

INTRODUCTION

1.1 About the Company

SIRMOR SOFTWARE SOLUTION PVT. LTD



Founded in 2014, **Sirmor Software Solution** is driven by the passion of understanding your business and thus providing value for your investments. Through a bunch of **Digital Marketing and Website Design & Development Services**, our name stands synonymous with **Digital Marketing** today. They help the clients achieve targeted traffic & high visibility and offer digital marketing techniques to reach a massive audience.

Providing optimal user-experience, error-free coding, picture-perfect website design is a routine for us. Either you select a theme for the website yourself or let us do the honours, a customized website development that fits your needs awaits you. Whether you require a correction or an advancement in your existing website or want to develop a completely new one, we create all keeping the aesthetics as well as high-ranking SEO in mind.

Experience, both in B2B as well as B2C website development and design, always goes on to create appealing avenues for our clients. With us in the picture get ready for exceptional outcomes because we-

- Work on the SEO's so that clients don't miss out on your profile on the already congested search engine page.
- Have a perfect CMS- Content Management System in place which due to its authenticity always makes sure that your brand communication is unique, plagiarism free and crisp.

Regardless of the size and depth of your website, we make sure that our best team of designers works on your project and presents you an identity that is incomparable in terms of design. Right from incorporating everything about the company in the initial planning stages to building a completely working website that is home to stunning designs and fully-mobile responsiveness, we intend to bring to your website the most prized element- TRAFFIC.

1.2 About the Project

PROJECT DESCRIPTION

Young children are at increased risk for infectious diseases because their immune systems have not yet built up the necessary defences to fight serious infections and diseases. As a result, diseases like whooping cough or pneumococcal disease can be very serious and even deadly for infants and young children. Vaccinations start early in life to protect children before they are exposed to these diseases. The proposed system of MIS for Children Vaccination provide proper schedule of children vaccine time interval for the parents. Parents can search nearby hospital and make a schedule. Hospital User will manage the child and vaccination report and approval of appointment. Hospital will update the status of vaccination applied for child.

MIS for Children Vaccination is a web application which is implemented in Ionic Framework platform using Codeigniter4 and PHP. The MIS for Children Vaccination is User and Admin friendly. Tool for Admin to Register Hospital, Create and Maintain User's Tool. And Tools for Users to Register Child, Add Vaccines, Make Vaccination Schedule for Child and send Emails to the parents.

In this user is a hospital user. First of all the hospital will be registered and through that the Administrator will give them UserId and Password. In user login section a user can login immediately but If her user login for the first time then they have to change the password only then they can be able to use the app further

MIS for Children Vaccination that I am proposing here is the web-based interpretation of the service provided to the Hospital. The purpose of this project is to manage the Hospital Details for the Admin. And manage Child personal details, Staff and Doctor Personal details, Vaccine Details for the user. So user can create vaccination schedule for the Registered Child and can manipulate the schedule as required.

The user and admin module are run separately in the present system. The MIS for Children Vaccination assists us with large amount of data in organized form. Here the purpose is to enhance the existing system work and to meet the updated technology which is upgrading day by day.

SYSTEM ANALYSIS

Introduction:-

“System analysis is the process of gathering and interpreting facts, diagnosis problems and using the information to recommends improvement to the system.”

System analysis is the first stage according to the System Development Life Cycle model. This system analysis is a process that starts with the analyst. Analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. One aspect of analysis is defining the boundaries of the system and determining where or not a candidate system should consider other related systems. During analysis , data are collected on the available files , decision points, and transaction handled by present system. Logical system models and tools that are used in analysis . Training , experience and common sense are required for collection of the information needed to do the analysis.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem area are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is the loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is the problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

2.1 EXISTING SYSTEM.

Earlier, when this technology was not there, a Parents had to go to the hospital many times to get their child vaccination. In this situation, the parents had to travel, no matter how far the hospital was.

Also, it would be difficult for the to handle multiple child data and creating and maintaining a vaccination schedule for child used to be difficult task at the same time. When there are many child, the paper work for the hospital staff increased greatly, due to which he used to

collect a lot of data in the form of paper. It is also a challenge for the hospital staff to keep all the paper works related to his account. And at the same time, handling and managing all these paper sections was also very difficult and sometimes such situations used to come in which all the paper work used to come on the verge of bursting.

2.2 DRAWBACKS OF EXISTING SYSTEM

In the existing system there are many problems which are faced during their work.

- Manual process
- Difficult to get data.
- Maintenance of the records is difficult.
- Involves large amount of paper work.
- Time consuming process.
- Slow Updating and Renewal of Data.
- There was a time when we want any scheme related information was very difficult and time consuming, but now we can find the scheme according to us with the help of “**MIS FOR CHILDREN VACCINATION**” web application.

2.3 PROPOSED SYSTEM

MIS for Children Vaccination that I am proposing here is the web-based interpretation of the service provided to the users. The purpose of this project is to manage the hospital details, user personal details, child personal details, vaccine details, including maintenance of vaccination schedule, and sends the emails to the parents of next vaccination to the child according to his/her schedule so that child will not miss any vaccination. Even if for some reason child's parent are not able to get vaccinate them from the same hospital then at least through emails they will get reminder that their child has to get this vaccine so they will able to get the vaccine from other hospital.

In this user is a hospital user. First of all the hospital will be registered and through that the Administrator will give them UserId and Password. In user login section a user can login immediately but If her user login for the first time then they have to change the password only then they can be able to use the app further

The user and admin module are run separately in the present system. The MIS for Children Vaccination assists us with large amount of data in organized form. Here the purpose is to enhance the existing system work and to meet the updated technology which is upgrading day by day.

