Testing	10
6.2	Automated Testing	10
7	Conclusion and Future Work	cxv
7.1	Conclusion	cxv
7.2	Future Work	cxv
8	References	cxvi

List of Figures

Fig 1.1 Block Diagram.....1

Fig 2.1 Use Case Diagram2

1 Introduction

Pakistan is an underdeveloped country with numerous problems but the top of them is not having basic health facilities. There is a lack of hospitals, health centers, education, and responsibility. Pakistan is one of those countries which has the highest infant and child mortality rate. Half of them are due to vaccine-preventable diseases. In 2019 mortality rate of children was 62 out of 1000. These numbers are reducing with time. In 1970 the mortality rate was 190 out of 1000. But still, we are unable to stop these deaths. Because we don't have a proper system to monitor all the health-related issues. Pakistan has started its Expanded Program on immunization (EPI) in 1978 but its result is not satisfactory. One of the major things is that we are one of the two countries where polio is not eradicated. Another is Afghanistan.

The government spends millions on polio and other vaccination campaigns but results are not much fruitful. The main problem which we have noted is that our vaccination system is not tracking infants from their birth. We have a manual system which consists of a book. Another problem which we noted during our research is the lack of education in people. Many people think especially in the rural and backward areas where there is a lack of education that these vaccines are not good for their children. Many religious extremists believe that these vaccines will create infertility problems in the children so they don't take their children to vaccination centers and even if the polio workers come, they lie to them, and in extreme cases especially in KPK, some polio workers got killed too for this reason.

Apart from the vaccination, another reason for the high mortality rate among children is lack of nutrition among them. There is no proper monitoring of the child's growth which leads them towards having diseases or in many cases these infants or children die too.

1.1 Brief Overview

We are proposing a model Child immunization and growth tracking which covers both of the above-mentioned problems. This model will try to track the children and will tell the authorities about the infants and their parents' details so the authorities can track them. This system will add all the details about infants, notify them before their vaccination dates, mark their presence or absence, and report of the absentees. This will be going to help the authorities to track the children and their parents so if anyone tries to escape they will be tracked easily. It will also predict which area could have more children with these diseases. Stats on a yearly and monthly basis will be generated to see how many children got the vaccine and how many were left. Child growth and vaccination tracking will also be done by the system. Through strong database, this system will be able to track all the vaccination details. With the help of machine learning, parents will be able to check growth of their child time to time.

1.2 Relevance to Course Modules

Below are the Concepts that we applied while implementing this project:

1.2.1 Machine Learning:

Machine learning is a type of artificial intelligence which uses the historical data to predict new results. It allows applications to become more accurate with the passage of time.

1.2.2 Databases

Collection of data in an organized structure from which data can be easily accessed and manipulated according to our own desire. We can store large amount of that with the minimal risk of losing the records. Apart from managing data, databases also provide privacy of the data too

1.2.3 Mobile Development

In this concept we learned how to make an application for mobile devices. As the technology is revolving and today's time requirement is to make an application that works on both famous operating system that is Android and IOS.

1.2.4 Website Development

A concept where we implement our all programming skills to make a program that is used by the end user comfortably. These are built on the requirements of the stakeholders. To make it easily interactive, front-end is made user friendly, while for optimize performance, its backend is coded in optimal way to avoid errors.

1.3 Project Background

Pakistan is an underdeveloped country with numerous problems but the top of them is not having basic health facilities. There is a lack of hospitals, health centers, education, and responsibility. Pakistan is one of those countries which has the highest infant and child mortality rate. Half of them are due to vaccine-preventable diseases. In 2019 mortality rate of children was 62 out of 1000. These numbers are reducing with time. In 1970 the mortality rate was 190 out of 1000. But still, we are unable to stop these deaths. Because we don't have a proper system to monitor all the health-related issues. Pakistan has started its Expanded Program on immunization (EPI) in 1978 but its result is not satisfactory. One of the major things is that we are one of the two countries where polio is not eradicated. Another is Afghanistan.

The government spends millions on polio and other vaccination campaigns but results are not much fruitful. The main problem which we have noted is that our vaccination system is not tracking infants from their birth. We have a manual system which consists of a book. Another

problem which we noted during our research is the lack of education in people. Many people think especially in the rural and backward areas where there is a lack of education that these vaccines are not good for their children. Many religious extremists believe that these vaccines will create infertility problems in the children so they don't take their children to vaccination centers and even if the polio workers come, they lie to them, and in extreme cases especially in KPK, some polio workers got killed too for this reason.

Apart from the vaccination, another reason for the high mortality rate among children is lack of nutrition among them. There is no proper monitoring of the child's growth which leads them towards having diseases or in many cases these infants or children die too.

1.4 Literature Review

In 2019 mortality rate of children was 62 out of 1000. These numbers are reducing with time. In 1970 the mortality rate was 190 out of 1000. But still, we are unable to stop these deaths. Because we don't have a proper system to monitor all the health-related issues. Pakistan has started its Expanded Program on immunization (EPI) in 1978 but its result is not satisfactory.

1.5 Analysis from Literature Review

Application Name	Weakness	Proposed Project Solution
Immunization information system	<ul style="list-style-type: none"> Limited to only tracking of vaccines Not keeping track of individuals Cannot predict about future vaccine requirement Only US-based Only web-based application 	<ul style="list-style-type: none"> The system will be able to track individuals too It will keep all the record of vaccine and child too It will be able to predict the future requirement of the vaccine in a specific area The system will be able to generate all the reports which can help authorities The system will be web and mobile-based The system will also keep track of polio workers
Immunization exercise management system	<ul style="list-style-type: none"> Only web-based application Limited to only track the vaccine record 	<ul style="list-style-type: none"> Track both child and vaccination record Track the child growth Mobile and web-based application

	<ul style="list-style-type: none"> • Don't have prediction ability 	<ul style="list-style-type: none"> • Can generate reports and statistical graphs
Child vaccination schedule	<ul style="list-style-type: none"> • User can only add a schedule and they will get the reminder • Only mobile application 	<ul style="list-style-type: none"> • As mentioned about the system will automatically give a reminder to parents • Web and mobile applications will be made
Operability, Acceptability, and Usefulness of a Mobile App to Track Routine Immunization Performance in Rural Pakistan: Interview Study Among Vaccinators and Key Informants	<ul style="list-style-type: none"> • Only track record of people in rural areas • Not centralized • Only mobile application • Don't keep a record of children whose vaccination is not done • Don't keep a record of vaccinator and vaccine • Don't have a good UI 	<ul style="list-style-type: none"> • The system will be centralizing so it can be used in any area • It will keep a record of whether the child is vaccinated or not • It will keep track of the vaccinator so it will be easy if something wrong happens • The system will have a user-friendly UI

Table 1: Similar System Analysis

1.6 Methodology and Software Lifecycle for this Project

Below are the methodologies used while implementing this application

1.6.1 Process Methodology

We are going to use the incremental model in our project. Reasons for choosing an incremental model are

- Our project is long which consists of different modules
- Some modules can be implemented individually
- Requirements of the project are defined and won't change during development
- So, the incremental model suits our project as we can implement different modules and later we can develop and test them.
- In this way, the system will get a test in every possible way and we will get efficient results.

1.6.2 Design Methodology

We are going to Object Oriented Methodology. Reason for using OOP methodology is

- System can be decomposed in smaller parts components which makes management easy

- Communication between components become easy
- System becomes understandable and less complex which makes implementation easy
- System can be easily modified if required

2 Problem Definition

2.1 Problem Statement

All the developed countries have centralized health care system which helps them to track record of children and if someone is not getting his child vaccinated, he is penalized heavily. The reason being doing that is these are communicable diseases that can spread from one to others. These diseases include TB, Polio, Hepatitis, Flu, Measles, etc. To prevent these diseases to spread among others, immunization against them is done by the vaccine. As we know that Pakistan has poor health facilities and there is no proper infrastructure so no track of childbirth and their parent. And because of not having a centralized health care system it is difficult to track the children who are not getting the vaccination. Another thing is the manual system. People often forget the dates of vaccination of their children.

In many health care centers, there is a shortage of vaccines sometimes. One more problem is that sometimes children get expired vaccine which causes serious diseases and, in some cases, the child dies too. Apart from the vaccination-related problem, Pakistan is also facing a child undergrowth problem. Many children don't grow properly in the initial years which causes many deficiencies among them. Parents cannot monitor their child growth.

So, this system will cover all these problems and assist the parents and authorities in reducing these communicable diseases and monitor the growth of children so that no child gets undergrowth diseases.

2.2 Deliverables and Development Requirements

Following are the deliverables for Child Immunization and Growth Tracking system

1. Scope Document
2. Software Requirement Document
3. Software Design Description Document
4. Test Document
5. Final Report
6. Source Code of System
7. APK files for mobile applications

3 Requirement Analysis

3.1 Use Cases Diagram(s)

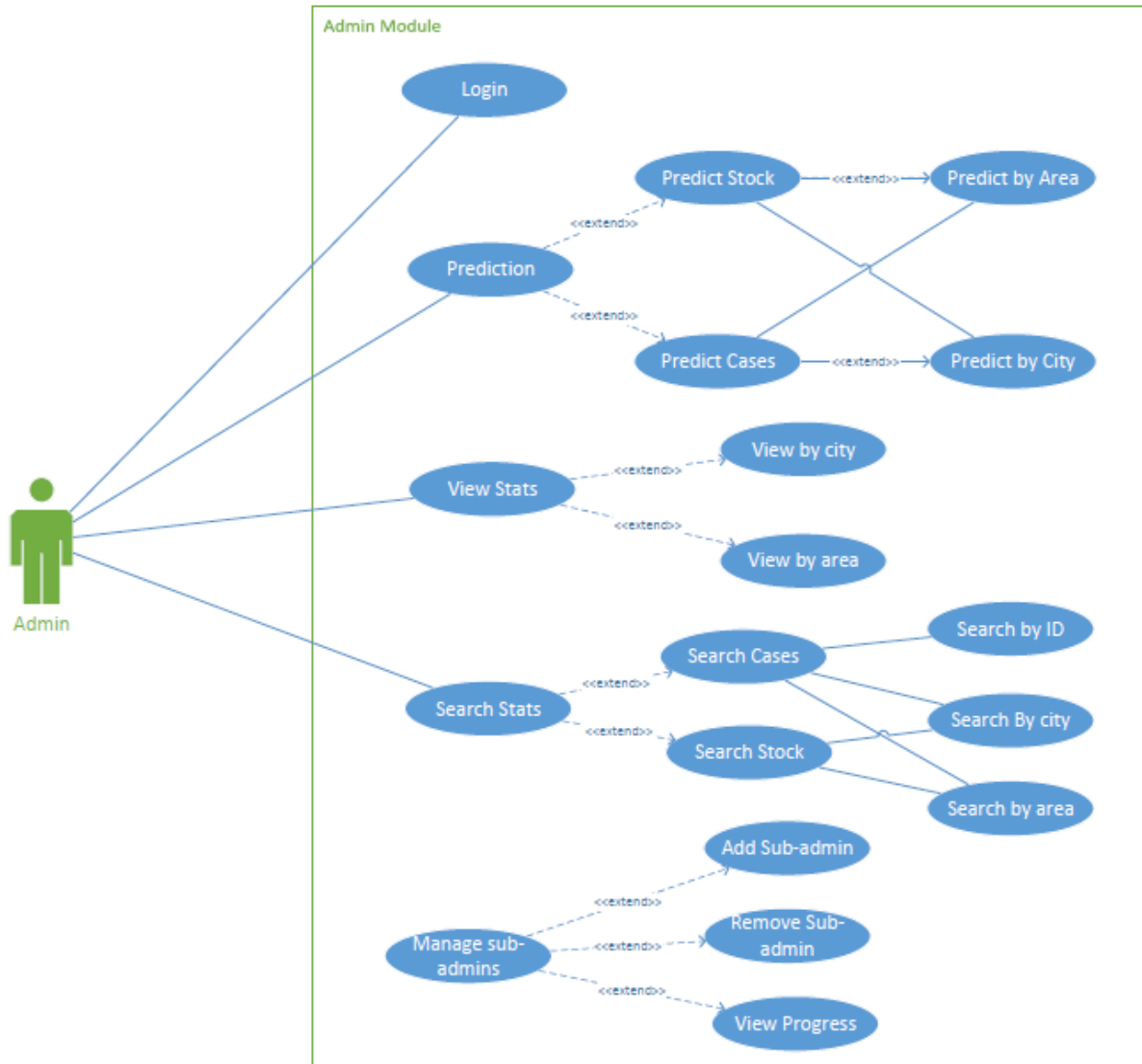


Figure 1:Usecase-Admin

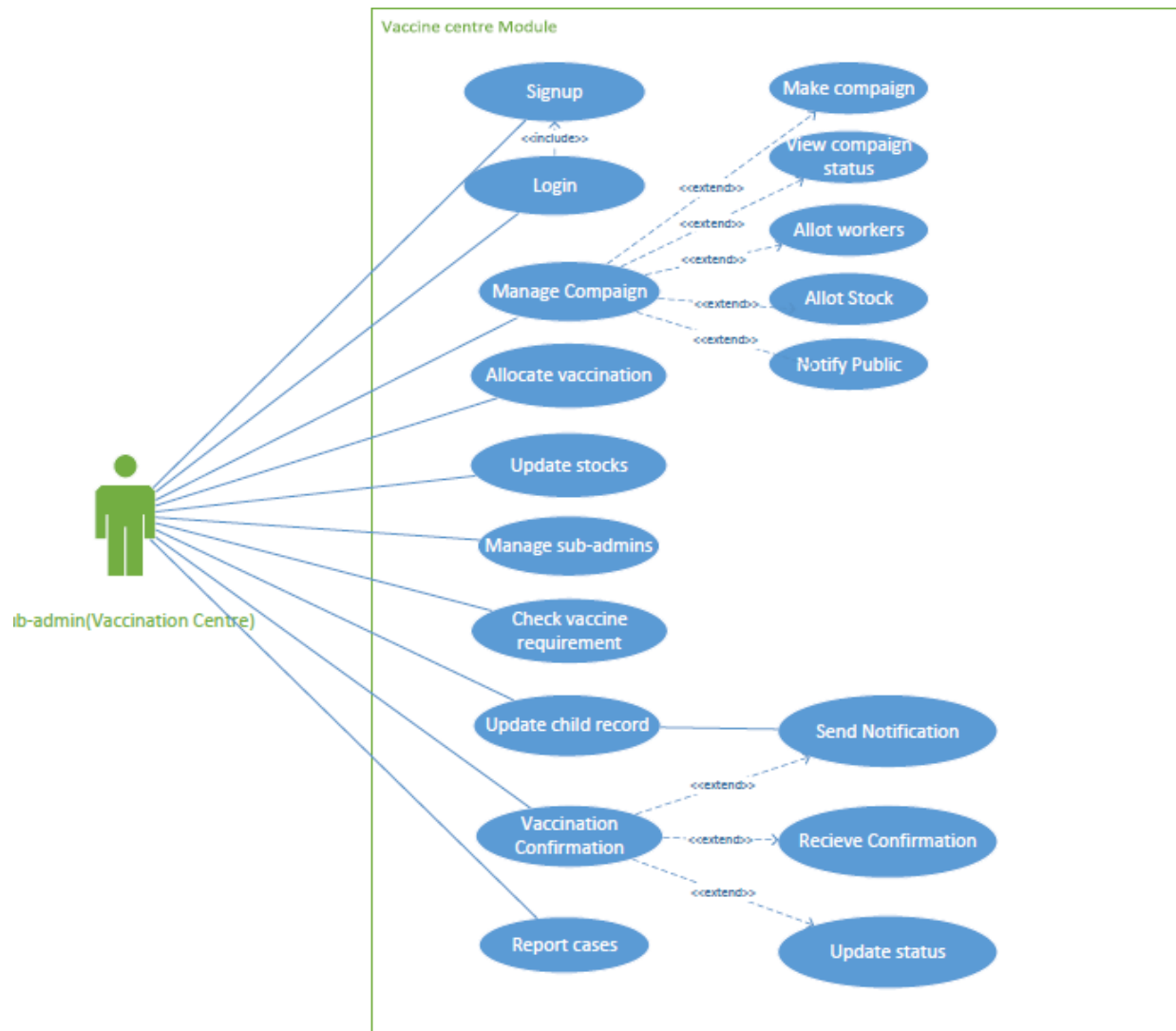


Figure 2: Use case Sub-admin(Vaccination center)

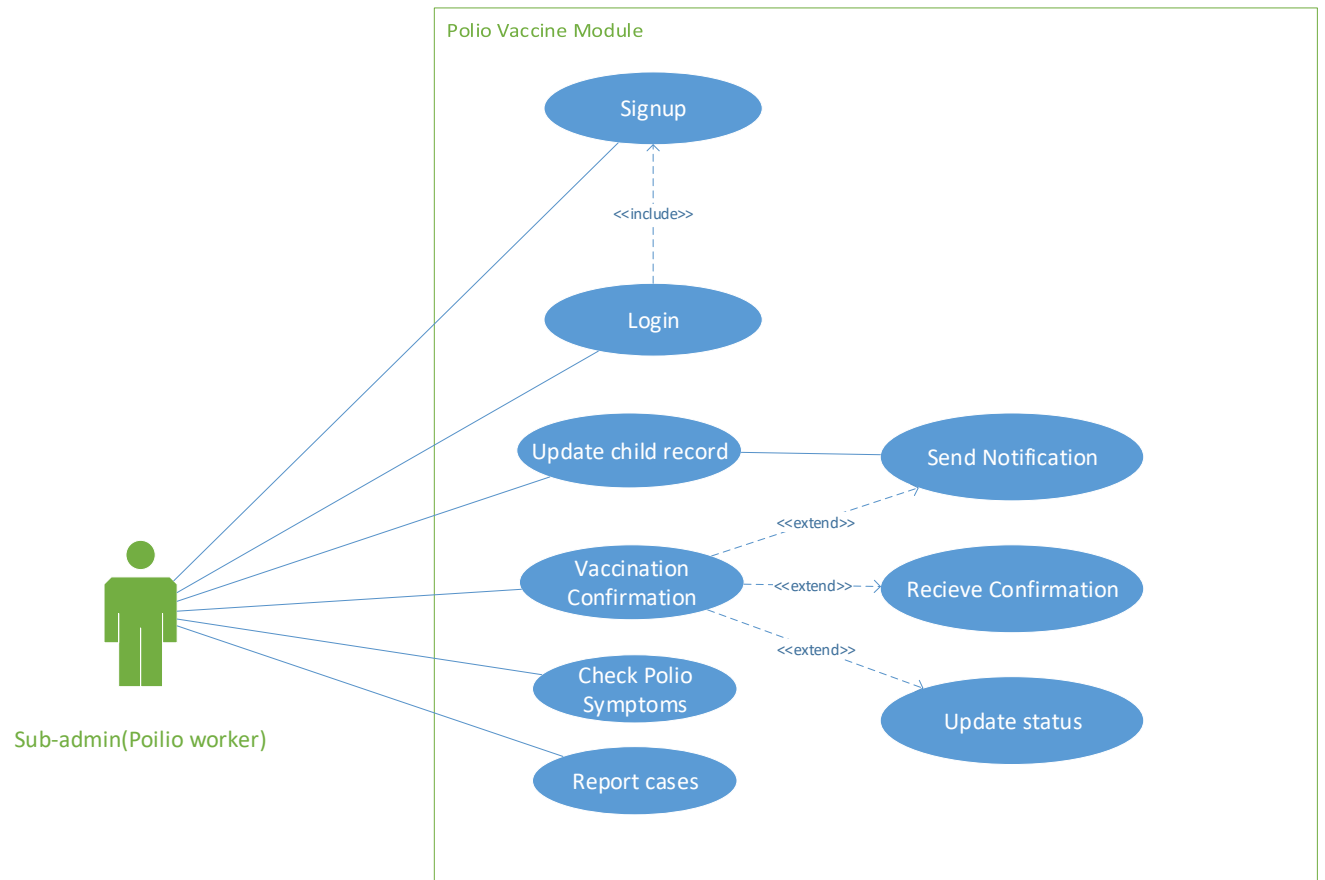


Figure 3:Usecase Sub-admin(Polio worker)

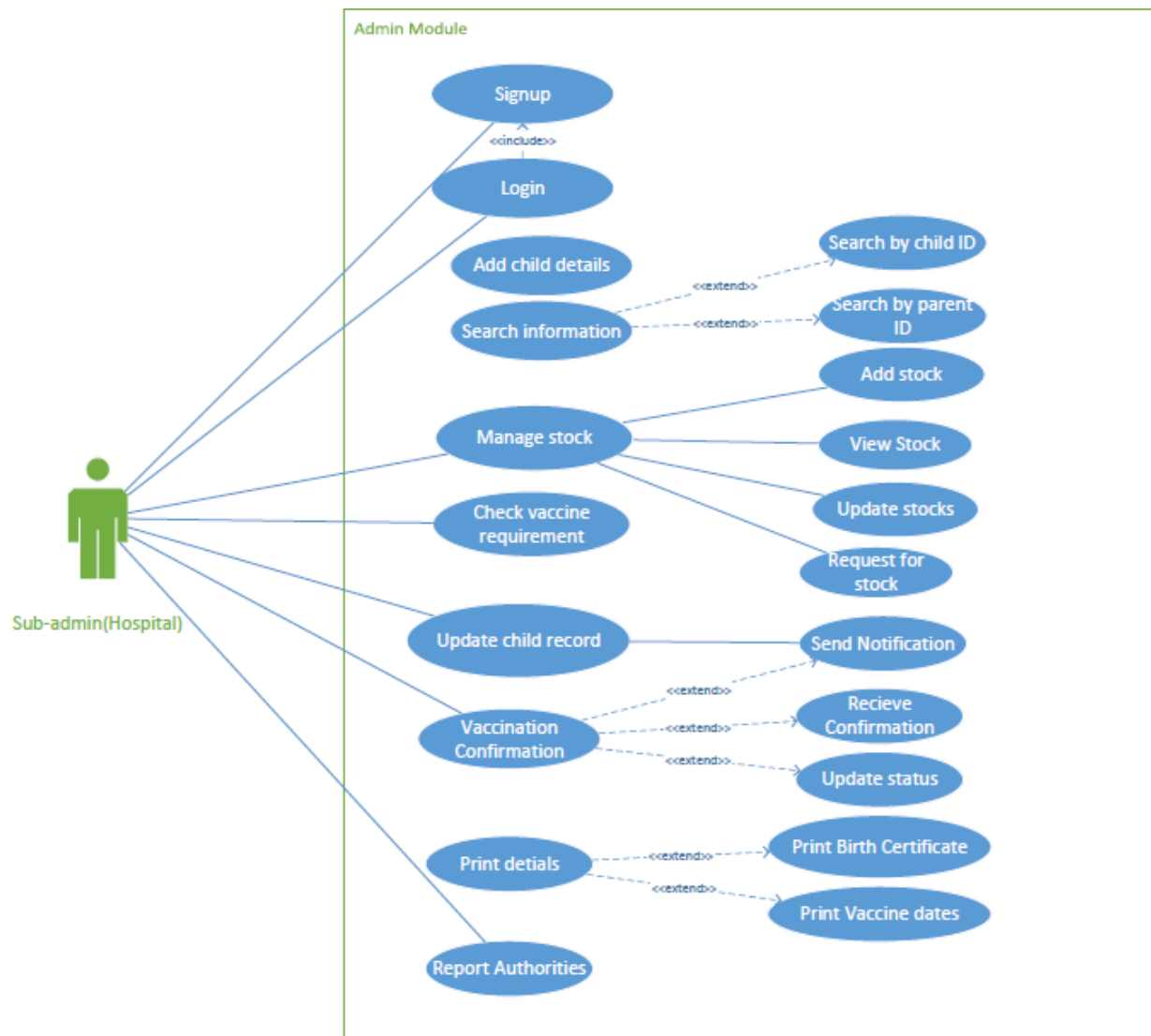


Figure 4:Usecase Sub-admin(Hospital)

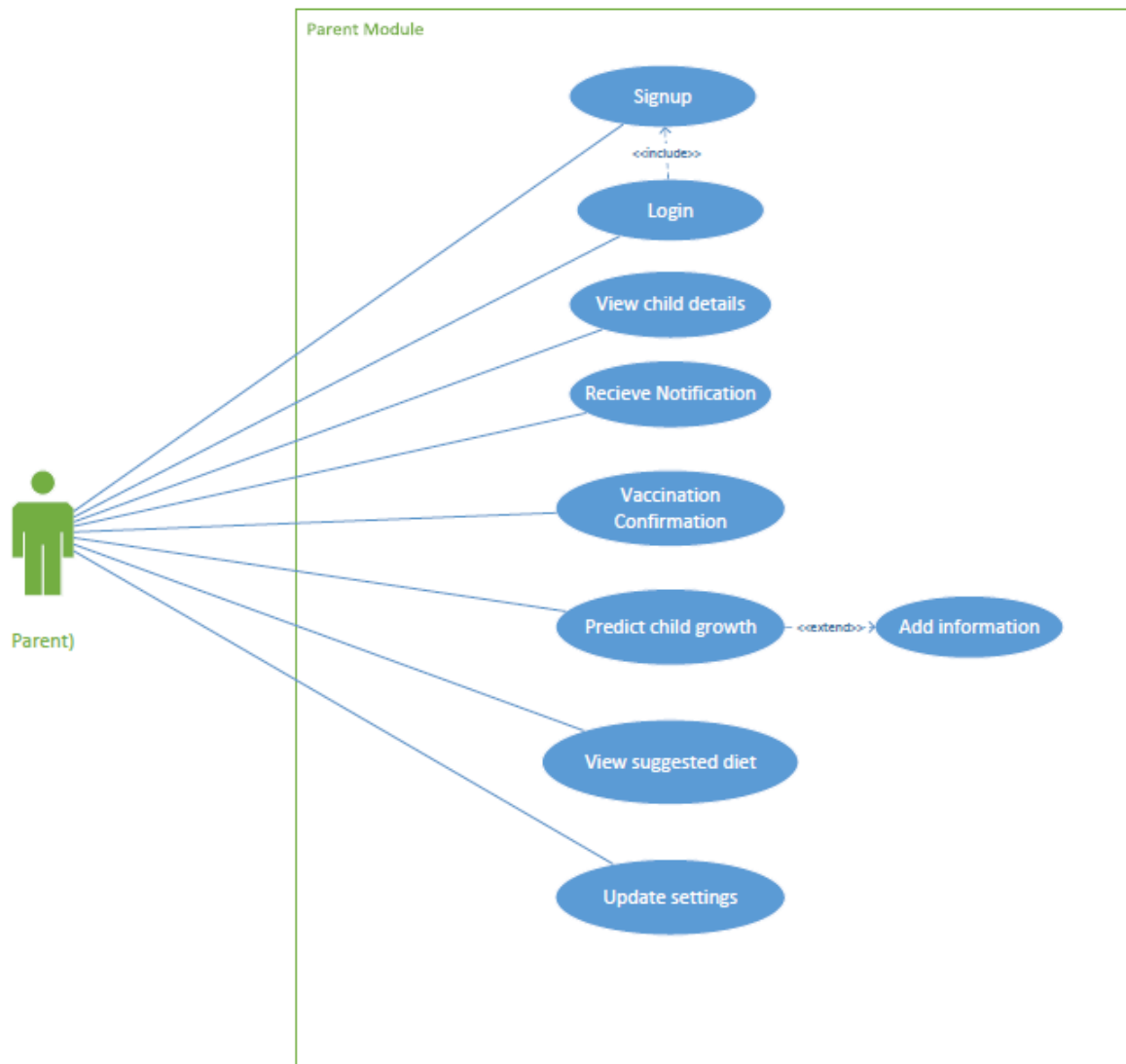


Figure 5:Usecase Sub-admin(Parent)

3.2 Detailed Use Case

Use Case ID:	UC-1
Use Case Name:	Login on Web app
Actors:	Primary Actor: Admin, Hospital, Vaccine Center
Description:	The user will be able to login and use the application.
Trigger:	A User will click on “Sign in” to go to the sign in page.
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User already have an account
Post conditions:	POST-1. User will be given access to fill form to login.
Normal Flow:	<ol style="list-style-type: none"> 1. User will type the web URL. 2. User will have option to Log in or Sign up option. 3. User will click on “log in” option. 4. User will select login as admin. 5. The form will be filled by user. 6. User will be logged in to the dashboard.
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User should have registered account.

Table 2:Textual Description of Admin

Use Case ID:	UC-2
Use Case Name:	Prediction
Actors:	Primary Actor: Admin
Description:	The admin will open the web application and will select the future prediction option.
Trigger:	A User will click on Prediction from the dashboard.
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the prediction screen.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on future prediction. 3. Another screen will be displayed on screen asking the user to choose between stock prediction and future cases prediction.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 3: Textual Description of Prediction

Use Case ID:	UC-3
Use Case Name:	Predict Stock
Actors:	Primary Actor: Admin
Description:	The admin will select the future prediction option and then will click on Stock Prediction.
Trigger:	A User will click on Stock Prediction from the prediction screen.
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the stock prediction screen.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on future prediction. 3. Another screen will be displayed on screen asking the user to choose between stock prediction and future cases prediction. 4. User will click on stock prediction.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 4:Textual Description of stock prediction

Use Case ID:	UC-4
Use Case Name:	Predict Stock by area/city
Actors:	Primary Actor: Admin
Description:	The admin will select the future prediction option and then will click on Stock Prediction and will select either by area or by city
Trigger:	A User will click on Stock Prediction from the prediction screen.
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the stock prediction screen.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on future prediction. 3. Another screen will be displayed on screen asking the user to choose between stock prediction and future cases prediction. 4. User will click on stock prediction. 5. User will select predict stock by city or by area to see the prediction.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 5:Textual Description of Stock Prediction by area/city

Use Case ID:	UC-5
Use Case Name:	Predict Cases
Actors:	Primary Actor: Admin
Description:	The admin will select the future prediction option and then will click on Cases Prediction.
Trigger:	A User will click on Cases Prediction from the prediction screen.
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the cases prediction screen.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on future prediction. 3. Another screen will be displayed on screen asking the user to choose between stock prediction and future cases prediction. 4. User will click on future case prediction.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 6:Textual Description of cases prediction

Use Case ID:	UC-6
Use Case Name:	Predict future cases by area/city
Actors:	Primary Actor: Admin
Description:	The admin will select the future prediction option and then will click on future cases Prediction and will select either by area or by city
Trigger:	A User will click on future cases Prediction from the prediction screen.
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the future cases prediction screen.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on future prediction. 3. Another screen will be displayed on screen asking the user to choose between stock prediction and future cases prediction. 4. User will click on future case prediction. 5. User will select predict future cases by city or by area to see the prediction.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 7:Textual Description of prediction of cases by city/area

Use Case ID:	UC-7
Use Case Name:	View Stats
Actors:	Primary Actor: Admin
Description:	The admin will select view stats from the navbar
Trigger:	A User will click on view stats from navbar
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the statistics screen.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on statistics. 3. Another screen will be displayed on screen where country stats will be show and asking the user to choose between city stats or area stats
Alternative Flows:	Whole country stats will already be displayed on dashboard from where you can also view stats.
Exceptions:	Stats will not be shown if there is no data available
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 8:Textual Description of view stats

Use Case ID:	UC-8
Use Case Name:	View Stats by city/area
Actors:	Primary Actor: Admin
Description:	The admin will select view stats by city or by area from the statistics screen.
Trigger:	A User will click on view stats from navbar
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1. User will be given access to the statistics screen where statistics will be shown.
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on statistics. 3. Another screen will be displayed on screen where country stats will be show and asking the user to choose between city stats or area stats. 4. User will select either city wise or area wise from the screen.
Alternative Flows:	Whole country stats will already be displayed on dashboard from where you can also view stats.
Exceptions:	User must not select area where our system is not available.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 9:Textual Description of view stats by city/area

Use Case ID:	UC-9
Use Case Name:	Search Stats
Actors:	Primary Actor: Admin
Description:	The admin will select search stats from the statistics screen
Trigger:	A User will click on search stats
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the search statistics screen where you will search stats
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on statistics. 3. Statistics screen will be displayed where you can search stats
Alternative Flows:	Whole country stats will already be displayed on dashboard user can also search stats from there.
Exceptions:	User must not search area where our system is not available.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 10:Textual Description of search statistics

Use Case ID:	UC-10
Use Case Name:	Search cases or stock info
Actors:	Primary Actor: Admin
Description:	The admin will select search stats from the statistics screen
Trigger:	A User will click on search stats
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1. User will be given access to the search statistics screen where user will search stats
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on statistics. 3. Statistics screen will be displayed where you can search stats 4. User will search cases stats or stock info.
Alternative Flows:	Whole country stats will already be displayed on dashboard you can also search from there.
Exceptions:	User must not search area where our system is not available.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 11:Textual Description of search stock stats

Use Case ID:	UC-11
Use Case Name:	Search child info
Actors:	Primary Actor: Admin
Description:	The admin will select search child info from the statistics screen
Trigger:	A User will click on search child info
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1.User will be given access to the search statistics screen where user will search child info
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on statistics. 3. Statistics screen will be displayed where you can search stats 4. User will select search child info. 5. User will enter child ID to see his info
Alternative Flows:	User can also select search child info from navbar
Exceptions:	User must not search child whose ID is invalid.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 12:Textual Description of specific child record

Use Case ID:	UC-12
Use Case Name:	Manage sub admins
Actors:	Primary Actor: Admin, Polio center
Description:	The admin will select manage sub admins from the navbar
Trigger:	A User will click on manage sub admins
Preconditions:	PRE-1. User has already accessed to the web page PRE-2- User must be logged in as admin
Post conditions:	POST-1. User will be given access to the manage sub admins screen where all the admins will be shown
Normal Flow:	<ol style="list-style-type: none"> 1. User will log in to the web application. 2. User will click on manage sub admins 3. Statistics screen will be displayed where user will add, remove sub admin or view progress.
Alternative Flows:	N/A
Exceptions:	User must not be able to manage sub-admins who are not registered.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User should be able to access website. 2. User must be logged in as admin.

Table 13:Textual Description of manage sub admins

Tabular use cases for Parent:

Use Case ID:	UC-13
Use Case Name:	Sign up from mobile app
Actors:	Primary Actor: Parent, Polio Worker Secondary Actor: Admin
Description:	The user will be able to Register account and use the application.
Trigger:	A User will click on “Sign up” to start registering.
Preconditions:	PRE-1. User has already downloaded the application
Post conditions:	POST-1. User will be given access to fill from to register account.
Normal Flow:	<ol style="list-style-type: none">1. User will download application.2. User will have option to Log in or Sign up option.3. User will click on “sign up” option.4. User will select sign up as parent/polio worker.5. The form will be filled by user.6. User will be registered.
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none">1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none">1. User should be downloaded the application2. User should have an email account and phone no.

Table 14:Textual Description of sign up

Use Case ID:	UC-14
Use Case Name:	Sign in from mobile app
Actors:	Primary Actor: Parent, Polio Worker Secondary Actor: Admin
Description:	The user will be able to login and use the application.
Trigger:	A User will click on “Sign in” to go to the landing page.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be given access to the landing page
Normal Flow:	<ol style="list-style-type: none"> 1. User will download application. 2. User will have option to Log in or Sign up option. 3. User will click on “sign in” option. 4. User will the form with email and password. 5. User will be logged in to the landing page.
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered.

Table 15: Textual Description of sign in

Use Case ID:	V15
Use Case Name:	View child info
Actors:	Primary Actor: Parent Secondary Actor: Admin
Description:	The user will be able to see his child info
Trigger:	A User will click on see child info from the landing page.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be given access to the child info screen where he will enter his ID
Normal Flow:	<ol style="list-style-type: none"> 1. User will click in check child info. 2. User will enter his child ID which is written on birth certificate. 3. User will see his child details
Alternative Flows:	N/A
Exceptions:	1. If his child is not registered in our system, details will not be shown.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered. 2. Child must also be registered in our system.

Table 16:Textual Description of view child stats

Use Case ID:	UC-16
Use Case Name:	Vaccination Confirmation
Actors:	Primary Actor: Parent
Description:	The user will be able to confirm his child vaccination
Trigger:	A User will receive notification which he will confirm.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will confirm his child vaccination
Normal Flow:	<ol style="list-style-type: none"> 1. User will receive notification of his child vaccination. 2. User will confirm it,
Alternative Flows:	N/A
Exceptions:	1. If his child is not registered in our system, no notification will be received.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered. 2. Child must also be registered in our system.

Table 17: Textual Description of Vaccine Confirmation

Use Case ID:	UC-17
Use Case Name:	Check child growth
Actors:	Primary Actor: Parent
Description:	The user will be able to check his child growth.
Trigger:	A User will click on check child growth
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be accessed to the growth checking screen where he will enter his child details.
Normal Flow:	<ol style="list-style-type: none"> 1. User will click on check child growth. 2. User will enter his child details like height, weight etc. 3. User will click on submit. 4. User will see his child growth.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered.

Table 18: Textual Description of Check Child Growth

Use Case ID:	UC-18
Use Case Name:	View diet plans
Actors:	Primary Actor: Parent
Description:	The user will be able to check his child growth and then view the diet plan according to the growth.
Trigger:	A User will click on see suggested diet.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be accessed to the growth checking screen where he will enter his child details and then see diet plans.
Normal Flow:	<ol style="list-style-type: none"> 1. User will click on check child growth. 2. User will enter his child details like height, weight etc. 3. User will click on submit. 4. User will see his child growth. 5. User will click on see suggested diet. 6. Suggested diet will be displayed.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered.

Table 19: Textual Description of View Diet Plan

Use Case ID:	UC-19
Use Case Name:	Update settings
Actors:	Primary Actor: Parent
Description:	The user will be able to update his settings.
Trigger:	A User will click on settings.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be accessed to the settings screen
Normal Flow:	<ol style="list-style-type: none"> 1. User will click on settings. 2. User will update his settings i.e change profile pic, change password etc.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered.