- Web Application designing with Ionic Framework components and updated information.
- Dynamically updation of schemes and information.
- Systematic flow of process.

2.4 ADVANTAGE OF PROPOSED SYSTEM

This system provides a Common User Interface for the system to login

- Very simple and easy to use.
- Less time consuming.
- All registered User detail's and their order details.
- Admin discretion and control over the entire system.
- Reduces a lots of paperwork.
- Easy to manage large amount of data.
- Easy to get complete reports of each orders and delivery.

SYSTEM PLANNING

The overall goal of “**MIS FOR CHILDREN VACCINATION**” planning is to establish a pragmatic strategy for controlling, monitoring a complex technical project. So the end result gets done on time, with quality hence planning and control even more important in software development an engineering approach:

- We have to attempt to estimate cost/effort of our project.
- Plan and schedule work, how the time will maintain.
- Involve user in defining requirements of project.
- Identify stages in development.
- Schedule reviews both for control and quality.
- Plan extensive testing.

3.1 PERT CHART

PERT Chart is acronym for (**Program Evaluation and Review Technique**). A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project. It is basically a method to analyse the tasks involved in completing a given project, especially the time needed to complete each task, and to identify the minimum time needed to complete the total project.

A PERT chart is a project management tool that provides a graphical representation of a project's timeline. PERT, or Program Evaluation Review Technique, breaks down the individual tasks of a project for analysis.

3.1.1 INTERPRETING PERT CHART

- A PERT chart is a visual representation of a series of events that must occur within a project's lifetime.
 - The direction of arrows indicates the flow and sequence of events required for project completion.
 - Dotted activity lines represent dummy activities, which are items located on another PERT path.
1. Numbers and time allotments are assigned and shown inside each vector.

3.1.2 BENEFITS OF PERT CHARTS

2. PERT charts are useful for what-if analyses.

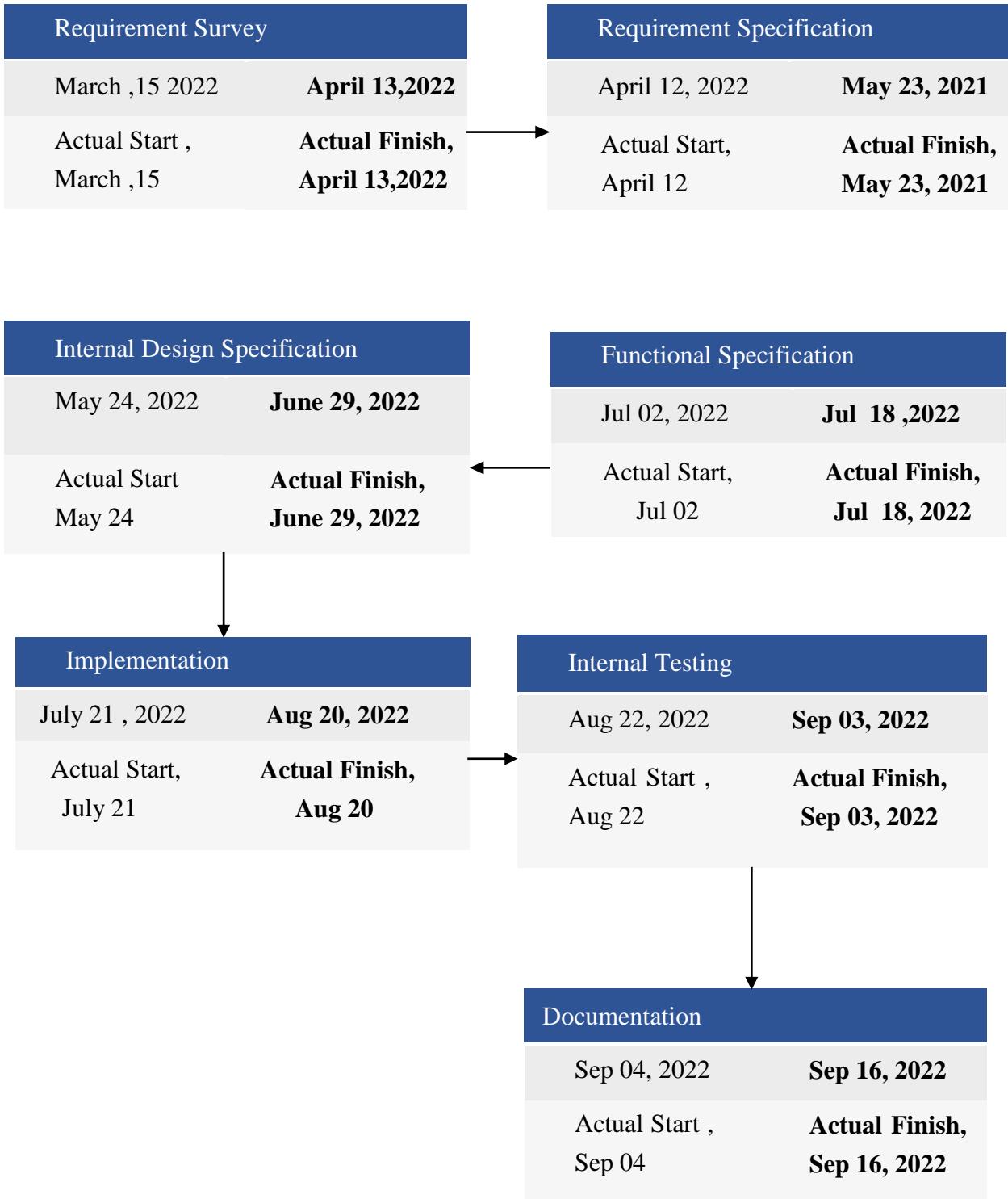
3. Understanding the possibilities concerning the flow of project resources and milestones allows management to achieve the most efficient and useful project path.