Table 20:Textual Description of update profile

Use cases of Polio Worker:

Use Case ID:	UC-20
Use Case Name:	Update child record
Actors:	Primary Actor: Polio Worker
Description:	The user will be able to update child record whose vaccination is done.
Trigger:	A User will click on update child record.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be able to update child vaccination data
Normal Flow:	<ol style="list-style-type: none">1. User will click on update child data2. User will enter updated vaccines info.3. Notification will be sent to parent4. Parent will confirm it.5. Data will be updated.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none">1. User must be registered.

Table 21: Textual Description of Update Child data

Use Case ID:	UC-21
Use Case Name:	Report cases
Actors:	Primary Actor: Polio Worker
Description:	The user will be able to report authorities about polio cases.
Trigger:	A User will click on report authorities.
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be able to send data to authorities i.e admin
Normal Flow:	<ol style="list-style-type: none"> 1. User will click on report authorities 2. User will click on send report 3. A report will be sent to authorities.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered.

Table 22:Textual Description of Report Authorities

Use Case ID:	UC-22
Use Case Name:	Check Polio symptoms
Actors:	Primary Actor: Polio Worker Secondary Actor: Parent
Description:	This will help workers to check Polio symptoms on regular basis
Trigger:	A User will click on check polio symptoms
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1. User will be able to get results on polio symptoms
Normal Flow:	Polio Worker will click on Check polio symptoms Polio worker will add additional information taken from parents Polio worker will click on check and results will display
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	2. User must be registered.

Table 23: Textual Description of report authorities

Tabular Use cases for Hospital:

Use Case ID:	UC-23
Use Case Name:	Signup
Actors:	Primary Actor: Sub-admin(Hospital)
Description:	The user will be able to Register account and use the application.
Trigger:	A User will click on “Sign up” to start registering.
Level	High
Preconditions:	PRE-1. User has already accessed to the web page
Post conditions:	POST-1. User will be given access to fill from to register account.
Normal Flow:	<ol style="list-style-type: none">1. User will type the web URL2. User will have option to Log in or Sign up option.3. User will click on “sign up” as hospital.4. User will select sign up as hospital.5. The form will be filled by user.6. User will be registered.
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none">1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none">1. User should be able to access website.2. User should have an email account and phone no.

Table 24:Textual Description of hospital signup

Use Case ID:	UC-24
Use Case Name:	Sign in as hospital
Actors:	Primary Actor: Sub-admin(Hospital)
Description:	The user will be able to login and use the application.
Trigger:	A User will click on “Sign in” to go to the landing page.
Level	High
Preconditions:	PRE-1. User has already signed up to the application
Post conditions:	POST-1.User will be given access to fill from to register account.
Normal Flow:	<ol style="list-style-type: none"> 1. User will type the web URL or download application. 2. User will have option to Log in or Sign up option. 3. User will click on “sign in” option. 4. User will the form with email and password. 5. User will be logged in to the landing page.
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. Sub-admin must be registered.

Table 25:Textual Description of hospital sign in

Use Case ID:	UC-25
Use Case Name:	Add child
Actors:	Primary Actor: Sub-admin(Hospital)
Description:	User will add all basic information of parent and child
Trigger:	A user will click on the add child
Level	High
Preconditions:	PRE-1. User has already signed in to the application
Post conditions:	POST-1. User will be able to add a new born child information in the system
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on add child. 3. User will fill all the necessary details of parent and child. 4. User will click on submit. 5. User will see the information and code assigned to new baby.
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor doesnot fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered. 2. Child must be a new born baby

Table 26:Textual Description of Add child information

Use Case ID:	UC-26
Use Case Name:	Search child
Actors:	Primary Actor: Sub-admin(Hospital)
Description:	User will type child name or id and his information will be shown. It will help user to see and update the child record of vaccine.
Trigger:	A user will click on the add child
Level	High
Preconditions:	PRE-1. User has already signed in to the application
Post conditions:	POST-1. User will be able to view and update the existing child information in the system
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will add id of child in search tab. 3. User will click on search button. 4. Child data will appear and user can see the child information 5. User can select child and update its vaccine information
Alternative Flows:	<p>User can also search the child through parent id</p> <p>If parent have multiple child, user can see, select and update it</p>
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered. 2. Child must be a new born baby 3. Child must be registered before

Table 27:Textual Description of search child details

Use Case ID:	UC-27
Use Case Name:	Manage stock (Add or request)
Actors:	Primary Actor: Sub-admin(Hospital), Vaccine center
Description:	User will be able to add or request for vaccine stock to the admin
Trigger:	A user will click on the manage stock
Level	High
Preconditions:	PRE-1. User has already signed in to the application
Post conditions:	POST-1. User will be able to add a new born child information in the system
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on manage stock and will see multiple options there. 3. User will click on the add stock which will be allotted by vaccine center. 4. User will click on add stock and stock will be shown in the record. 5. User can also request for stock to vaccine center or admin
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields. 2. if stock is not allotted then you can't add
Business Rules	<ol style="list-style-type: none"> 1. Stock must be allotted by vaccine center before and its expiry and manufacturing dates should be clearly mentioned 2. This is only valid for the new vaccines like corona etc, for existing vaccines user can only use update information
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered. 2. Vaccines are allotted by admin 3. User is adding new vaccines in the system

Table 28:Textual Description of manage stocks

Use Case ID:	UC-28
Use Case Name:	Manage stock (View or update)
Actors:	Primary Actor: Sub-admin(Hospital), Vaccine center
Description:	User will be able to view or update the existing vaccines record
Trigger:	A user will click on the manage stock
Level	High
Preconditions:	PRE-1. User has already signed in to the application PRE-2. Vaccines are already existing in the system
Post conditions:	POST-1. User will be able to view or update the existing vaccine stock
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on manage stock and will see multiple options there. 3. User will click on the view stock and multiple vaccines will be shown there 4. User will click on the specific vaccine which need upgradation 5. User will add new stock information 6. User will click on update stock and stock will be updated 7. User will see the updated stock.
Alternative Flows:	Same process can be followed in the vaccine center
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields. 2. if stock is not allotted then you can't update
Business Rules	<ol style="list-style-type: none"> 1. Stock must be allotted by vaccine center before and its expiry and manufacturing dates should be clearly mentioned 2. This is only valid for the existing vaccines.
Assumptions:	<ol style="list-style-type: none"> 1. User must be registered. 2. Vaccines are allotted by admin. 3. User is updating new vaccines.

Table 29:Textual Description of manage stocks(Vaccine center)

Use Case ID:	UC-29
Use Case Name:	Check vaccine requirements
Actors:	Primary Actor: Sub-admin(Hospital), Vaccine center
Description:	This is an algorithm based feature which will help the admins to check the upcoming requirement of vaccine in a particular area or hospital
Trigger:	A user will click check upcoming requirement button
Level	High
Preconditions:	PRE-1. All details of new born child and vaccine are already entered in the system
Postconditions:	POST-1. User will be able to view the upcoming vaccine stock requirement
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on check upcoming requirements of vaccine 3. System at backend will work and check how many children belongs to this area. 4. On this information, system will tell the upcoming requirements of vaccine 5. User will be able to see this 6. User can request to admin for this requirement.
Alternative Flows:	Same process can be followed in the vaccine center
Exceptions:	N/A
Business Rules	This is only valid for the existing vaccines.
Assumptions:	<p>User must be registered.</p> <p>Vaccines are allotted by admin.</p>

Table 30:Textual Description of vaccine requirments

Use Case ID:	UC-30
Use Case Name:	Print details
Actors:	Primary Actor: Sub-admin(Hospital)
Description:	Through this feature hospital can give birth certificate and vaccination codes to parents
Trigger:	A user will click print details
Level	High
Preconditions:	PRE-1. All details of new born child is already entered in the system
Post conditions:	POST-1.User will be able to provide the birth certificate or vaccine details
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on print details 3. User can click on print birth certificate 4. A birth certificate will be provided to the end user(Parent). 5. User can click on print vaccine details 6. Vaccine details will be provided to the end user(Parent).
Alternative Flows:	User can use control + p to print both things
Exceptions:	N/A
Business Rules	N/A
Assumptions:	User must be registered.

Table 31:Textual Description of print birth certificate

Tabular use cases of Vaccine center, Polio center:

Use Case ID:	UC-31
Use Case Name:	Manage campaign (Make)
Actors:	Primary Actor: Vaccine center, Polio center
Description:	User will be able to make new vaccine campaigns time to time
Trigger:	A user will click on the manage campaigns
Level	High
Preconditions:	PRE-1. User has already signed in to the application PRE-2. User is authorized by admin to make these campaigns
Post conditions:	POST-1. User will be able to make a new campaign of vaccine for general public
Normal Flow:	1. User will login his account. 2. User will click on manage campaign. 3. User will click on make new campaign. 4. User will add details like area, starting and ending dates etc 5. User will click on make campaign
Alternative Flows:	N/A
Exceptions:	1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	User must be registered. Vaccines are allotted by admin User is authorized by Admin

Table 32:Textual Description of manage campaigns

Use Case ID:	UC-32
Use Case Name:	Manage campaign (View)
Actors:	Primary Actor: Vaccine center, Polio center
Description:	User will be able to view existing vaccine campaigns.
Trigger:	A user will click on the manage campaigns
Level	High
Preconditions:	PRE-1. User has already signed in to the application PRE-2. User is authorized by admin to make these campaigns
Post conditions:	POST-1. User will be able to see the status of existing vaccine campaigns
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on manage campaign. 3. User will click View campaigns 4. User will be able to see the status of campaigns running
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields.
Business Rules	N/A
Assumptions:	User must be registered. Vaccines are allotted by admin User is authorized by Admin

Table 33: Textual Description of view campaign results

Use Case ID:	UC-33
Use Case Name:	Manage campaign (Allotment)
Actors:	Primary Actor: Vaccine center, Polio center
Description:	User will be able to Allot workers and vaccine stock to the campaigns
Trigger:	A user will click on the manage campaigns
Level	High
Preconditions:	PRE-1. User has already signed in to the application PRE-2. User is authorized by admin to manage these campaigns
Post conditions:	POST-1. Workers are allotted the area and stock of vaccines
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on manage campaign. 3. User will click on the running campaigns 4. User will click on allotment 5. User will allot workers and vaccine stock to the campaigns 6. User will click on the update.
Alternative Flows:	Same procedure can be done while making campaigns
Exceptions:	<ol style="list-style-type: none"> 1. If actor does not fill the required fields. Prompt message will be shown to user to fill in required fields. 2. User is not authenticated
Business Rules	N/A
Assumptions:	N/A

Table 34:Textual Description of Allot vaccines

Use Case ID:	UC-34
Use Case Name:	Manage campaign (Notify)
Actors:	Primary Actor: Vaccine center, Polio center
Description:	This feature is to notify the public via sms about the vaccine campaigns so that they can get their children vaccinated
Trigger:	A user will click on the notify public
Level	High
Preconditions:	PRE-1. User has already signed in to the application PRE-2. User is authorized by admin to manage campaigns
Post conditions:	POST-1. Parents will receive sms notification about campaigns
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on manage campaign. 3. User will click on the running campaigns 4. User will click on notify public. 5. System will send notification to the end users.
Alternative Flows:	Same procedure can be done while making campaigns
Exceptions:	<ol style="list-style-type: none"> 1. User is not authenticated 2. Correct contact numbers are not provided
Business Rules	N/A
Assumptions:	N/A

Table 35: Textual Description of campaign notification

Use Case ID:	UC-35
Use Case Name:	Allocate stock
Actors:	Primary Actor: Vaccine center, Polio center
Description:	User will be able to allot vaccination stock to the sub admins
Trigger:	A user will click allot stock button
Level	High
Preconditions:	PRE-1. User has already signed in to the application PRE-2. User is authorized by admin to allot these vaccine stock
Post conditions:	POST-1.Sub-admins are allotted stock of vaccines
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click allot stock 3. User will fill the details and stock will be allotted 4. User will click on allot and sub-admins can see the allotted stock
Alternative Flows:	N/A
Exceptions:	<ol style="list-style-type: none"> 1. If actor doesnot fill the required fields. Prompt message will be shown to user to fill in required fields. 2. User is not authenticated
Business Rules	N/A
Assumptions:	N/A

Table 36:Textual Description of allot stocks

Use Case ID:	UC-36
Use Case Name:	Report cases
Actors:	Primary Actor: Vaccine center, Polio center
Description:	User will be able report the cases to admins which got unvaccinated
Trigger:	A user will click on the report to admin button
Level	High
Preconditions:	PRE-1. User has already signed in to the application Pre-2. Parents has been notified many times
Post conditions:	POST-1.Admin will get the data of such child who didn't got vaccinated
Normal Flow:	<ol style="list-style-type: none"> 1. User will login his account. 2. User will click on report to admin 3. A system generated report will appear in front of user 4. User will click on report.
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	N/A

Table 37:Textual Description of Report Cases

3.3 Functional Requirements

Identifier	FR-1
Title	Login as admin
Requirement	The user shall be able to login
Source	Admin, Hospital, Vaccine Center
Rationale	To login into the system
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 38: Textual Description of FR-1

Identifier	FR-2
Title	Input information
Requirement	The user shall be able to enter email, password in the required fields to login
Source	Admin, Hospital, Vaccine Center
Rationale	User will enter his login information to login into the system
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 39: Textual Description of FR-2

Identifier	FR-3
Title	Check Remember me check box
Requirement	The user shall be able to check remember me check box.
Source	Hospital, Admin, Vaccine Center
Rationale	To check remember me check box so that user will be remembered when he visit the website.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 40: Textual Description of FR-3

Identifier	FR-4
Title	Click on Login Button
Requirement	The user shall be able to click on login button to login to the web application
Source	Hospital, Vaccine center, admin
Rationale	To login to the system
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 41: Textual Description of FR-4

Identifier	FR-5
Title	Prediction
Requirement	The user shall be able to click on prediction tab from nav bar.
Source	Admin
Rationale	User will click on prediction to predict future cases and stocks.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Medium

Table 42: Textual Description of FR-5

Identifier	FR-6
Title	Predict Vaccine Stock
Requirement	The user shall be able to click on predict stock from the prediction screen.
Source	Admin
Rationale	To obtain vaccine stock prediction for future needs.
Business Rule (if required)	N/A
Dependencies	FR-5
Priority	Medium

Table 43: Textual Description of FR-6

Identifier	FR-7
Title	Predict future cases
Requirement	The user shall be able to click on predict future cases from the prediction screen.
Source	Admin
Rationale	To obtain future cases prediction.
Business Rule (if required)	N/A
Dependencies	FR-5
Priority	Medium

Table 44: Textual Description of FR-7

Identifier	FR-8
Title	Predict future cases by city, area
Requirement	The user shall be able to predict cases by city or area from the prediction screen.
Source	Admin
Rationale	To obtain future cases prediction by city/area.
Business Rule (if required)	N/A
Dependencies	FR-5
Priority	Medium

Table 45: Textual Description of FR-8

Identifier	FR-9
Title	Predict future stock needs by city, area
Requirement	The user shall be able to predict future vaccine stock needs by city or area from the stock prediction screen.
Source	Admin
Rationale	To obtain future stock needs by city/area.
Business Rule (if required)	N/A
Dependencies	FR-5
Priority	Medium

Table 46: Textual Description of FR-9

Identifier	FR-10
Title	View Stats
Requirement	The user shall be able to view stats of entire country
Source	Admin
Rationale	To view stats of entire country.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	medium

Table 47: Textual Description of FR-10

Identifier	FR-11
Title	View Stats by city/area
Requirement	The user shall be able to view stats of specific city or area.
Source	Admin
Rationale	To view stats of specific city or area.
Business Rule (if required)	N/A
Dependencies	FR-10
Priority	medium

Table 48: Textual Description of FR-11

Identifier	FR-12
Title	Search Stats
Requirement	The user will click on search stats to search.
Source	Admin
Rationale	To search stats.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 49: Textual Description of FR-12

Identifier	FR-13
Title	Search Stock by city/area
Requirement	The user will click on search stock stats from the search screen
Source	Admin
Rationale	To search stock stats by city/area.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 50: Textual Description of FR-13

Identifier	FR-14
Title	Search Cases by city/area
Requirement	The user will click on search cases stats from the search screen
Source	Admin
Rationale	To search cases stats by city/area.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 51: Textual Description of FR-14

Identifier	FR-15
Title	Search child info
Requirement	The user will click on search child info from the search screen
Source	Admin
Rationale	To search child vaccination info by entering his ID to the search field.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 52: Textual Description of FR-15

Identifier	FR-16
Title	Manage sub admins
Requirement	The user will click on manage sub admins.
Source	Admin
Rationale	To manage sub admins.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 53: Textual Description of FR-16

Identifier	FR-17
Title	Add sub admin
Requirement	The user will add new sub admin from manage admin screen.
Source	Admin
Rationale	To add new sub admins from the system.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 54: Textual Description of FR-17

Identifier	FR-18
Title	Remove sub admin
Requirement	The user will remove sub admin from manage admin screen.
Source	Admin
Rationale	To remove sub admins from the system.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 55: Textual Description of FR-18

Identifier	FR-19
Title	View sub admin progress
Requirement	The user will view sub admin progress.
Source	Admin
Rationale	To view sub admin progress by click on sub admin name from manage sub admins screen.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 56: Textual Description of FR-19

Functional Requirements of Parent:

Identifier	FR-20
Title	Sign up
Requirement	The user will click on sign up to register to the application.
Source	Hospital, Polio worker
Rationale	To Register account.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 57: Textual Description of FR-20

Identifier	FR-21
Title	Fill information
Requirement	The user shall be able to fill information in the input fields
Source	Hospital, Polio worker
Rationale	To input information of user i.e name, email, password etc.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 58: Textual Description of FR-21

Identifier	FR-22
Title	Sign up Button
Requirement	The user shall be able to click on sign up button to register.
Source	Hospital, Polio worker
Rationale	To Register account by clicking on sign up
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 59: Textual Description of FR-22

Identifier	FR-23
Title	Login button
Requirement	The user shall be able to click on login button if he already have an account.
Source	Hospital, Polio worker
Rationale	To Login into the system if user already have an account
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 60: Textual Description of FR-23

Identifier	FR-24
Title	Login
Requirement	The user shall be able to login from mobile app
Source	Parent, Polio worker
Rationale	To login into the system from mobile application
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 61: Textual Description of FR-24

Identifier	FR-25
Title	Input information
Requirement	The user shall be able to enter email, password in the required fields to login from mobile app
Source	Parent, Polio worker
Rationale	User will enter his login information to login into the system from mobile app.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 62: Textual Description of FR-25

Identifier	FR-26
Title	Check Remember me check box
Requirement	The user shall be able to check remember me check box.
Source	Parent, Polio worker
Rationale	To check remember me check box so that user will be remembered when he open the mobile app.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 63: Textual Description of FR-26

Identifier	FR-27
Title	Click on Login Button
Requirement	The user shall be able to click on login button to login to the mobile application
Source	Parent, Polio worker
Rationale	To login to the system from mobile app
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 64: Textual Description of FR-27

Identifier	FR-28
Title	Click on View child details
Requirement	The user shall be able to click on view child details to view his child information
Source	Parent
Rationale	To view the child vaccination and other details
Business Rule (if required)	N/A
Dependencies	N/A
Priority	medium

Table 65: Textual Description of FR-28

Identifier	FR-29
Title	Enter child ID
Requirement	The user shall be able to enter his child ID to view his details
Source	Parent
Rationale	To enter child ID to view his details.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	medium

Table 66: Textual Description of FR-29

Identifier	FR-30
Title	Receive notifications
Requirement	The user shall be able to receive notifications when vaccine center will update his child info.
Source	Parent
Rationale	To receive notifications.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	medium

Table 67: Textual Description of FR-30

Identifier	FR-31
Title	Confirm vaccinations
Requirement	The user shall be able to confirm vaccination of his child.
Source	Parent
Rationale	To confirm vaccination.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	medium

Table 68: Textual Description of FR-31

Identifier	FR-32
Title	Check child growth
Requirement	The user shall be able to click on check child growth.
Source	Parent
Rationale	To check growth of child.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 69: Textual Description of FR-32

Identifier	FR-33
Title	Fill information
Requirement	The user shall be able to fill his child information to check his child growth
Source	Parent
Rationale	To fill child info to see child growth
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 70: Textual Description of FR-33

Identifier	FR-34
Title	Click submit
Requirement	The user shall be able to click on submit to see his child growth.
Source	Parent
Rationale	To click on submit to check growth of child.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 71: Textual Description of FR-34

Identifier	FR-35
Title	View suggested diet
Requirement	The user shall be able to click on view suggested diet from child growth screen
Source	Parent
Rationale	To click on view suggested diets to view diet
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 72: Textual Description of FR-35

Identifier	FR-36
Title	Update settings
Requirement	The user shall be able to click on settings to update his user settings
Source	Parent, Polio worker
Rationale	To click on settings to update his settings
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 73: Textual Description of FR-36

Identifier	FR-37
Title	Update password
Requirement	The user shall be able to click on update password from settings
Source	Parent, Polio worker
Rationale	To click on update password from settings to update password.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 74: Textual Description of FR-37

Identifier	FR-38
Title	Add new password
Requirement	The user shall be able to add new password from the update password screen.
Source	Parent, Polio worker
Rationale	To fill update password fields to update password.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 75: Textual Description of FR-38

Functional Requirements of Polio Worker:

Identifier	FR-39
Title	Update child record
Requirement	The user shall be able to update child vaccination record.
Source	Polio worker
Rationale	To update child vaccination record.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 76: Textual Description of FR-39

Identifier	FR-40
Title	Send notifications
Requirement	The user shall be able to send notifications when he updates child vaccination record.
Source	Polio worker
Rationale	To send notification to parent.
Business Rule (if required)	N/A
Dependencies	FR-39
Priority	low

Table 77: Textual Description of FR-40

Identifier	FR-41
Title	Receive confirmation
Requirement	The user shall be able to receive confirmation of vaccination from parent when he updates the child vaccination info.
Source	Polio worker
Rationale	To receive confirmation of vaccination from parent.
Business Rule (if required)	N/A
Dependencies	FR-39
Priority	low

Table 78: Textual Description of FR-41

Identifier	FR-42
Title	Update status
Requirement	The user shall be able to update child vaccination status when the parents confirm vaccination.
Source	Polio worker
Rationale	To update child vaccination status when parent confirms vaccination.
Business Rule (if required)	N/A
Dependencies	FR-39
Priority	low

Table 79: Textual Description of FR-42

Identifier	FR-43
Title	Report cases
Requirement	The user shall be able to report authorities about polio cases.
Source	Polio worker
Rationale	To report authorities about polio cases.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 80: Textual Description of FR-43

Identifier	FR-44
Title	Get Symptoms Data
Requirement	The user shall be able to add symptoms data in system
Source	Polio worker
Rationale	To check polio symptoms
Business Rule (if required)	N/A
Dependencies	N/A
Priority	low

Table 81: Textual Description of FR-44

Identifier	FR-45
Title	Check Symptoms
Requirement	The user shall be able to check symptoms through system
Source	Polio worker
Rationale	To check polio symptoms
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 82: Textual Description of FR-45

Functional Requirements of sub-admin (Hospital):

Identifier	FR-46
Title	Sign up from web app.
Requirement	The user shall be able to click on sign up from web application
Source	Hospital, Vaccination center
Rationale	To Register account as hospital, vaccination center
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 83: Textual Description of FR-46

Identifier	FR-47
Title	Fill information
Requirement	The user shall be able to fill information for sign up
Source	Hospital, Vaccination center
Rationale	To fill information about sign up like name, email, password etc
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 84: Textual Description of FR-47

Identifier	FR-48
Title	Sign up Button
Requirement	The user shall be able to click on sign up button to register.
Source	Hospital, Vaccination center
Rationale	To Register account.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 85: Textual Description of FR-48

Identifier	FR-49
Title	Login button
Requirement	The user shall be able to click on login button if he already have an account.
Source	Hospital, Vaccination center
Rationale	To Login account if user already have an account.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 86: Textual Description of FR-49

Identifier	FR-50
Title	Click on add child record Button
Requirement	The user shall be able to click on add child record to add record of newly born child
Source	Hospital
Rationale	To enter record of newly born child into the system.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 87: Textual Description of FR-50

Identifier	FR-51
Title	Enter child details
Requirement	The user shall be able to enter details of newly born child
Source	Hospital
Rationale	To enter record of newly born child like date of birth, time, father CNIC etc.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Table 88: Textual Description of FR-51

Identifier	FR-52
Title	Submit child record
Requirement	The user shall be able to click on submit button to add record of newly born child
Source	Hospital
Rationale	To submit record into the database.
Business Rule (if required)	N/A
Dependencies	FR-49
Priority	High

Table 89: Textual Description of FR-52

Identifier	FR-53
Title	Click on update child record Button
Requirement	The user shall be able to click on update child record to update record of newly born child
Source	Hospital
Rationale	To update record of newly born child i.e name.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 90: Textual Description of FR-53

Identifier	FR-54
Title	Enter id of child
Requirement	The user shall be able to enter ID of child to update child record.
Source	Hospital
Rationale	To update record of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 91: Textual Description of FR-54

Identifier	FR-55
Title	Enter updated child details
Requirement	The user shall be able to enter updated details of a child
Source	Hospital
Rationale	To update record of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	medium

Table 92: Textual Description of FR-55

Identifier	FR-56
Title	Submit updated child record
Requirement	The user shall be able to click on update record button to update record of newly born child
Source	Hospital
Rationale	To update record of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 93: Textual Description of FR-56

Identifier	FR-57
Title	Click on print birth certificate button
Requirement	The user shall be able to click on print birth certificate button to print birth certificate of newly born child
Source	Hospital
Rationale	To print birth certificate of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	High

Table 94: Textual Description of FR-57

Identifier	FR-58
Title	Adjust print options
Requirement	The user shall be able to adjust print options
Source	Hospital
Rationale	To print birth certificate of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 95: Textual Description of FR-58

Identifier	FR-59
Title	Click on print
Requirement	The user shall be able to click on print button to print birth certificate.
Source	Hospital
Rationale	To print birth certificate of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	High

Table 96: Textual Description of FR-59

Identifier	FR-60
Title	Click on print vaccination dates button
Requirement	The user shall be able to click on print vaccination dates button to print auto generated vaccination dates of newly born child
Source	Hospital
Rationale	To print vaccination dates of newly born child.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	high

Table 97: Textual Description of FR-60

Identifier	FR-61
Title	Search child information
Requirement	The user shall be able to click on search child information to search child information by ID.
Source	Hospital
Rationale	To search specific child info by their ID.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 98: Textual Description of FR-61

Identifier	FR-62
Title	Check vaccine requirements
Requirement	The user shall be able to click on vaccine requirements to check future requirements of vaccine.
Source	Hospital, Vaccine center
Rationale	To check future vaccine requirements.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 99: Textual Description of FR-62

Identifier	FR-63
Title	Update child vaccination record
Requirement	The user shall be able to update child vaccination record from web application.
Source	Hospital, Vaccine center
Rationale	To update child vaccination record.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	High

Table 100: Textual Description of FR-63

Identifier	FR-64
Title	Send notifications
Requirement	The user shall be able to send notifications when he updates child vaccination record.
Source	Hospital, Vaccine center
Rationale	To send notification to parent.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 101: Textual Description of FR-64

Identifier	FR-65
Title	Receive confirmation
Requirement	The user shall be able to receive confirmation of vaccination from parent when he updates the child vaccination info.
Source	Hospital, Vaccine center
Rationale	To receive confirmation of vaccination from parent.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 102: Textual Description of FR-65

Identifier	FR-66
Title	Update status
Requirement	The user shall be able to update child vaccination status when the parents confirm vaccination.
Source	Hospital, Vaccine center
Rationale	To update child vaccination status when parent confirms vaccination.
Business Rule (if required)	N/A
Dependencies	FR-50
Priority	Low

Table 103: Textual Description of FR-66

Identifier	FR-67
Title	Report authorities
Requirement	The user shall be able to report authorities about polio cases.
Source	Hospital, Vaccine center
Rationale	To report authorities about polio cases.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 104: Textual Description of FR-67

Identifier	FR-68
Title	Manage vaccine stock
Requirement	The user shall be able to click on manage vaccine stock.
Source	Hospital, Vaccine center
Rationale	To manage vaccine stock.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 105: Textual Description of FR-68

Identifier	FR-69
Title	Add stock
Requirement	The user shall be able to click on Add vaccine stock
Source	Hospital, Vaccine center
Rationale	To add new vaccine stock
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 106: Textual Description of FR-69

Identifier	FR-70
Title	update stock
Requirement	The user shall be able to click on update vaccine stock
Source	Hospital, Vaccine center
Rationale	To update vaccine stock
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 107: Textual Description of FR-70

Identifier	FR-71
Title	View stock
Requirement	The user shall be able to click on view vaccine stock
Source	Hospital, Vaccine center
Rationale	To view vaccine stock
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 108: Textual Description of FR-71

Identifier	FR-72
Title	Request for stock
Requirement	The user shall be able to click on request more stock from the higher authorities
Source	Hospital
Rationale	To request for more vaccine stock
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 109: Textual Description of FR-72

Functional Requirements for Vaccination Center:

Identifier	FR-73
Title	Manage campaigns
Requirement	The user shall be able to click on manage vaccine campaigns to manage new campaigns
Source	Vaccine center
Rationale	To manage campaigns
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 110: Textual Description of FR-73

Identifier	FR-74
Title	Make new campaigns
Requirement	The user shall be able to click on make new campaign to initiate new campaign
Source	Vaccine center
Rationale	To make new campaigns
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 111: Textual Description of FR-74

Identifier	FR-75
Title	View campaign status
Requirement	The user shall be able to click on any campaign to view their status
Source	Vaccine center
Rationale	To view existing campaign status
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 112: Textual Description of FR-75

Identifier	FR-76
Title	Allot workers for campaign
Requirement	The user shall be able allot worker for campaigns
Source	Vaccine center
Rationale	To allot worker for new campaigns
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 113: Textual Description of FR-76

Identifier	FR-77
Title	Allot vaccine for campaign
Requirement	The user shall be able allot vaccine for campaigns
Source	Vaccine center
Rationale	To allot vaccine for new campaigns
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 114: Textual Description of FR-77

Identifier	FR-78
Title	Notify public
Requirement	The user shall be able to notify public by sending them text messages
Source	Vaccine center
Rationale	To notify public by sending text messages to already registered users.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	Low

Table 115: Textual Description of FR-78

3.4 Non-Functional Requirements

Non-Functional requirements are required to be explicitly mention in this section.

3.4.1 Reliability

REA-1: Parent and child data should be added accurately as whole system is build on this single thing

REA-2: Admin approval is important for all the work done

3.4.2 Usability

USE-1: The system must allow user to see his child vaccine data with only one click

USE-2: The system shall provide information for conforming notification

USE-3: The system shall work in those areas too where internet facility is not good

3.4.3 Accessibility

ACC-1- It must be accessible from remote areas. It must be accessible through mobile and web efficiently

3.4.4 Compatible

COM-1- The system shall be compatible with both Android and iOS. Web should be easily compatible on all major browsers

3.4.5 Security

SEC-1: Security in this system should be very high as it contains sensitive information.

4 Design and Architecture.

4.1 System Architecture

4.1.1 Three Tier Architecture Description

4.1.1.1 Tier-1

This is the front end of the system where user can access all the functionalities using the interface. It is usually called as Presentation Layer. This layer is implemented in React and React Native.

4.1.1.2 Tier-2:

This is the backend of the system where all the application logic is implemented. It is a server which make communication between tier 1 and tier 3. This layer is called as Business Layer. It is implemented in Node JS.

4.1.1.3 Tier-3:

This is the database of the system where all the application data will be stored. This layer is called as Data Layer. It is implemented Mongo DB.

4.1.2 3 Tier Architecture

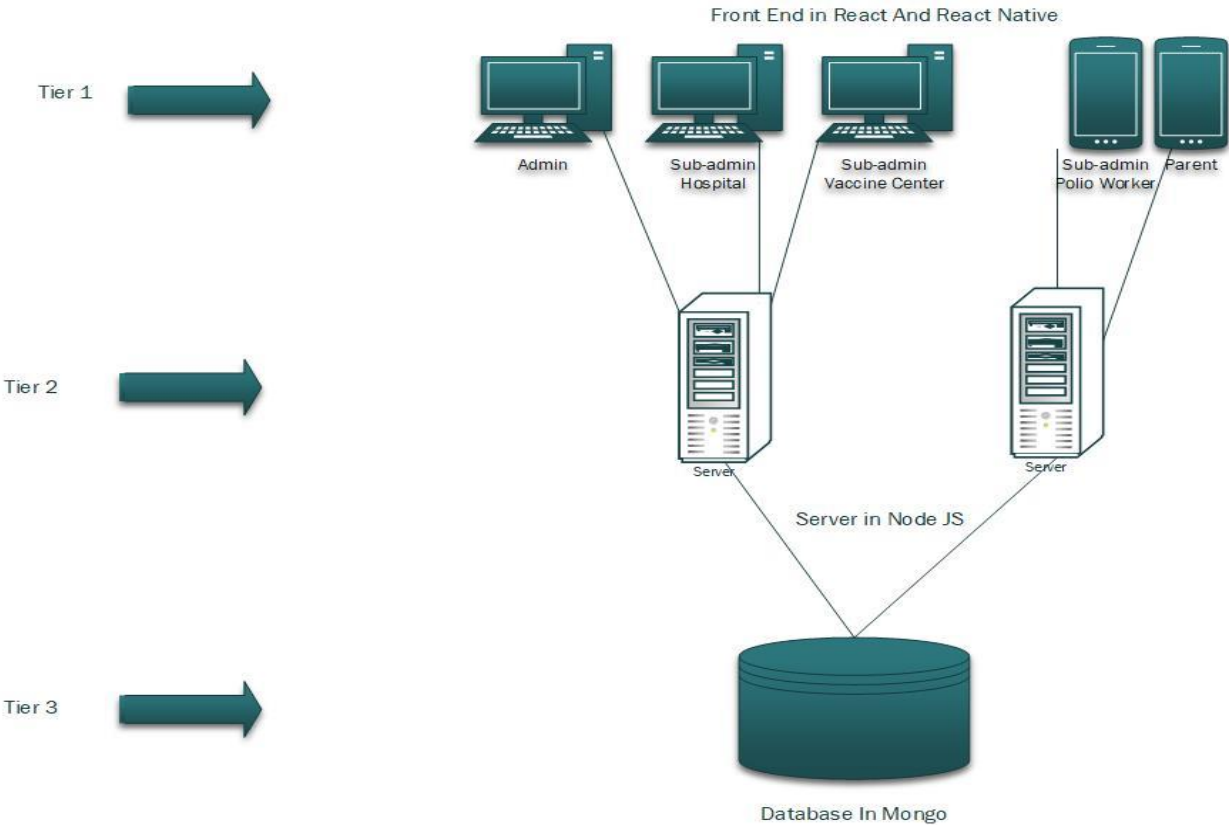


Figure 6: 3 Tier Architecture Diagram

4.2 Data Representation

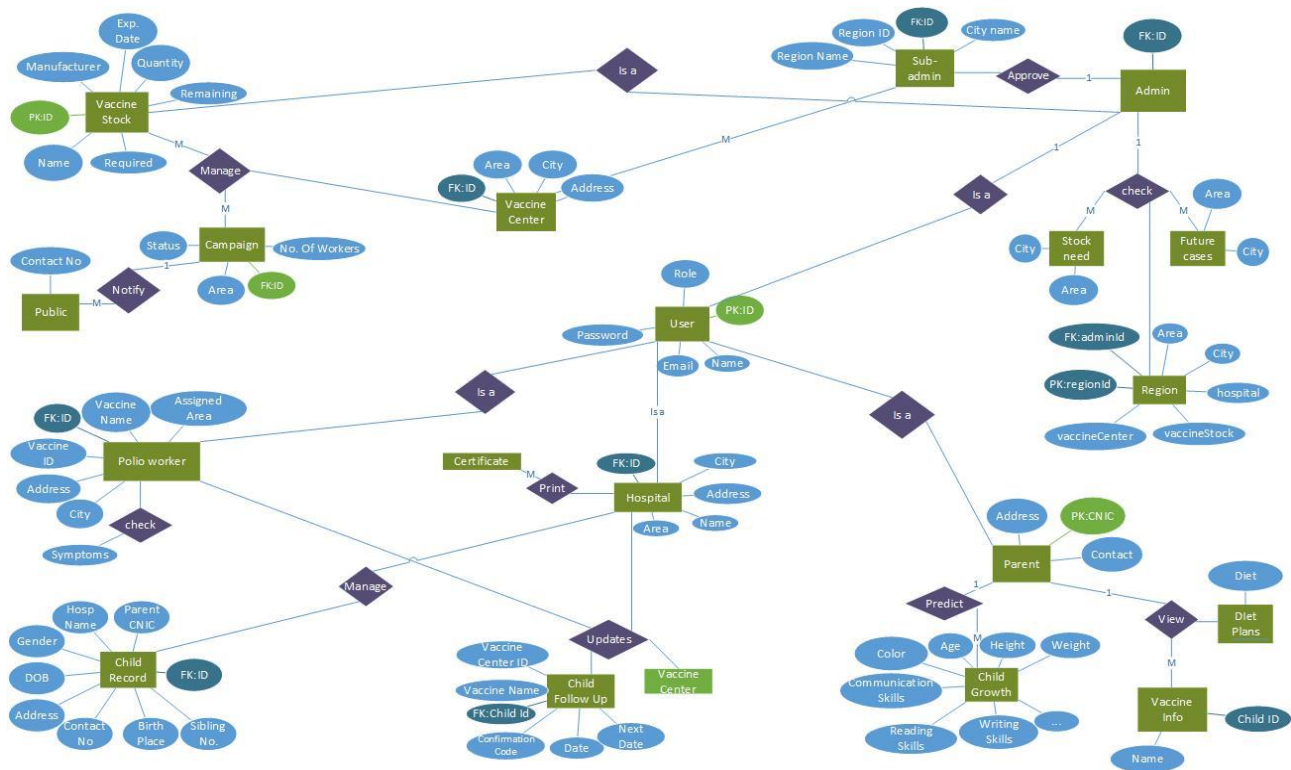


Figure 7: ERD

4.2.1 Vaccine Schema

```
const vaccineSchema = new schema({
  name: { type: String, default: "", required: true },
  manufacturer: { type: String, default: "", required: true },
  quantity: { type: Number, default: 50, required: true },
  expiryDate: { type: Date, default: Date.now, required: true },
});
```