3.1.3 WHEN TO DRAW PERT CHART

Project managers can use PERT charts to solve the following problems:

1. Plan a realistic timetable for project completion.
2. Identify critical path - since the path leads to the minimum time the project requires, any delays to these tasks will impact the completion of the overall project.
3. Identify tasks that can be carried out concurrently.
4. Identify tasks that need to be compressed if the overall project time needs to be reduced.
5. Identify slack time where certain tasks are not as time-critical to the overall deadline.

PERT CHART



3.2 COST BENEFIT ANALYSIS

For “**MIS for Children Vaccination**”, Cost/benefit analysis gives a picture of the various costs, benefits, and rules associated with each alternative system. In developing cost estimates for a system, we need to consider several cost elements. Among them are hardware, personal, facilities, operating.

Hardware cost relate to the actual purchase or lease of the computer and peripherals. Costs incurred during the development of a system are one-time costs and are labeled developmental costs. Once the system is installed the cost of operating and maintaining the system become recurring cost.

Facility costs are expenses incurred in the preparation of the physical site where the application or the computer will be in operation.

Operating costs include costs associated with the day-to-day operation of the system. One approach is to treat operating costs as overhead. Another approach is to charge each authorized user for the amount of processing they request from the system.

A system is also expected to provide benefits. The first-task is to identify each benefit and turn assigns a monetary value to it for cost/benefit analysis. Benefits may be tangible and intangible, direct or indirect.

Terms:

1. Tangible or Intangible Costs – Tangibility refers to the ease with which costs or benefits can be measured. An outlay of cash for a specific item or activity is referred to as a tangible cost. The purchase of hardware or software personnel training and employee salaries are examples of tangible costs. They are readily identified and measured.

Costs that are known to exist but whose financial value cannot be accurately measured are referred to as intangible costs. For example, employee movable problems caused by a new system or lowered company image is an intangible costs. In some cases, intangible costs may be easy to identify difficult to measure.

Benefits – benefits are also classified as tangible or intangible. Like cost, they are often difficult to specify accurately. Tangible, benefits such as completing jobs in fewer hours or producing reports with no errors are quantifiable. Intangible benefits such as more satisfied customers or an improved corporate image, are not easily quantified. Both tangible and intangible costs and benefits, however, should be considered in the evaluation process.

2. Direct or indirect costs- direct costs are those with which a dollar figure can be directly associated in a project. They are applied directly to the operation. For example, the purchase of a box of diskettes for \$35 is a direct cost because the diskettes can be associated with the dollars expended.

Indirect costs are the results of operations that are not directly associated with a given system or activity. They are often referred to as overhead. A system that reduces overhead realizes a savings.

Benefits – direct benefits also can be specifically attributable to a given project. For example, a new system than can handle 25 percent no transaction per day is a direct benefit.

Indirect benefits are realized as a by product of another activity or system. For example, proposed safe deposit billing system that provides profits showing vacant boxes by sizes, location, and price, will help management decide on how much advertising to do for box rental.

Direct or indirect costs and benefits are readily identified for tangible costs and benefits respectively.

3. Fixed or Variable Costs - fixed costs are sunk costs. They are straight-line depreciation of hardware, exempt employee salaries and insurance. In contrast, variable costs are incurred on a regular basis. They are usually proportional to work volume and continue as long as the system is in operation. For example the costs of computer forms vary in proportion to the amount of processing or the length of the reports required.

Benefits - fixed benefits are also constant and do not change. An example is a decrease in the number of personnel by 20 percent resulting from the use of a new computer.

In this project cost is incurred in terms of time consumed, electricity used etc.

Costs and benefits are as follows:

COST

- **Hardware Cost:** Single computer system is used in development of application. All hardware parts are working good and its quality is perfect.
- **Personnel Cost:** Staff is not required. Administrator will be responsible for maintaining the system and its records. So the developer is responsible for extra cost.
- **Facility Cost:** Electricity is being consumed in developing this application.

BENEFIT

- This application leads to less time consuming.
- Our database can store large amount of data can be stored serially and accessed frequently.
- Staff reduction-only single person is required which will act as an administrator to maintain the system.

FEASIBILITY STUDY

Feasibility study is an important phase in the software development process. It enables the developer to have an assessment of the product being developed. It refers to the feasibility study of the product in terms of outcomes of the product, operational use and technical support required for implementing it.

Feasibility study should be performed on the basis of various criteria and parameters. The various feasibility studies are:

The data collection that occurs during preliminary investigation examines system feasibility, the likelihood that the system will be beneficial to the organization.

4.1 WHAT WILL A FEASIBILITY STUDY TELLS?

A feasibility study will cover all potential threats to your project. It will focus on any specific project concerns, but in general, the feasibility study will:

1. Determine if an environmental impact assessment (EIA) is required.
2. Assist in the development of project documentation: business case, execution plan, and strategic brief.
3. Determine the necessary planning permissions needed.
4. Determine other legal/statutory approvals needed.
5. Analyze the budget relative to the client requirements.
6. Assess the potential to re-use any existing facilities.
7. Assess any and all site information provided by the client.
8. Include site appraisals, including geotechnical studies, assessment of any site contamination, availability of services, uses of adjoining land, easements and restrictive covenants, environmental impacts, etc.
9. Assess operational and maintenance issues.
10. Appraise servicing strategies.
11. Address programming considerations.
12. Address procurement options.
13. Overall, establish whether the project is viable.
14. Help identify feasible options.