4.2.2 Assign Vaccination Schema

```
const organizationVaccinesSchema = new schema({
  organization: { type: schema.Types.ObjectId, required: true, default: "", ref:
"userAccounts" },
  vaccines: {
    polio: { quantity: Number, default: 0 },
    diphtheria: { quantity: Number, default: 0 },
    homophiles: { quantity: Number, default: 0 },
    rotaVirus: { quantity: Number, default: 0 },
    measles: { quantity: Number, default: 0 },
    hepatitisA: { quantity: Number, default: 0 },
    hepatitisB: { quantity: Number, default: 0 },
  }
});
```

```

        papillomaVirus: { quantity: Number, default: 0 },
        influenza: { quantity: Number, default: 0 },
    },
});

```

4.2.3 Daily Consumption Schema

```

const dailyConsumptionSchema = new schema({
    vaccineName: { type: schema.Types.ObjectId, required: true, ref: "Vaccine" },
    date: { type: Date, default: Date.now(), required: true },
    gender: { type: String, default: "", enum: ["male", "female"] },
    region: { type: String, default: "" },
    organization: { type: schema.Types.ObjectId, required: true, ref: "userAccounts" },
});

```

4.2.4 Child Follow up Schema

```

const childFollowUpSchema = new schema({
    child: { type: schema.Types.ObjectId, required: true, ref: "Child" },
    date: { type: Date, default: Date.now() },
    nextDate: { type: Date, default: Date.now() },
    vaccineName: { type: schema.Types.ObjectId, required: true, ref: "Vaccine" },
    organization: { type: schema.Types.ObjectId, required: true, ref: "userAccounts" },
    confirmationCode: { type: String, default: "" },
});

```

4.2.5 Child Schema

```

const childrenSchema = new schema({
    childID: { type: String, required: true, default: "" },
    parentName: { type: String, required: true, default: "" },
    parentCNIC: { type: String, required: true, default: "" },
    contactNo: { type: String, required: true, default: "" },
    emergencyContact: { type: String, required: true, default: "" },
    address: {
        addr: { type: String, default: "", required: true },
        area: { type: String, default: "", required: true },
        city: { type: String, default: "", required: true },
    },
    dateOfBirth: { type: Date, required: true, default: Date.now() },
    gender: { type: String, required: true, default: "" },
    birthPlace: { type: String, required: true, default: "" },
    siblingNo: { type: Number, required: true, default: 1 },
    hospitalName: { type: schema.Types.ObjectId, required: true, ref: "userAccounts" },
    vaccination: [
        {

```

```

        polio: { noOfDoses: { type: Number, default: 0 } },
        diphtheria: { noOfDoses: { type: Number, default: 0 } },
        homophiles: { noOfDoses: { type: Number, default: 0 } },
        rotaVirus: { noOfDoses: { type: Number, default: 0 } },
        measles: { noOfDoses: { type: Number, default: 0 } },
        hepatitisA: { noOfDoses: { type: Number, default: 0 } },
        hepatitisB: { noOfDoses: { type: Number, default: 0 } },
        papillomaVirus: { noOfDoses: { type: Number, default: 0 } },
        influenza: { noOfDoses: { type: Number, default: 0 } },
    },
],
});

```

4.2.6 Campaign Schema

```

const campaignSchema = new schema({
    campaignID: { type: String, default: "", required: true },
    vaccineCenter: { type: schema.Types.ObjectId, required: true, ref: "userAccounts" },
    status: { type: String, default: "active", required: true, enum: ["active", "inactive"] },
    area: { type: String, default: "", required: true },
    noOfWorkers: { type: String, default: "5", required: true },
    startDate: { type: Date, default: Date.now() },
    endDate: { type: Date, default: Date.now() },
});

```

4.2.7 Assign Vaccine Schema

```

const assignVaccineToSchema = new schema({
    vaccine: { type: schema.Types.ObjectId, required: true, ref: "Vaccine" },
    date: { type: Date, default: Date.now(), required: true },
    quantity: { type: Number, default: "" },
    organization: { type: schema.Types.ObjectId, required: true, ref: "userAccounts" },
});