4.2 FOCUSED AND SPECIFIC

Feasibility studies are focused and specific. They start with a single question -- asking whether the idea, event or action is a viable solution. A feasibility study is an analysis of how successfully a project can be completed.

Three tests are studies:

1. Technical Feasibility.
2. Economical Feasibility.
3. Operational Feasibility.

4.2.1 TECHNICAL FEASIBILITY

It involves determining whether or not a system can actually be constructed to solve the problem at hand. Some users expect too much of computers, assuming that computers can accurately predict the future, immediately reflect all information in an organization, easily understand speech, or figure out how to handle difficult problems. Such systems, even if they exist, are not yet available for widespread use.

The technical issues raised during the feasibility stage of the investigation are:

1. Does the necessary technology exist (can it be acquired) to do what is suggested?
2. Does the proposed equipment have the technical capacity to hold the data required to use the new system?
3. Will the proposed system and components provide adequate responses to queries, regardless of the number or location of users?
4. Can the system be expanded, if developed?
5. Are there technical guarantees of accuracy, reliability, ease of access and data security?

Following are the results on the basis of technical feasibility study:

Issues	Solution	Reason
Necessary technology?	Ionic Framework ,Angular 12,Nodejs and MSSQL	The Ionic Framework comes with number of high level UI components for frontend design. Angular documentation is the best on the web application Nodejs Best for RestApi and MSSQL for database.
Technical capacity?	Computer System, Mobile, Tablet.	Tablet & Mobile has the capacity to run on OS and computer system has the capacity to access Angular 9 & node.js.
Expand number of user?	Yes	System can be used by multiple users who have the authority of access.
Accuracy of system?	Guaranteed	This system always provide accurate Result according the query.
Reliability and security?	Guaranteed	Windows is secured operating system.
Ease of access?	Guaranteed	Angular is User Friendly Technology
Can System be expanded?	Yes	System can be expanded by adding new features as required.

4.2 ECONOMICAL FEASIBILITY

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as cost / benefit analysis; in this procedure we determine the benefits and savings that are expected from a proposed system and compare them with costs. We found the benefits outweigh the costs; we take a decision to design and implement the new proposed system.

During the feasibility phase, broad alternatives solutions are examined. For each alternate solution the cost and benefits have to be examined before designing one of the alternatives.

Broad solutions will consist of:

1. Specifications of information to be made available by the system.
2. Description of what will be done manually and what the computer will do.
3. Specification of new computing equipment needed or specification of expansion of an existing computer.

In performing cost and benefit analysis it is important to identify cost and benefits factors.

Cost and benefits can be categorized into the following categories:

1. **Development cost:** A Development cost is the costs that are incurred during the development of the system. It is one time investment.
2. **Operating cost:** Operating cost are the expenses required for the day to-day running of the system. As, operating cost are wages, supplies and overheads.
3. **Hardware/Software cost:** It includes the cost of purchasing or leasing of computers and its peripherals. Software costs involve required software cost.
4. **Supply cost:** These are variable costs that are very proportionately with the amount of use of paper, ribbons, disks, and others.

It's take the costs required to assemble and run my project

ITEM	COST(Rs.)
Computer	20,000(depend upon the configuration)
Tablet	12,000(depend upon the configuration)
Mobile	7000(depend upon the configuration)
Project Cost	5,000
Total	44,000

BENEFITS

1. Fast and easy access to all procedures and functions.
2. No need for large storage spaces sized of rooms for storing the cabinets because all the information about the members and other details is saved in the computer's hard disks.
3. High level of security and authentication of each and every user.
4. Reliability is increased, as backups of files, and records can be made and saved.
5. Different locations and information will be highly secure, unlike in file cabinets where entries can easily be ripped or tampered with by users.
6. There will be no longer need for all the paper work required to make timely reports lists or other lists as the program generates them at any time at a very quick pace.

4.3 OPERATIONAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

It is mainly related to human organizational and political aspects. The points to be considered are:

1. What changes will be brought with the system?
2. What new skills will be required?
3. Do the existing staff members have these skills?

Following are the results on the basis of technical feasibility study:

Issues	Solution
What changes will be brought with the System?	New system is computerized.
What new skills will be required?	Basic Computer and android mobile Knowledge.
Do the existing staff members have these Skills?	yes

SYSTEM REQUIERMENT SPECIFICATION

The requirement phase basically consists of three activities:

- Requirement Analysis
- Requirement Specification
- Requirement Validation

Requirement Analysis is a software engineering task that bridges the gap between system level software allocation and software design. It provides the system engineer to specify software function and performance indicate software's interface with the other system elements and establish constraints that software must meet.

The basic aim of this stage is to obtain a clear picture of the needs and requirements of the end-user and also the organization. Usually analysts research a problem by asking questions and reading existing documents. The analysts have to uncover the real needs of the user even if they don't know them clearly. During analysis it is essential that a complete and consistent set of specifications emerge for the system. Here it is essential to resolve the contradictions that could emerge from information got from various parties. This is essential to ensure that the final specifications are consistent.