```

4.3 Process Flow/Representation

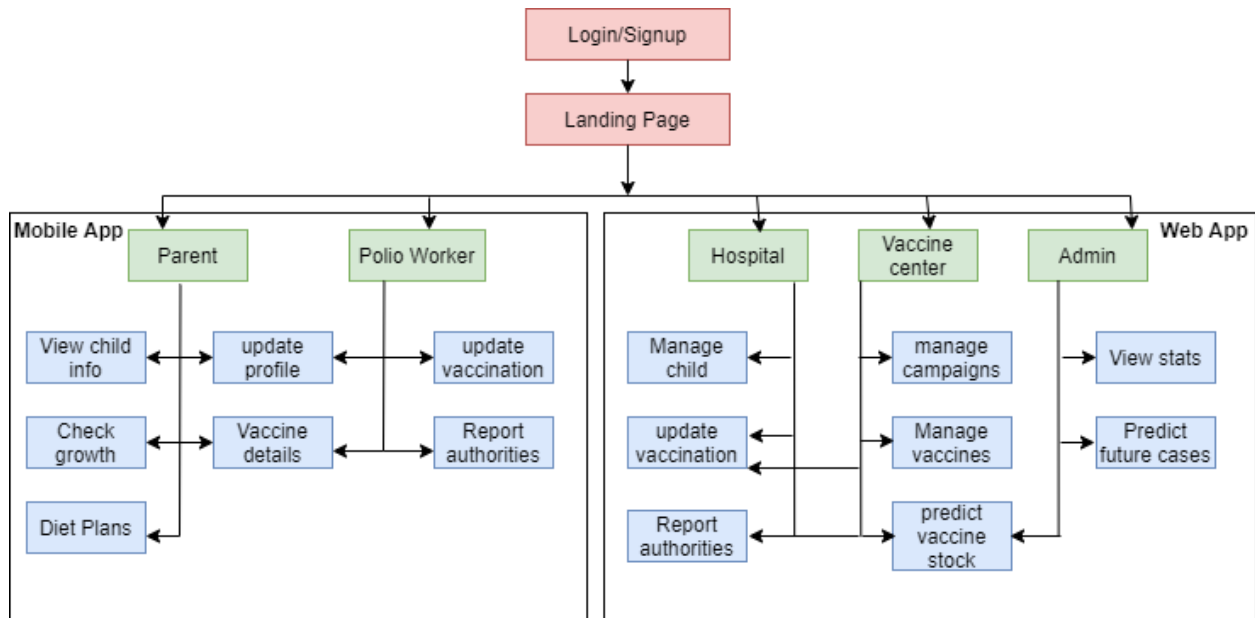


Figure 8: Process Flow

4.4 Design Models

4.4.1 Structural Diagrams

4.4.1.1 Class diagram Web



Figure 9: Web Class Diagram

4.4.1.2 Mobile Class Diagram

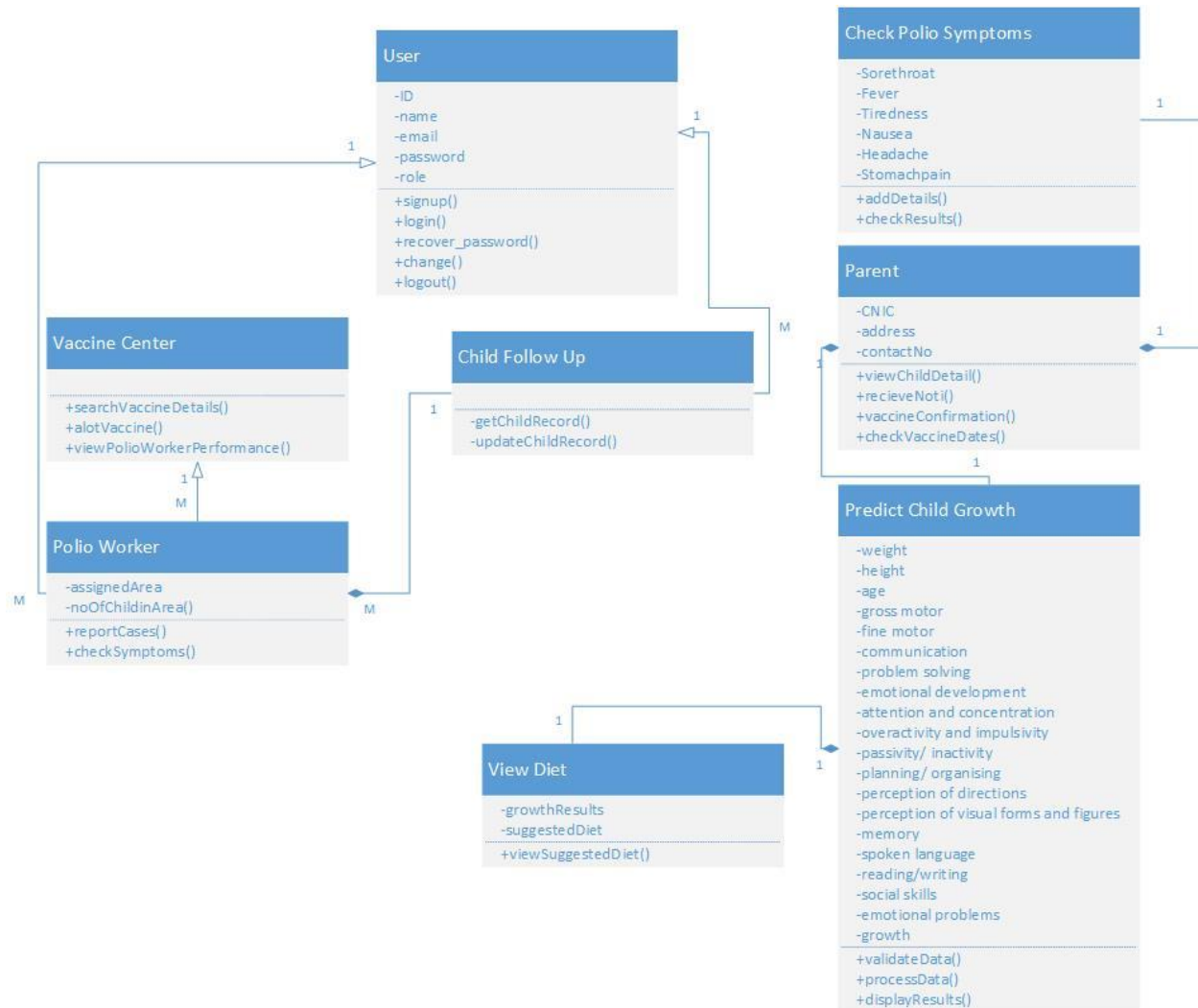


Figure 10: Mobile Class Diagram

4.4.2 Behavioral Diagrams

4.4.2.1 Activity diagram

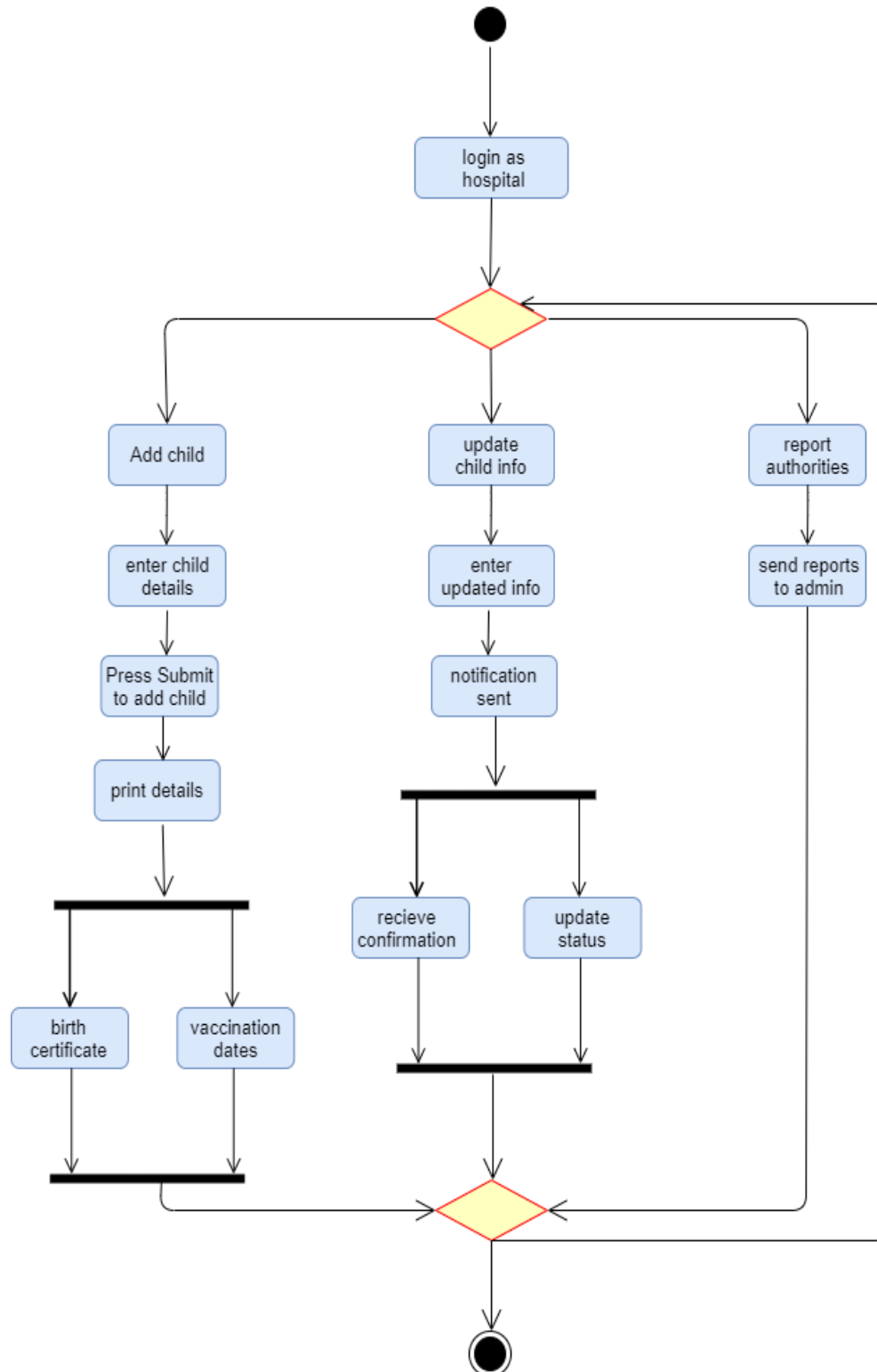


Figure 11: Hospital (sub admin) activity 1

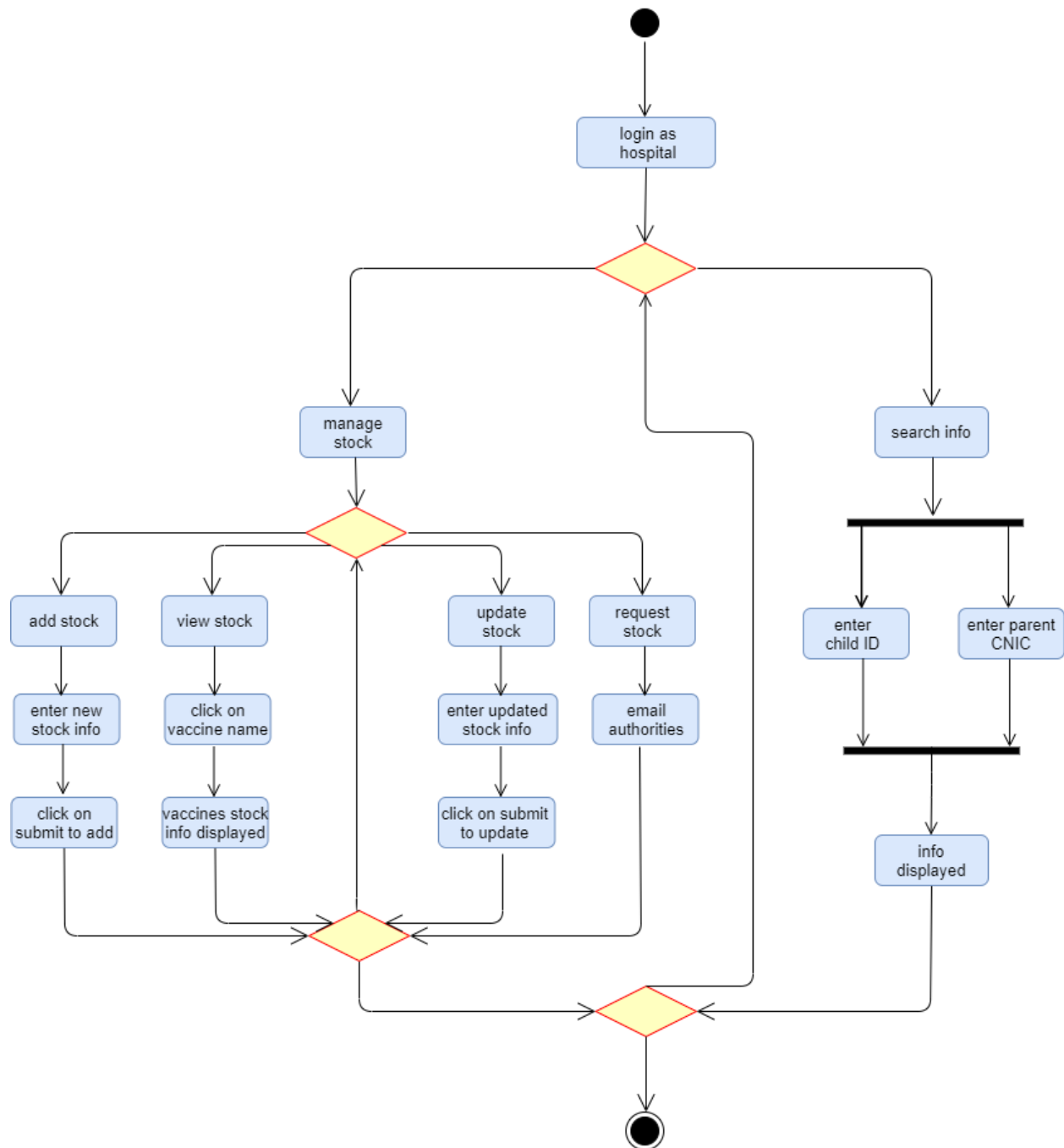


Figure 12: Hospital Activity 2

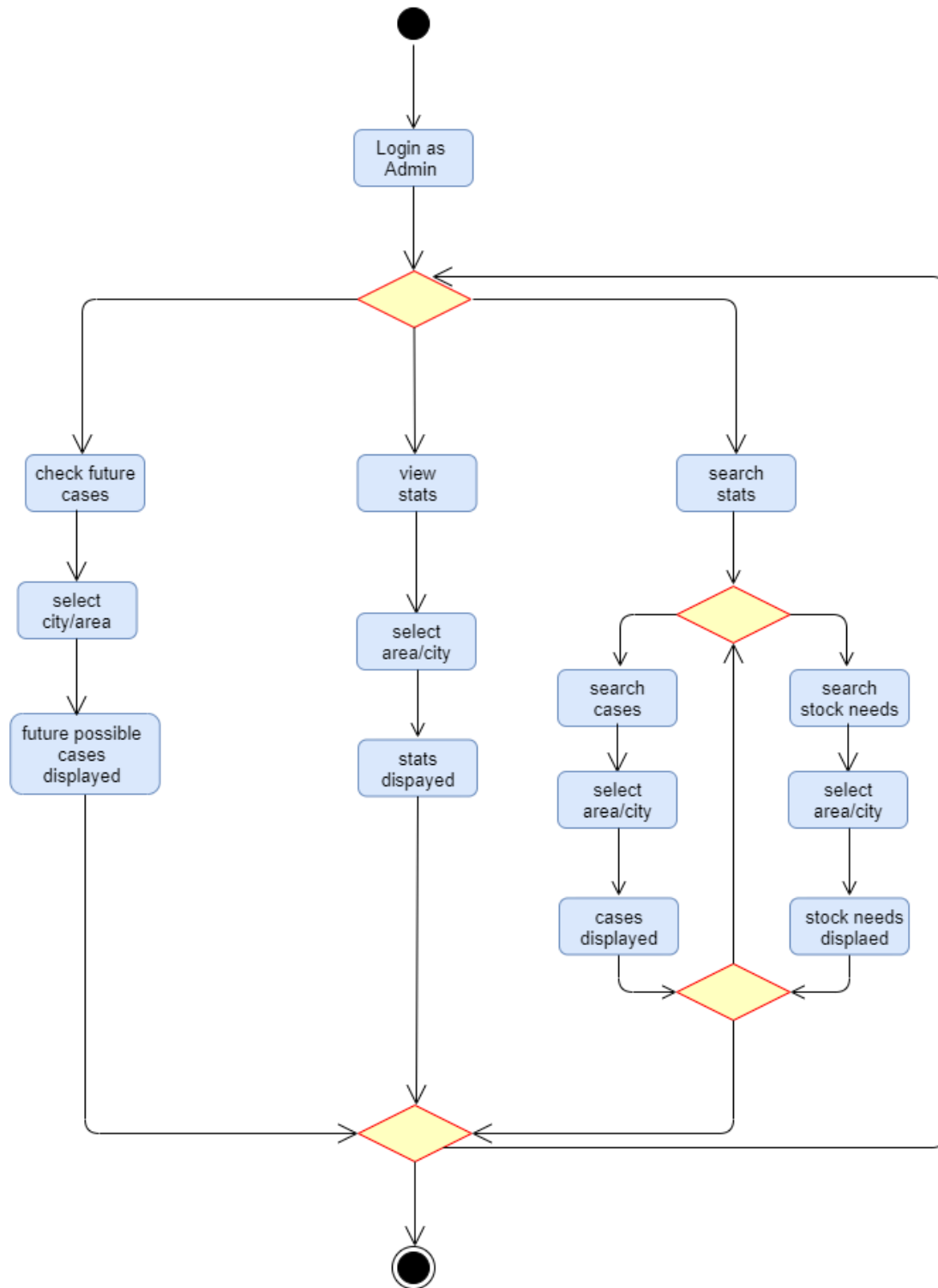


Figure 13: Admin Activity 1

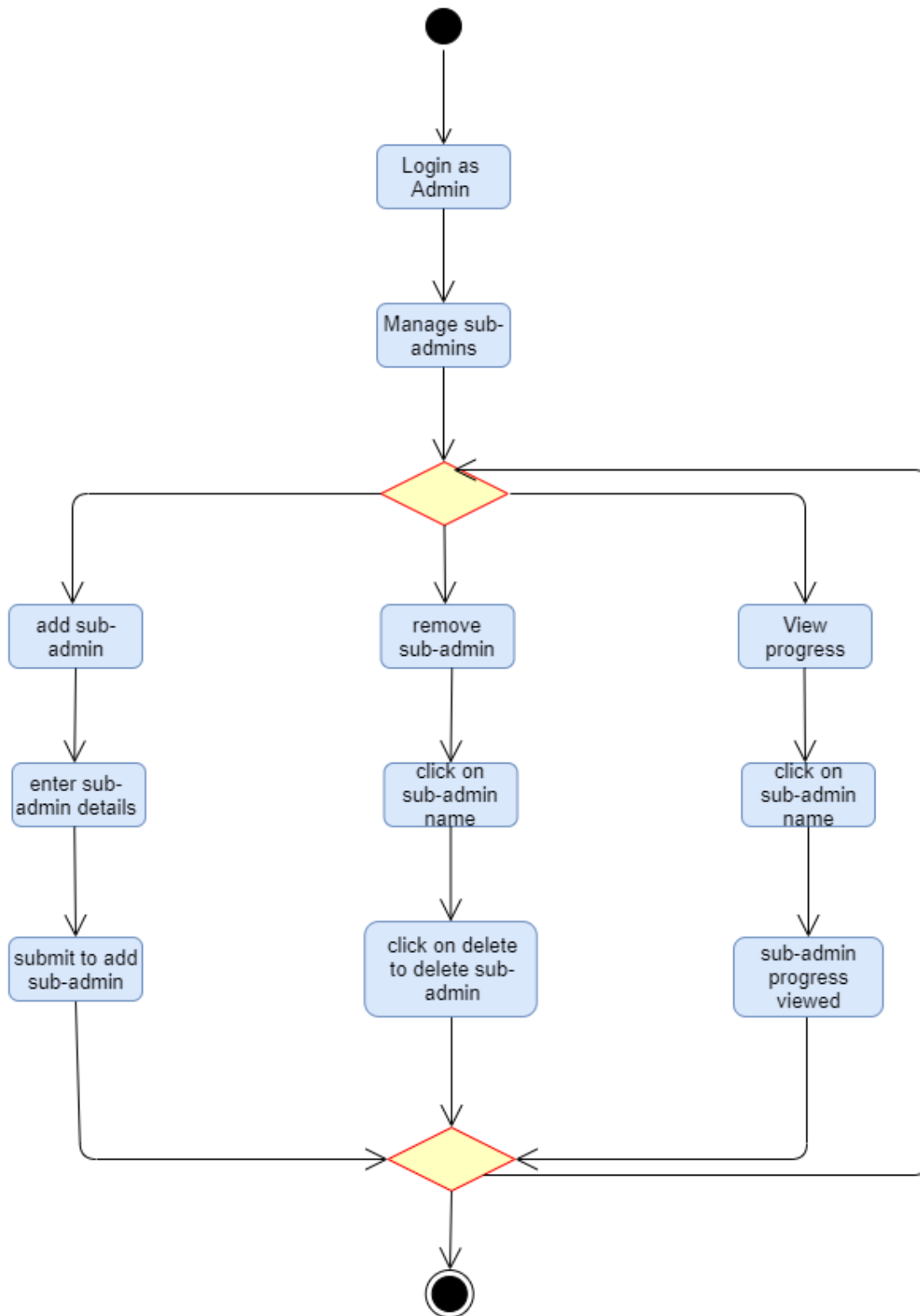


Figure 14: Admin Activity 2

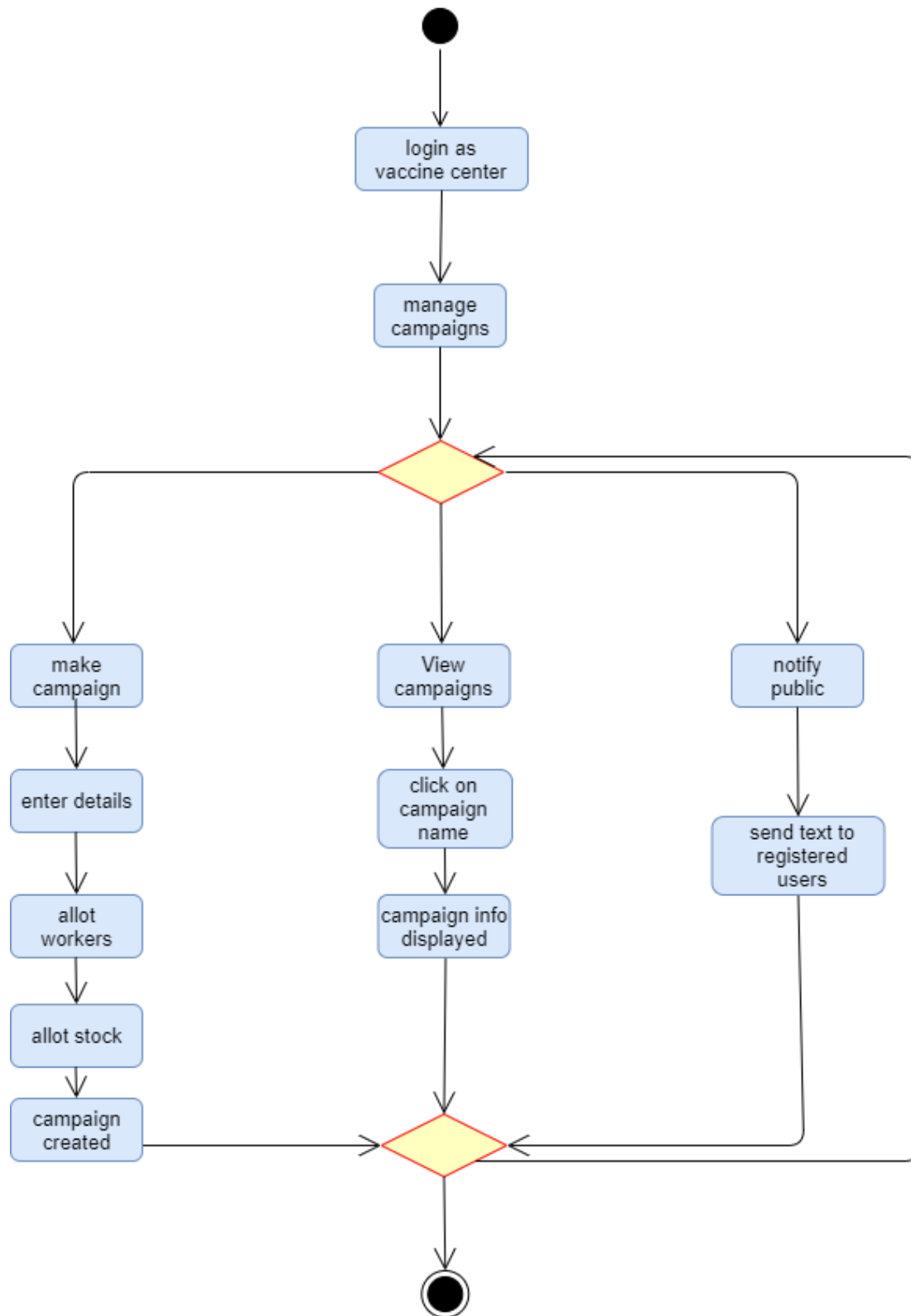


Figure 15: Vaccine Center Activity 1

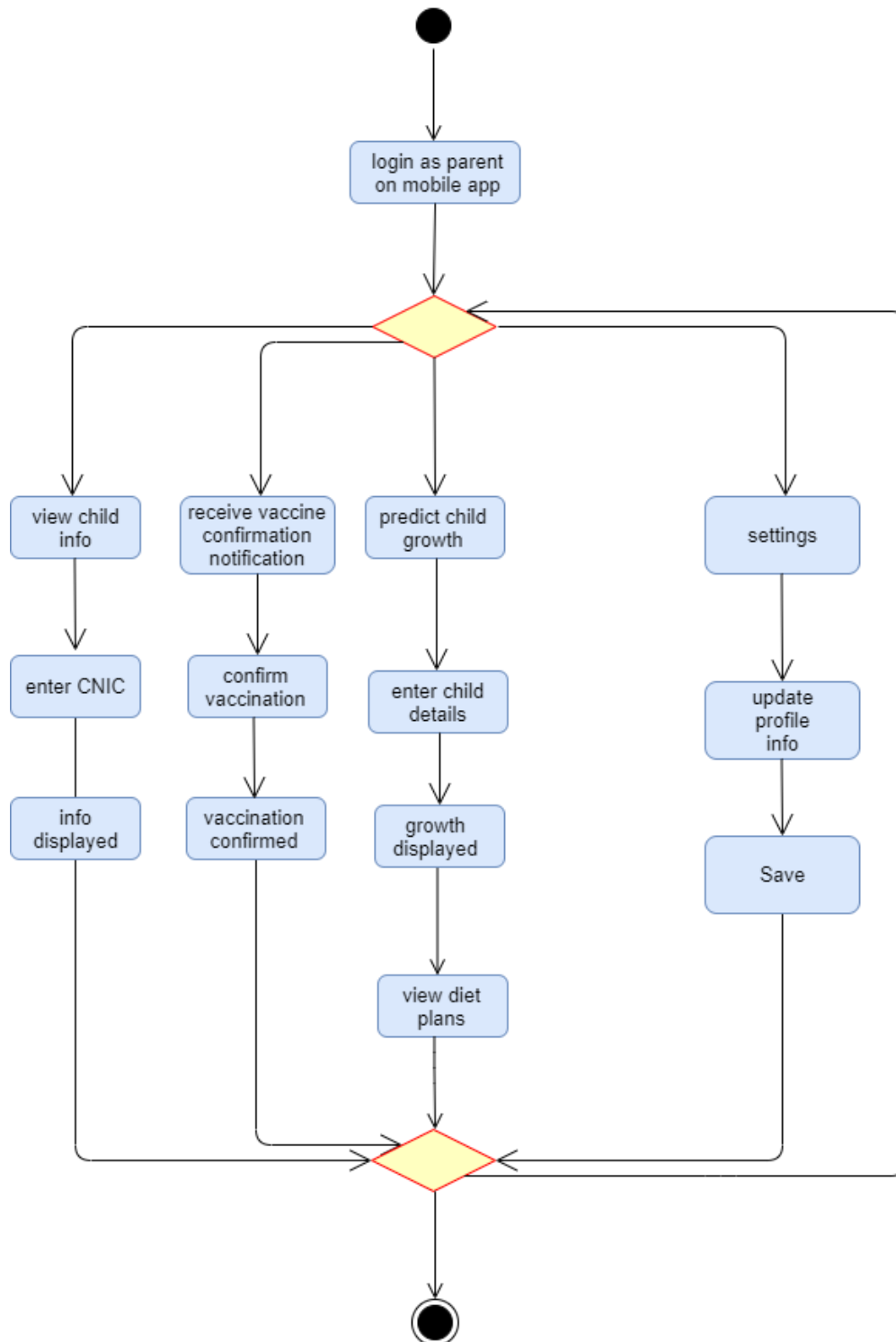


Figure 16:Parent Activity 1

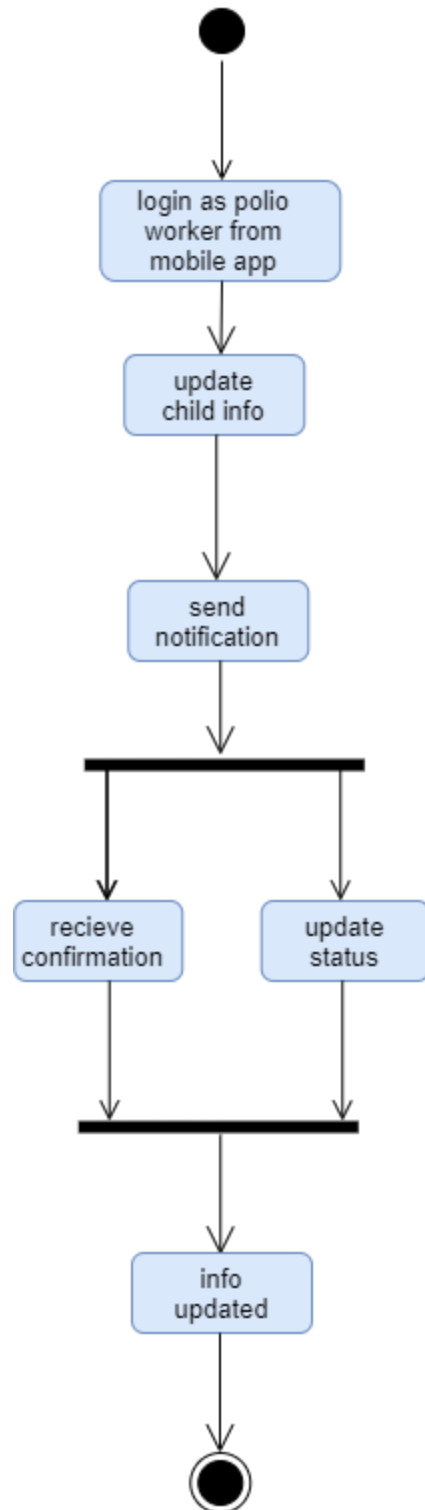


Figure 17: Polio Worker activity

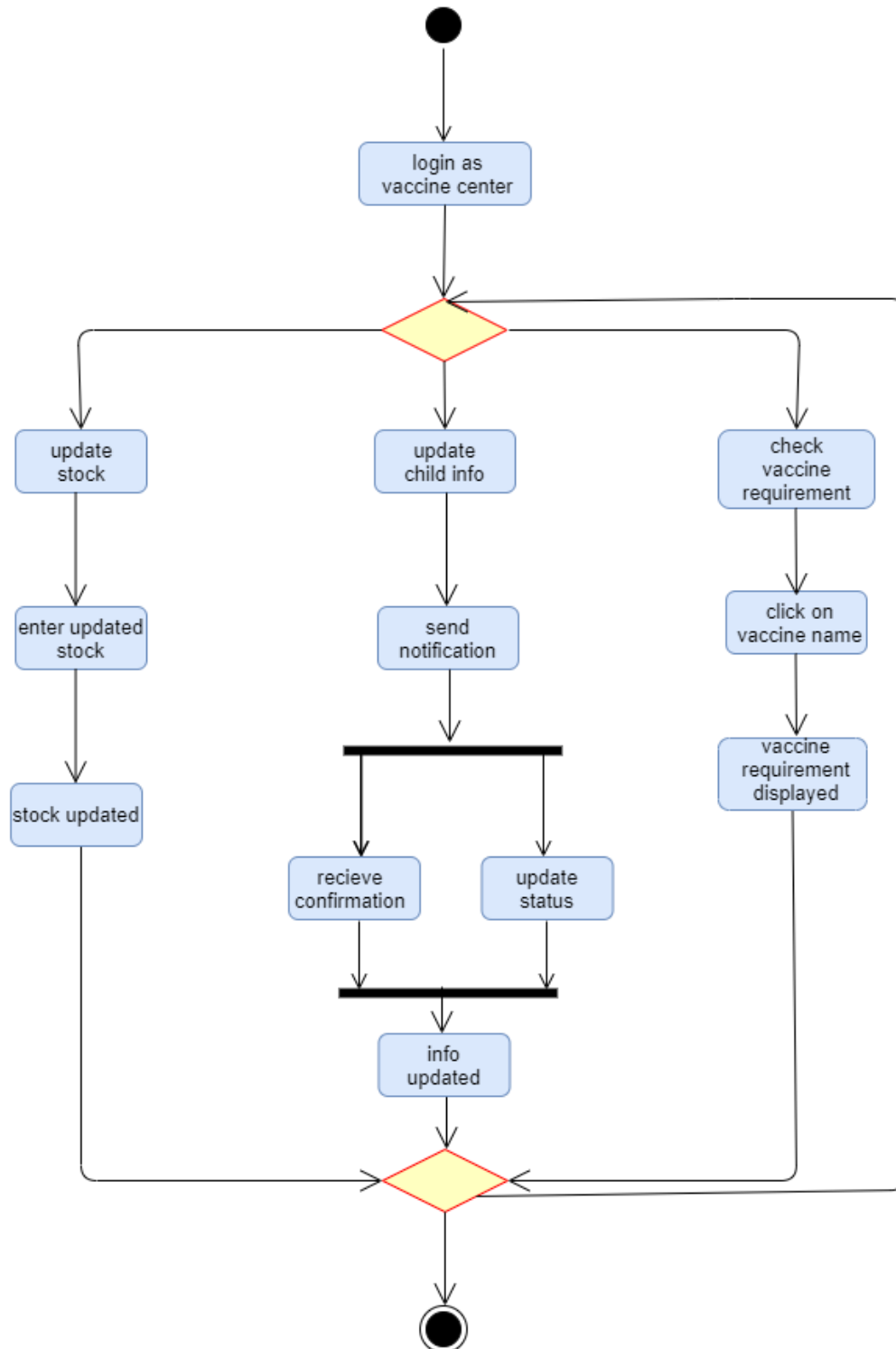


Figure 18:Vaccine Center Activity 3

4.4.2.2 Sequence diagram

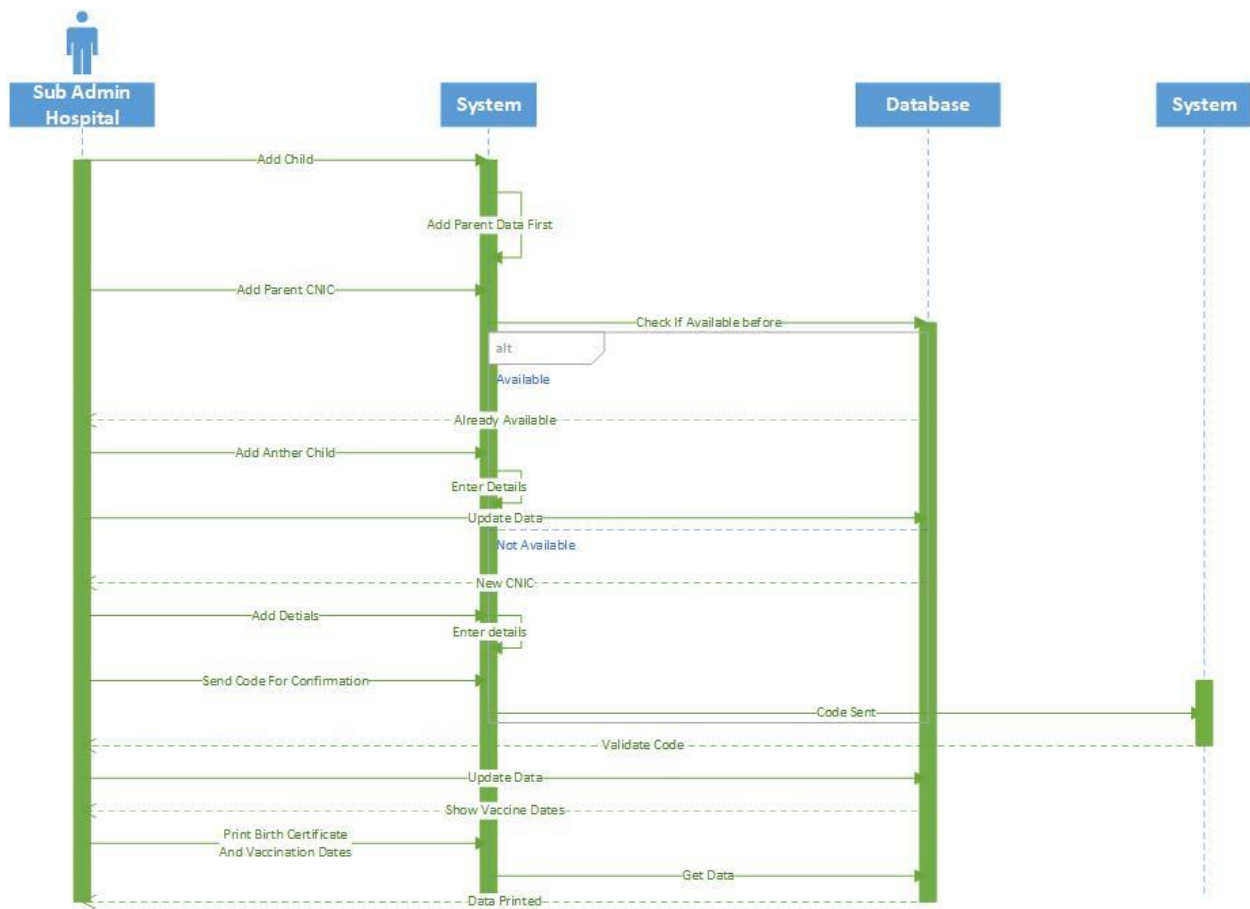


Figure 19:Sequence Diagram For Add Child Data

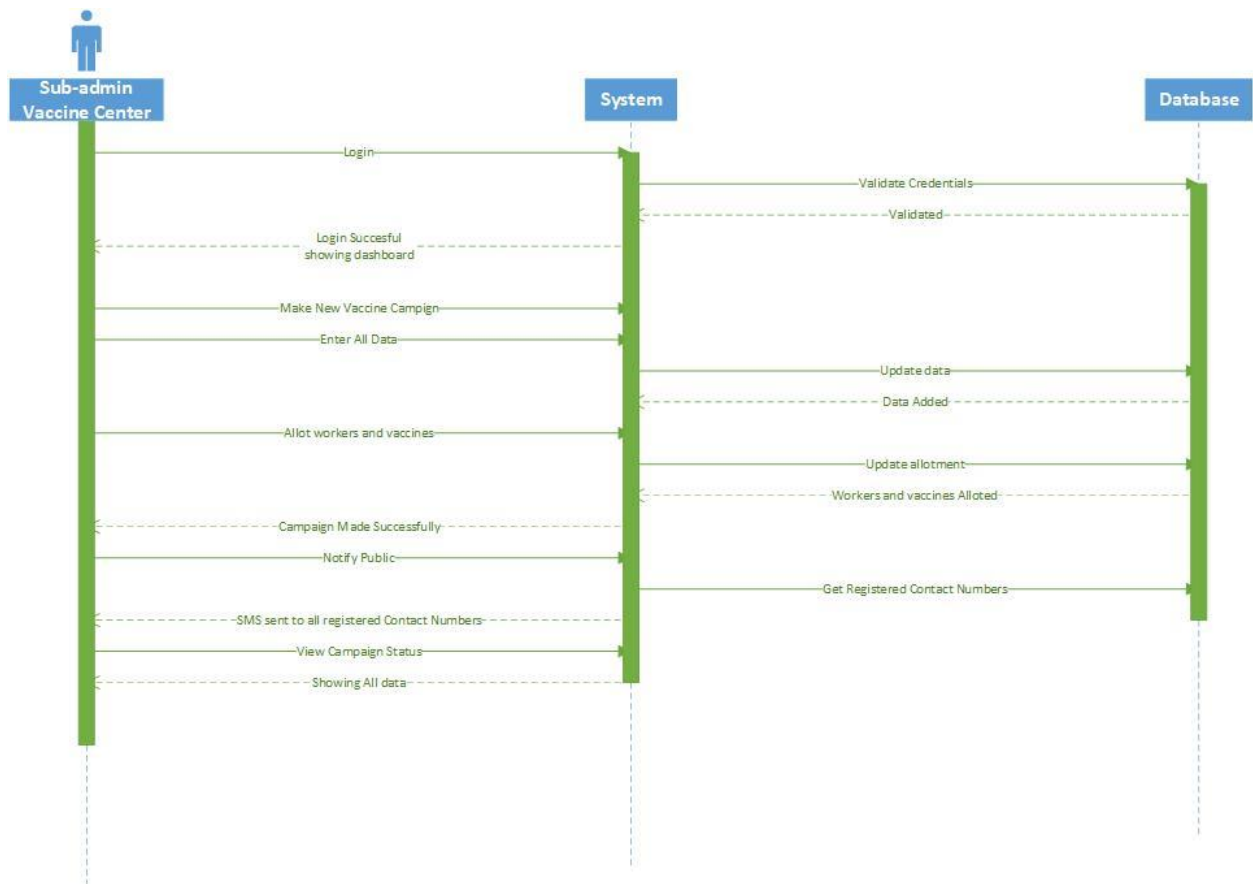


Figure 20:Sequence Diagram For Campaign Creation

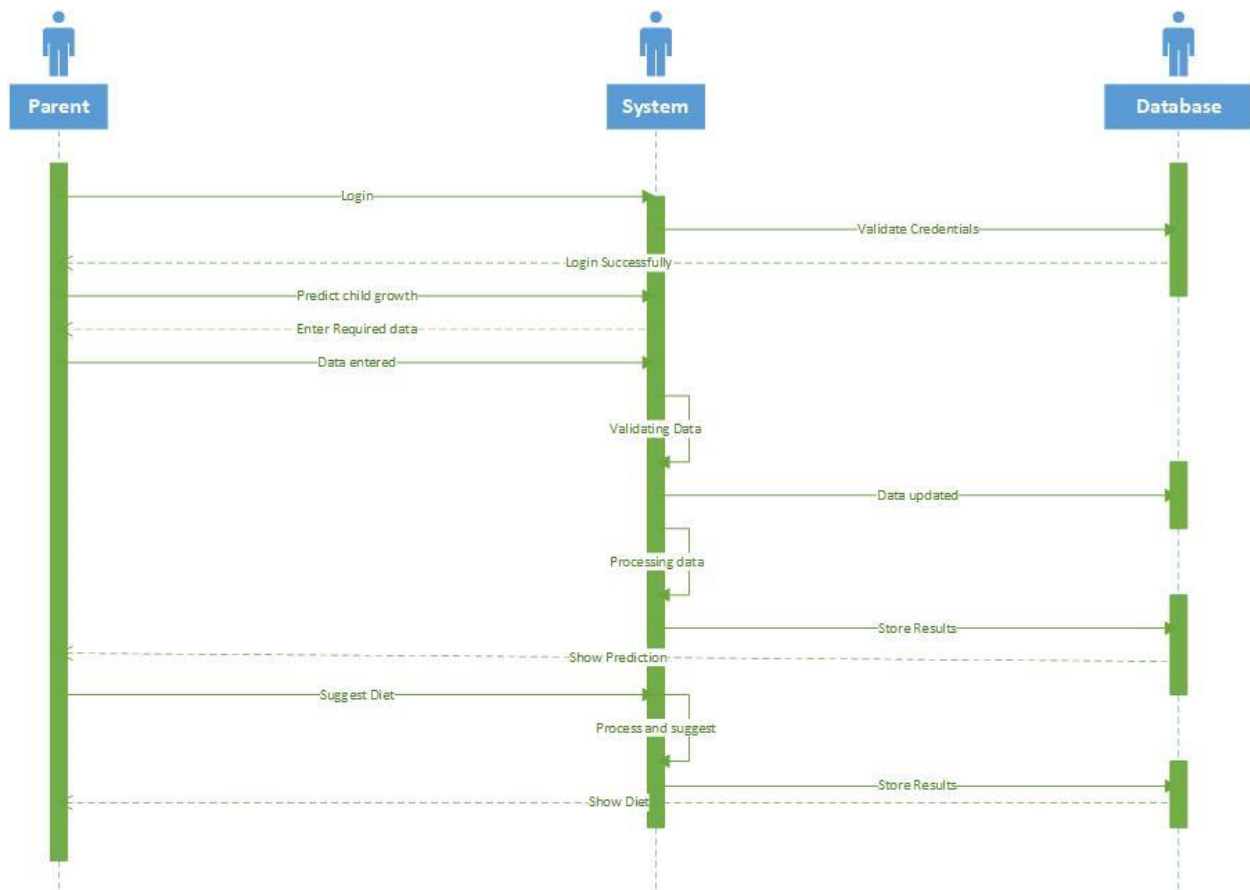


Figure 21: Sequence Diagram For Child Growth Prediction

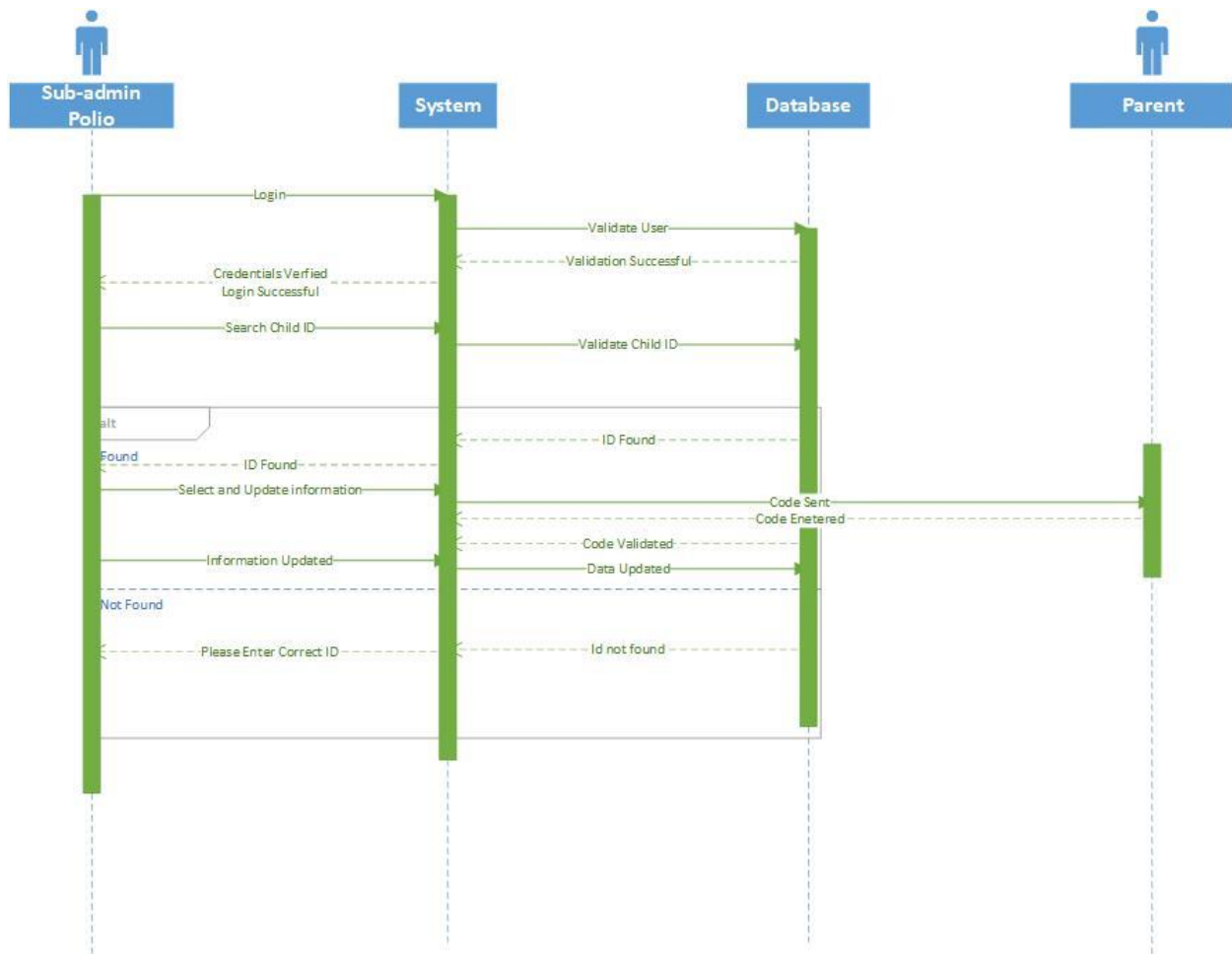


Figure 22: Sequence Diagram For Child Vaccine Update

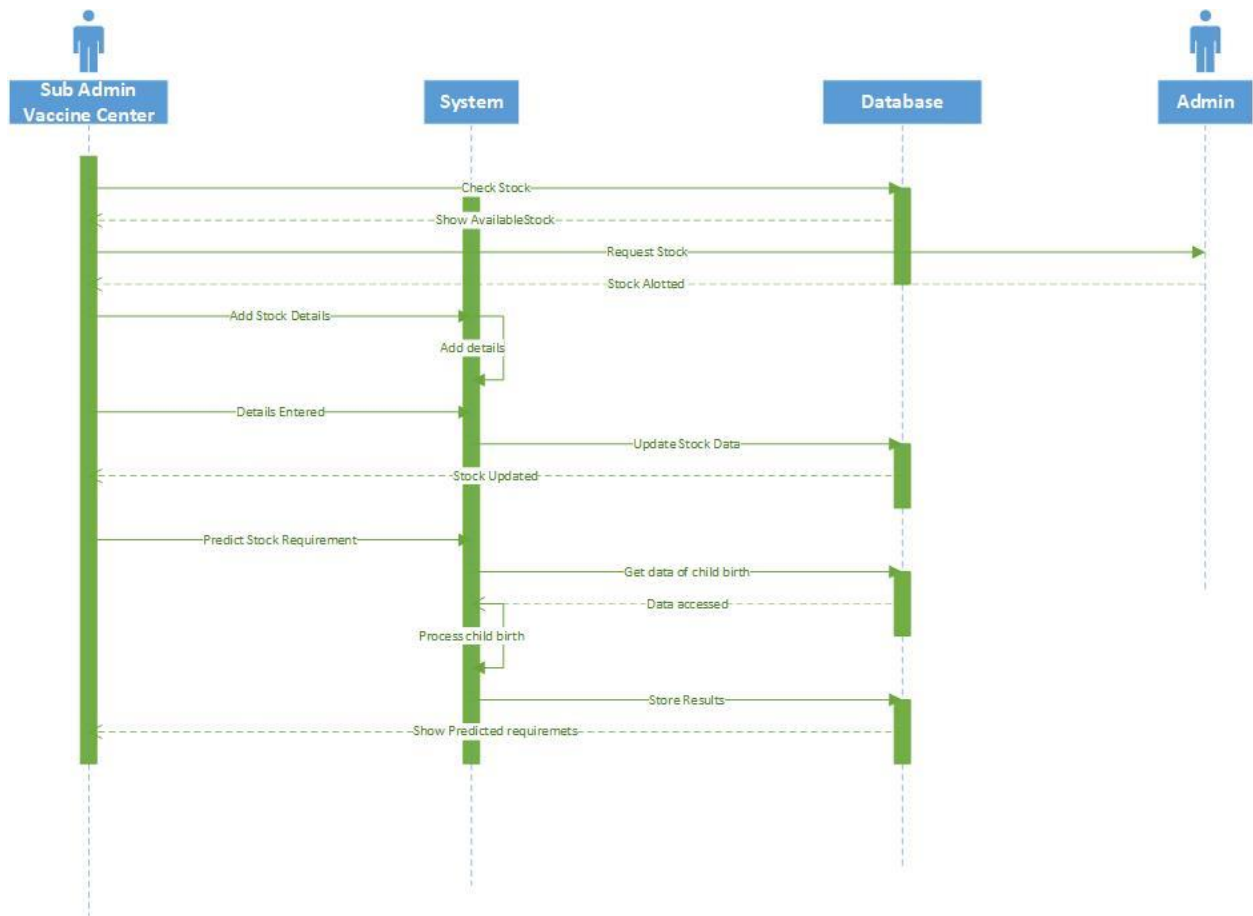


Figure 23: Sequence Diagram For Predict Vaccine Stock

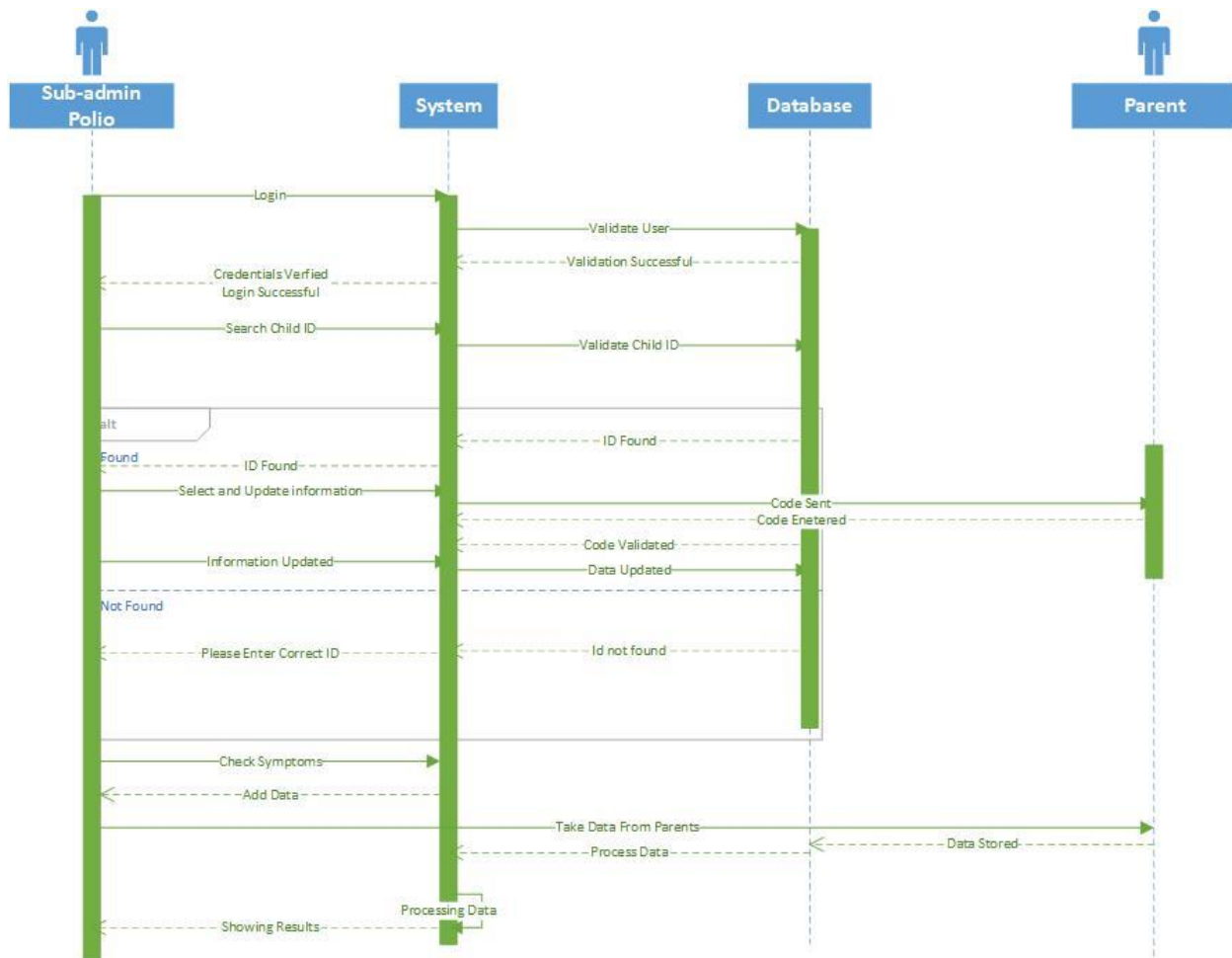


Figure 24: Sequence Diagram For Polio Vaccine Update

5 Implementation

This chapter will discuss implementation details supported by UML diagrams (if applicable). You will not put your source code here. Any of the following sections may be included based on your project.

5.1 Algorithm

Mention the algorithm(s) used in your project to get the work done with regards to major modules. Provide a pseudocode **OR** a natural language explanation regarding the functioning of main features. Be sure to use the correct syntax and semantics for algorithm representations.

5.2 Implementation

5.2.1 Server

```
const express = require("express"),
    cors = require("cors"),
    mongoose = require("mongoose");

require("dotenv").config();

const app = express();
const PORT = process.env.PORT || 5000;

app.use(cors());
app.use(express.json());

// db connection
const ATLAS_URI = process.env.ATLAS_URI;
mongoose.connect(ATLAS_URI, {
    useNewUrlParser: true,
    useCreateIndex: true,
    useUnifiedTopology: true,
});

const connection = mongoose.connection;
connection.once("open", () => {
    console.log(`Mongodb connected successfully`);
});
```

```
// routers

const childRouter = require("./routes/children");
const vaccineRouter = require("./routes/vaccines");
const campaignRouter = require("./routes/campaigns");

app.use("/children", childRouter);
app.use("/vaccines", vaccineRouter);
app.use("/campaigns", campaignRouter);

// start server
app.listen(PORT, () => {
  console.log(`Server is running at ${PORT}`);
});
```

5.2.2 Campaign Routing

```
const router = require("express").Router();
let Campaign = require("../models/Campaign.model");

// get Campaigns
router.route("/").get((req, res) => {
  Campaign.find()
    .then((campaign) => res.json(campaign))
    .catch((err) => res.status(400).json({ error: err }));
});

// get single Campaign
router.route("/:id").get((req, res) => {
  Campaign.findById(req.params.id)
    .then((campaign) => res.json(campaign))
```

```

        .catch((err) => res.status(400).json({ error: err }));
    });

// delete Campaign
router.route("/:id").delete((req, res) => {
    Campaign.findByIdAndDelete(req.params.id)
        .then((campaign) => res.json("Campaign deleted "))
        .catch((err) => res.status(400).json({ error: err }));
});

// add Campaign
router.route("/add").post((req, res) => {
    const campaignID = req.body.campaignID;
    const status = req.body.status;
    const area = req.body.area;
    const noOfWorkers = Number(req.body.noOfWorkers);
    const startDate = Date.parse(req.body.startDate);
    const endDate = Date.parse(req.body.endDate);

    const newCampaign = new Campaign({
        campaignID,
        status,
        area,
        noOfWorkers,
        startDate,
        endDate,
    });
});

```

```

    newCampaign
      .save()
      .then(() => res.json("Campaign added!"))
      .catch((err) => res.status(400).json({ error: err }));
  });

// update Campaign
router.route("/update/:id").post((req, res) => {
  Campaign.findById(req.params.id)
    .then((campaign) => {
      campaign.campaignID = req.body.campaignID;
      campaign.status = req.body.status;
      campaign.area = req.body.area;
      campaign.noOfWorkers = Number(req.body.noOfWorkers);
      campaign.startDate = Date.parse(req.body.startDate);
      campaign.endDate = Date.parse(req.body.endDate);

      campaign
        .save()
        .then(() => res.json("Campaign updated!"))
        .catch((err) => res.status(400).json({ error: err }));
    })
    .catch((err) => res.status(400).json({ error: err }));
});

module.exports = router;

```

5.2.3 Child Routing

```
const router = require("express").Router();
```

```

let Child = require("../models/child.model");

// get children
router.route("/").get((req, res) => {

    Child.find()

        .then((children) => res.json(children))

        .catch((err) => res.status(400).json({ error: err }));

});

// get single child
router.route("/:id").get((req, res) => {

    Child.findById(req.params.id)

        .then((children) => res.json(children))

        .catch((err) => res.status(400).json({ error: err }));

});

// delete child
router.route("/:id").delete((req, res) => {

    Child.findByIdAndDelete(req.params.id)

        .then((children) => res.json("Child deleted "))

        .catch((err) => res.status(400).json({ error: err }));

});

// add child
router.route("/add").post((req, res) => {