Requirement Analysis in this Project

The main aim in this stage is to assess what kind of a system would be suitable for a problem and how to build it. The requirements of this system can be defined by going through the existing system and its problems. They discussing (speak) about the new system to be built and their expectations from it. The steps involved would be

Evaluation and Synthesis:

In the proposed system this application saves the lot of time, and it is time saving process when we use this application. Using this application we can easy to manage farmer details, such as registration forms and reporting for the entire report and easy to maintain the all data. No specific training is required for the users to use this application. They can easily use the tool that decreases manual hours spending for normal things and hence increases the performance.

5.1) MODULE DESCRIPTION

MODULES OF THE PROJECT ARE:

1. ADMIN MODULE : These type of Admin can manages the functionalities of Register and update Hospitals and Details and generate UserId and Password for the hospital user so user can logs in the system, creates state, add district, add city, and can do modification as required, fulfils the user's requirements.

ROLE OF ADMIN MODULE:

1. Login

- Client Side and Server Side Validation
- Admin and User Login
- Password Encryption
- If User is login for the first time then he will have to change the password

2. Change Password(For User Only)

- Client Side and Server Side Validation
- Password Encryption

Admin Module:-

3. State Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Deletion, deleting state will also delete the related district.
- Check duplicate state name

4. District Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Deletion, deleting district will also delete the related city.
- Check duplicate state name

5. City Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Delete City Data
- Check duplicate city name

6. Hospital Master

- Client Side and Server Side Validation

- Inserting Data(Hospital Registration)
- Updating Data
- Show Data
- Delete
- Hospital Registration Number and Mobile Number will also store in login form

2. USER MODULE:- These type of user can search a child and also create and maintain vaccination schedule.

In user login module section a user can login immediately, Then user add department, add designation, add department, add staff, add doctor, add blood group, add vaccine as requirement. Once system ready then it's ready to register child and create and maintain vaccination Schedule. Then the reminder message will be sent to the registered Email as per the schedule. User can any time modify the added data which won't affect the saved large amount of data.

ROLE OF USER MODULE:

1. Department Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Delete Department
- Check duplicate department name

2. Designation Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Deleting designation
- Check duplicate designation name

3. Child Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Delete Child Data

4. Staff Master

- Client Side and Server Side Validation
- Inserting Data

- Updating Data
- Show Data
- Delete data

5. Doctor Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Delete Data

6. Child Registration Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Delete Data

7. Vaccine Master

- Client Side and Server Side Validation
- Inserting Data
- Updating Data
- Show Data
- Delete Data
- Check duplicate vaccine name

8. Vaccination Schedule

- Client Side Validation
- Searching Child Data
- Make a Vaccination Schedule for Child
- Delete inserted data

9. Vaccination

- Client Side Validation
- Searching Child Data
- Reschedule the Vaccination Schedule
- Entry of child vaccinated
- Entry of child vaccine from somewhere else
- Remove child data if child has died
- Show the report of vaccinated child

10. Reschedule

- Client Side Validation
- Updating Data

11. Child Vaccinated

- Client Side Validation
- Inserting consulted Doctor and Staff

12. Child Vaccinated Report

- Show the report
- Print the report

5.2 MINIMUM HARDWARE & SOFTWARE REQUIREMENTS

5.2.1 HARDWARE REQUIREMENTS:

- Processor : Minimum 1.2 GHz, Dual Core or above
- Operating System : Windows 7(64 bit) or above
- RAM : Minimum 4 GB or above
- Internal Storage : 512 GB (Recommended) or above
- Mouse : Optical (Recommended)
- Keyboard : Multimedia (Recommended)
- Monitor : 14" or above

5.2.2 SOFTWARE REQUIREMENTS:

- Web Presentation : Ionic Framework Components ,HTML, CSS
- Client – side Scripting : Java Script/Jquery
- Development Framework : MS Visual Studio Code and Codeigniter4
- Programming Language : PHP
- Database Connectivity : MySql
- Backend Database : Xampp Server PHP Version 7.4.27
- Operating System : Windows 7(64 bit) or above
- Web Server : Xampp Server
- Supported Browsers : Google Chrome / Firefox / Opera / Internet Explorer

5.3 TECHNOLOGY DESCRIPTION

MIS for Children Vaccination is a web-based application. To develop this application following tools are required:

DESIGN FRAMEWORK:IONIC FRAMEWORK-

5.3.1 Ionic Framework's app development platform builds amazing cross-platform mobile, web, and desktop apps all with one shared code base and open-web standards. Ionic is a complete open-source SDK for hybrid mobile app development created by Max Lynch, Ben Sperry, and Adam Bradley of Drifty Co. in 2013. The original version was released in 2013 and built on top of AngularJS and Apache Cordova, Ionic framework needs native wrapper to be able to run on mobile devices.

IONIC FRAMEWORK UI Components

Ionic apps are made of high-level building blocks called Components, which allow you to quickly construct the UI for your app. Ionic comes stock with a number of components, including cards, lists, and tabs. Once you're familiar with the basics, refer to the API Index for a complete list of each component and sub-component.

5.3.2 HTML5:

HTML is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and current version of the HTML standard. HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a candidate for cross-platform mobile applications, because it includes features designed with low-powered devices in mind

5.3.3 CSS(Cascading Style Sheets)-

A **CSS framework** is a library allowing for easier, more standards-compliant web design using the Cascading Style Sheets language. Most of these frameworks contain at least a grid. More functional frameworks also come with more features and additional JavaScript based functions, but are mostly design oriented and focused around interactive UI patterns. This detail differentiates CSS frameworks from other JavaScript frameworks.

5.3.4 ANGULAR 12

CodeIgniter is an Application Development Framework - a toolkit - for people who build web sites using PHP. Its goal is to enable you to develop projects much faster than you could if you were writing code from scratch, by providing a rich set of libraries for commonly needed tasks, as well as a simple interface and logical structure to access these libraries. CodeIgniter

lets you creatively focus on your project by minimizing the amount of code needed for a given task.

CodeIgniter is based on the Model-View-Controller (MVC) development pattern. MVC is a software approach that separates application logic from presentation. In practice, it permits your web pages to contain minimal scripting since the presentation is separate from the PHP scripting.

Where possible, CodeIgniter has been kept as flexible as possible, allowing you to work in the way you want, not being forced into working any certain way. The framework can have core parts easily extended or completely replaced to make the system work the way you need it to. In short, CodeIgniter is the malleable framework that tries to provide the tools you need while staying out of the way. CodeIgniter 4 is the latest version of the framework, intended for use with PHP 7.3+ (including 8.1). The initial release was February 24, 2020.

5.3.4 DEVELOPMENT FRAMEWORK: MS VISUAL STUDIO CODE -

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity)

Visual Studio Code was announced on April 29, 2015, by Microsoft at the 2015 Build conference. A Preview build was released shortly thereafter.

On November 18, 2015, Visual Studio Code was released under the MIT License and its source code posted to GitHub. Extension support was also announced.

On April 14, 2016, Visual Studio Code graduated the public preview stage and was released to web.

FEATURES:

Visual Studio Code is a source code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js and C++. It is based on the Electron framework, which is used to develop Node.js web apps that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a language-agnostic code editor for any language, contrary to Microsoft Visual Studio which uses the proprietary `.sln` solution file and project-specific project files. It supports a number of programming languages and a set of features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many of Visual Studio Code features are not exposed through menus or the user interface, but can be accessed via the command palette.

Visual Studio Code can be extended via extensions, available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new languages, themes, and debuggers, perform static code analysis, and add code linters using the Language Server Protocol.

Visual Studio Code includes multiple extensions for FTP, allowing the software to be used as a free alternative for web development. Code can be synced between the editor and the server, without downloading any extra software.

Visual Studio Code allows users to set the code page in which the active document is saved, the newline character, and the programming language of the active document. This allows it to be used on any platform, in any locale, and for any given programming language.

Language support-

Out-of-the-box, Visual Studio Code includes basic support for most common programming languages. This basic support includes syntax highlighting, bracket matching, code folding, and configurable snippets. Visual Studio Code also ships with IntelliSense for JavaScript, TypeScript, JSON, CSS, and HTML, as well as debugging support for Node.js. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace.

Data collection-

Visual Studio Code collects usage data and sends it to Microsoft, although this can be disabled. In addition, because of the open-source nature of the app, the telemetry code is

accessible to the public, who can see exactly what is collected. According to Microsoft, the data is shared with Microsoft-controlled affiliates and subsidiaries, although the law enforcement may request it as part of a legal process.

Open-source versions

Visual Studio Code is a distribution of the "Code - OSS" repository with Microsoft-specific customizations released under a traditional Microsoft product license.

VSCodium is an alternative binary distribution of the app that uses only the open-source parts and omits Microsoft's trademarks and the telemetry component, while remaining fully functional and compatible in all other regards.

LANGUAGE USED: JAVA SCRIPT

5.3.5 JAVASCRIPT:

JavaScript is a high level, dynamic, untyped and integrated programming language. Alongside HTML and CSS, it is one of the three core technologies of World Wide Web content production; the majority of websites employ it and it is supported by all modern Web browsers without plug-ins. Java Script is a prototype-based with first class functions, making it a multi-paradigm language, supporting object-oriented imperative and functional programming styles. It has an API for working with text, arrays, dates and regular expressions, but does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

Features of JavaScript:

- Imperative and structured
- Dynamic
- Prototype-based (Object-oriented)
- Functional

jQuery:

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. jQuery is the most popular JavaScript library in use today, with installation on 65% of the top 10 million highest-trafficked sites on the Web. jQuery is free, open-source software licensed under the MIT License.

5.3.6 DATABASE: MYSQL

MYSQL. My Structured Query Language

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses.

MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

XAMPP Server :

XAMPP is one of the widely used cross-platform web servers, which helps developers to create and test their programs on a local webserver. It was developed by the Apache Friends, and its native source code can be revised or modified by the audience. It consists of Apache HTTP Server, MariaDB, and interpreter for the different programming languages like PHP and Perl. It is available in 11 languages and supported by different platforms such as the IA-32 package of Windows & x64 package of macOS and Linux.

What is XAMPP?

XAMPP is an abbreviation where X stands for Cross-Platform, A stands for Apache, M stands for MYSQL. and the Ps stand for PHP and Perl, respectively. It is an open-source

package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB,

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl

is a programming language used for web development, PHP

is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL.

SERVER SIDE LANGUAGE:

5.3.7 PHP :-

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Ramses Leadoff unleashed the first version of PHP way back in 1994.

- PHP is a recursive acronym for "PHP: Hypertext Pre-processor".
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
- PHP is forgiving: PHP language tries to be as forgiving as possible
- PHP Syntax is C-Like.

5.3.8 Supported Browsers-

Google Chrome –

Google Chrome is a cross-platform web browser developed by Google. It was first released in 2008 for Microsoft Windows, built with free software components from Apple WebKit and Mozilla Firefox. It was later ported to Linux, macOS, iOS, and Android, where it is the default browser.