    const childID = req.body.childID;

    const parentName = req.body.parentName;

    const parentCNIC = req.body.parentCNIC;

```

```

const contactNo = req.body.contactNo;

const address = req.body.address;

const dateOfBirth = Date.parse(req.body.dateOfBirth);

const gender = req.body.gender;

const birthPlace = req.body.birthPlace;

const siblingNo = Number(req.body.siblingNo);

const hospitalName = req.body.hospitalName;

const vaccinationInfo = req.body.vaccinationInfo;


const newChild = new Child({
  childID,
  parentName,
  parentCNIC,
  contactNo,
  address,
  dateOfBirth,
  gender,
  birthPlace,
  siblingNo,
  hospitalName,
  vaccinationInfo,
});

newChild
  .save()
  .then(() => res.json("Child added!"))
  .catch((err) => res.status(400).json({ error: err }));
});

```

```

// update child

router.route("/update/:id").post((req, res) => {

  Child.findById(req.params.id)

    .then((child) => {

      child.childID = req.params.childID;

      child.parentName = req.body.parentName;

      child.parentCNIC = req.body.parentCNIC;

      child.childID = req.body.childID;

      child.contactNo = req.body.contactNo;

      child.address = req.body.address;

      child.dateOfBirth = Date.parse(req.body.dateOfBirth);

      child.gender = req.body.gender;

      child.birthPlace = req.body.birthPlace;

      child.siblingNo = Number(req.body.siblingNo);

      child.hospitalName = req.body.hospitalName;

      child.vaccinationInfo = req.body.vaccinationInfo;

      child

        .save()

        .then(() => res.json("Child updated!"))

        .catch((err) => res.status(400).json({ error: err }));

    })

    .catch((err) => res.status(400).json({ error: err }));

});

module.exports = router;

```

5.2.4 Vaccine Routing

```
const router = require("express").Router();

let Vaccine = require("../models/Vaccine.model");

// get Vaccines

router.route("/").get((req, res) => {

  Vaccine.find()

    .then((vaccine) => res.json(vaccine))

    .catch((err) => res.status(400).json({ error: err }));

});

// get single Vaccine

router.route("/:id").get((req, res) => {

  Vaccine.findById(req.params.id)

    .then((vaccine) => res.json(vaccine))

    .catch((err) => res.status(400).json({ error: err }));

});

// delete Vaccine

router.route("/:id").delete((req, res) => {

  Vaccine.findByIdAndDelete(req.params.id)

    .then((vaccine) => res.json("Vaccine deleted "))

    .catch((err) => res.status(400).json({ error: err }));

});

// add Vaccine

router.route("/add").post((req, res) => {

  const vaccineID = req.body.vaccineID;
```



```

const name = req.body.name;

const manufacturer = req.body.manufacturer;

const quantity = Number(req.body.quantity);

const expiryDate = Date.parse(req.body.expiryDate);

const newVaccine = new Vaccine({

  vaccineID,

  name,

  manufacturer,

  quantity,

  expiryDate,

});

newVaccine

  .save()

  .then(() => res.json("Vaccine added!"))

  .catch((err) => res.status(400).json({ error: err }));

});

// update Vaccine

router.route("/update/:id").post((req, res) => {

  Vaccine.findById(req.params.id)

    .then((vaccine) => {

      vaccine.vaccineID = req.body.vaccineID;

      vaccine.name = req.body.name;

      vaccine.manufacturer = req.body.manufacturer;

      vaccine.quantity = Number(req.body.quantity);

      vaccine.expiryDate = Date.parse(req.body.expiryDate);

```

```
    vaccine
      .save()
      .then(() => res.json("Vaccine updated!"))
      .catch((err) => res.status(400).json({ error: err }));
  })
  .catch((err) => res.status(400).json({ error: err }));
});

module.exports = router;
```

5.3 External APIs

Describe the APIs used in the following table.

Table 5.1: Details of APIs used in the Project

Name of API	Description of API	Purpose of Usage	List down the function/class name in which it is used
--------------------	---------------------------	-------------------------	--

5.4 User Interface

Details about user interface with descriptions will be presented in this section.

6 Testing and Evaluation

This chapter may include the following sections. (Students are required to perform the testing both manually and automatedly).

6.1 Manual Testing

6.1.1 Unit Testing

Unit Testing 1: Admin Login

Testing Objective: To ensure the admin can be login successfully

Test Case Id: CI_001

Test Case Description: Test that the admin can login with the provided credentials successfully

Test Scenario: Verify on entering predefined username and password, admin can sign in.

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Enter the predefined login details	Username: ahmed@gmail.com Password: 12345	Successfully log into the main page of the system as Admin	As Expected	Pass
2.	Enter any other credentials to login	Username: umar@gmail.com Password: Umar123	Failed to login into the system as admin	As Expected	Pass

Table 116: Test case for admin login

Unit Testing 2: Add Sub-Admin

Testing Objective: To ensure the admin can add sub admins successfully

Test Case Id: CI_002

Test Case Description: Test that the admin can add sub-admins successfully into the system

Test Scenario: Verify admin can successfully setup the sub-admin details into the system

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Admin will click on the add sub-admins and enter credentials and click on add button	Username: umar@gmail.com Password: umar12345	Sub-admin successfully added in the system	As Expected	Pass

Table 117: Test case for add sub-admin

Unit Testing 3: Add stock levels

Testing Objective: To ensure the admin can add stock levels successfully

Test Case Id: CI_003

Test Case Description: Test that the admin can add stock levels successfully into the system

Test Scenario: Verify admin can successfully add stock details into the system

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Admin will click on the add stock select vaccine, enter details and click on add	Select Polio, Manufacturer: Global pharma Expiry-28-10-2021	Stock levels added in the system	As Expected	Pass