SYSTEM DESIGN

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”. It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. System design goes through two phases of development: Logical and Physical Design.

6.1 ITERATIVE MODEL

In this Model, you can start with some of the software specifications and develop the first version of the software. After the first version if there is a need to change the software, then a new version of the software is created with a new iteration. Every release of the Iterative Model finishes in an exact and fixed period that is called iteration.

The Iterative Model allows the accessing earlier phases, in which the variations made respectively. The final output of the project renewed at the end of the Software Development Life Cycle (SDLC) process.

An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of each iteration of the model.

The various phases of Iterative model are as follows:

- 1. Requirement gathering & analysis:** In this phase, requirements are gathered from customers and check by an analyst whether requirements will fulfil or not. Analyst checks that need will achieve within budget or not. After all of this, the software team skips to the next phase.
- 2. Design:** In the design phase, team design the software by the different diagrams like Data Flow diagram, activity diagram, class diagram, state transition diagram, etc.
- 3. Implementation:** In the implementation, requirements are written in the coding language and transformed into computer programmes which are called Software.

4. Testing: After completing the coding phase, software testing starts using different test methods. There are many test methods, but the most common are white box, black box, and grey box test methods.

5. Deployment: After completing all the phases, software is deployed to its work environment.

6. Review: In this phase, after the product deployment, review phase is performed to check the behaviour and validity of the developed product. And if there are any error found then the process starts again from the requirement gathering.

7. Maintenance: In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required. Maintenance involves debugging and new addition options.

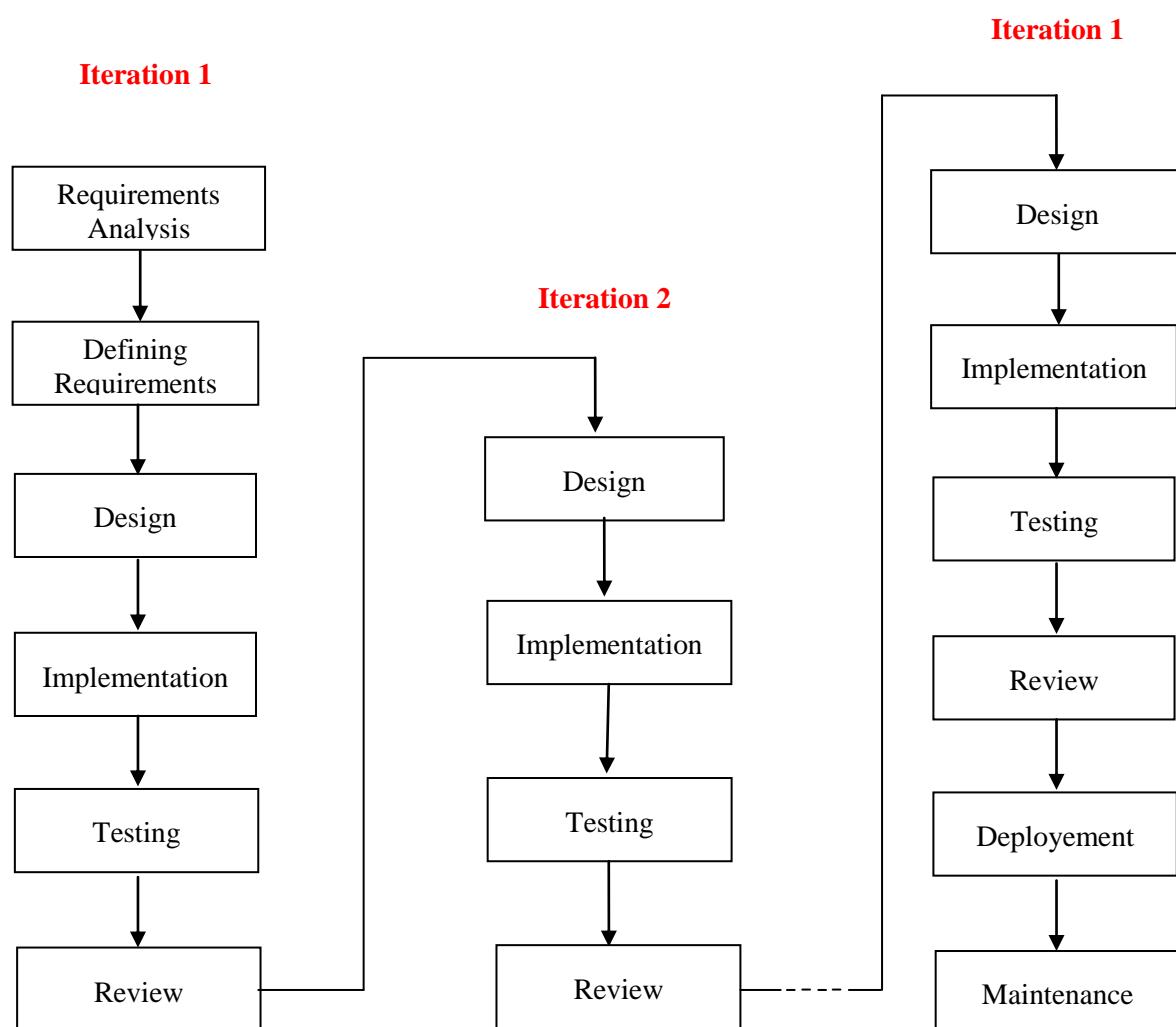


Figure 6.1.1 Iterative Model Diagram

Advantages:-

1. Testing and debugging during smaller iteration is easy.
2. A Parallel development can plan.
3. It is easily acceptable to ever-changing needs of the project.
4. Risks are identified and resolved during iteration.
5. Limited time spent on documentation and extra time on designing.