2	Admin don't enter the expiry date		Stock update don't	As Expected	Pass

Table 118: Test case for add stock levels

Unit Testing 4: Allot vaccine to Sub admin

Testing Objective: To ensure the admin can add stock levels successfully

Test Case Id: CI_004

Test Case Description: Test that the admin can add stock levels successfully into the system

Test Scenario: Verify admin can successfully add stock details into the system

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Admin will click on the assign stock select vaccine, enter details and click on assign	Available vaccine quantity, date of assigning	Stock levels added in the system Sub admin allotted vaccine	As Expected	Pass
2	Admin will click on the assign stock select vaccine, enter details and click on assign	Admin add more value than available quantity	Stock don't update Vaccine don't update	As Expected	Pass

Table 119: Test case for allot vaccines to sub-admin

Unit Testing 5: Monitor vaccinated children in city**Testing Objective:** To ensure the regional admin can view child details successfully**Test Case Id:** CI_005**Test Case Description:** Test that the regional admin can view child details successfully**Test Scenario:** Verify regional admin can see the details

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Regional Admin will click on children of particular city		System shows all the details of the children vaccinated in the city	As Expected	Pass

Table 120: Test case for monitor vaccinated children**Unit Testing 6:** add child details**Testing Objective:** To ensure the Hospital sub-admin can add child details successfully**Test Case Id:** CI_006**Test Case Description:** Test that the Hospital sub-admin can add child details successfully

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Hospital sub-admin at the time of child birth will add child information and system gets updated	35446523, Shariq,33665932, rwp etc.	System shows add this child into the system	As Expected	Pass

Table 121: Test case for add child details

Unit Testing 7: Print child details

Testing Objective: To ensure the Hospital sub-admin can print child details successfully

Test Case Id: CI_007

Test Case Description: Test that the Hospital sub-admin can print child details successfully

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Hospital sub-admin after adding child details will click on the print details		System prints the birth certificate and vaccine schedule dates	As Expected	Pass
2	Hospital sub-admin after adding child details will click on the print details		System prints the birth certificate and vaccine schedule dates	Printer not found	Fail

Table 122: Test case for print child details

Unit Testing 8: update child details

Testing Objective: To ensure the Vaccine Center sub-admin can update child details successfully

Test Case Id: CI_008

Test Case Description: Test the update function successfully

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Vaccine Center sub-admin after vaccination will select the vaccine and click on update. Parent will get the OTP which will be verified by the system and data will be updated	Homophiles,28/09/2021	System updated the child data	As Expected	Pass

Table 123: Test case for update child details

Unit Testing 9: Request stocks**Testing Objective:** To ensure Vaccine Center sub-admin can request stocks to regional admin**Test Case Id:** CI_009**Test Case Description:** Test the request function successfully

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Vaccine Center sub-admin click on request stocks, enter details, and click on request	Homophiles,5000,28/09/2021	Request sent to Regional admin	As Expected	Pass
2	Vaccine Center sub-admin click on request stocks, enter details, and click on request	Influenza, -5000,29/08/2021	Request Not sent	As Expected	Pass

Table 124: Test case for request stocks

Unit Testing 10: Add polio workers and allot areas

Testing Objective: To ensure Vaccine Center sub-admin can add polio worker and assign the areas for vaccination

Test Case Id: CI_010

Test Case Description: Test the update function successfully

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/Suspended
1.	Vaccine Center sub-admin click on add polio workers and assign them area and stocks	Umar, PWD, 28/09/2021, Influenza, 5000	Allot successfully	As Expected	Pass

Table 125: Test case for add polio workers and allot areas

7 Conclusion and Future Work

This chapter concludes the project and highlights future work.

7.1 Conclusion

This final report for Child immunization and growth tracking is providing all documents which contains the research, problem, proposed solution, requirements, system design, implementation, testing of the modules. From this project, we are able to learn machine learning, web development with updated tech, mobile development for cross platform.

7.2 Future Work

In the future, we will try to work on enhancing its efficiency by implementing this project with data warehouses. Apart from that in future, we can also update in to general vaccinations too.

8 References

- bank, W. (2020). <https://data.worldbank.org/indicator/SH.DYN.MORT?locations=PK>. Retrieved from worldbank.
- IIS. (2020). Retrieved from <https://www.cdc.gov/vaccines/programs/iis/index.html>
- Post, F. (n.d.). Retrieved from Relief Web : <https://reliefweb.int/report/pakistan/two-more-children-die-malnutrition#:~:text=According%20to%20the%20health%20department,facilities%20in%20the%20remote%20region.>
- unicef. (2021, January). *Unicef*. Retrieved Januray 15, 2021, from <https://www.unicef.org/pakistan/health>
- WHO. (2020). Retrieved from World health organization: <http://www.emro.who.int/pak/programmes/expanded-programme-on-immunization.html#:~:text=The%20Expanded%20Programme%20on%20Immunization,%2C%20pertussis%2C%20tetanus%20and%20measles.>