6.2 Database Design:-

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- ❖ Data Integrity
- ❖ Data independence

NORMALIZATION

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy i.e. repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insertion, updating, deletion anomalies.

Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relation.

Insertion anomaly: Inability to add data to the database due to absence of other data.

Deletion anomaly: Unintended loss of data due to deletion of other data.

Update anomaly: Data inconsistency resulting from data redundancy and partial update

Normal Forms: These are the rules for structuring relations that eliminate anomalies.

FIRST NORMAL FORM:

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this we mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

SECOND NORMAL FORM:

A relation is said to be in second Normal form is it is in first normal form and it should satisfy any one of the following rules.

- 1) Primary key is a not a composite primary key
- 2) No non key attributes are present
- 3) Every non key attribute is fully functionally dependent on full set of primary key.

THIRD NORMAL FORM:

A relation is said to be in third normal form if their exists no transitive dependencies. Transitive Dependency: If two non-key attributes depend on each other as well as on the primary key then they are said to be transitively dependent.

The above normalization principles were applied to decompose the data in multiple tables thereby making the data to be maintained in a consistent state.

DATA DICTIONARY:

A data dictionary contains:

- The definitions of all schema objects in the database (tables, views, indexes, clusters, synonyms, sequences, procedures, functions, packages, triggers, and so on)
- How much space has been allocated for, and is currently used by, the schema objects
- Default values for columns

- Integrity constraint information
- Privileges and roles each user has been granted
- Auditing information, such as who has accessed or updated various schema objects

\

User Login

Fields	Datatype	Description
Id	Int(11)	Auto increment
userId	Varchar(10)	Primary key
userType	Int(11)	Default (1)
userPassword	Longtext	
changePassword	Longtext	Default (0)
sessionId	Varchar(100)	Default (0)
IsActive	Int(11)	Default (0)
isAlreadyLogin	Varchar(20)	Default (0)

User Login Logout Date Time

Fields	Datatype	Description
Id	Int(11)	Primary Key, AI
userType	Int(11)	
userId	Varchar(100)	
loginDateTime	datetime	
logoutDateTime	datetime	
ipAddress	Varchar(100)	

Master - Hospital Registration

Fields	Datatype	Description
hospitalId	Int(11)	Auto increment
HospitalRegistrationNumber	Varchar(10)	Primary key
hospitalName	Text	
StateId	Int(11)	Foreignkey
DistrictId	Int(11)	Foreignkey
CityId	Int(11)	Foreignkey

Address	Text()	
contactPerson	Varchar(100)	
contactNumber	Varchar(10)	
Pincode	Varchar(6)	
hospitalEmail	Varchar(100)	
Flag	Int(11)	Define as 1
insertedBy	Varchar(50)	
insertedDateTime	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(100)	
updatedDateTime	Datetime	
updatedIp	Varchar(20)	`

Master - Child Registration

Fields	Datatype	Description
childId	Int(11)	Auto Increment
registrationNumber	Varchar(10)	PrimaryKey
fullName	Varchar(100)	
Father_Name	Varchar(100)	
Mother_Name	Varchar(100)	
Gender	Varchar(6)	
Blood_Group	Varchar(3)	
DOB	Datetime	
Mobile_Number	Varchar(10)	

Email	Varchar(100)	
StateId	Int(11)	Foreignkey
DistrictId	Int(11)	Foreignkey
CityId	Int(11)	Foreignkey
Pin	Varchar(6)	
Address	Text()	
Admission_date	Datetime	
Reffering_Doctor	Varchar(100)	
Consultant_Doctor	Varchar(100)	
Birth Hospital Datilas	Varchar(200)	
Flag	Int(11)	Define as 1
insertedBy	Varchar(10)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(10)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master -Vaccine

Fields	Datatype	Description
serialNumber	Int(11)	Auto Increment
vacchneId	Varchar(6)	Primarykey
vaccineName	Varchar(200)	
Prevent	Varchar(200)	

physicalForm	Varchar(20)	
Composition	Varchar(200)	
Brand	Varchar(200)	
Packaging Type	Varchar(20)	
PackingSize	Varchar(20)	
packagingdDose	Varchar(20)	
Type of Medicines	Varchar(60)	Allopathic
Flag	Int(11)	Define as 1
insertedBy	Varchar(10)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(10)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master – Blood Group

Fields	Datatype	Description
bloodGroupId	Int(11)	Primarykey, Auto Increment
bloodGroupName	Varchar(60)	
Flag	Int(11)	Define as 1
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	

updatedDate	Datetime	
updatedIp	Varchar(20)	

Master -Staff

Fields	Datatype	Description
Id	Int	AI
StaffId	Varchar(6)	PrimaryKey
staffName	Varchar(100)	
fatherName	Varchar(100)	
Gender	Varchar(6)	
Designation	Varchar(60)	
DOB	Datetime	
mobileNumber	Varchar(10)	
Email	Varchar(200)	
StateId	Int(11)	Foreignkey
DistrictId	Int(11)	Foreignkey
CityId	Int(11)	Foreignkey
Pin	Varchar(6)	
localAddress	Text()	
parmanentAddress	Text()	
joiningDate	Datetime	
Image	Varchar(200)	
departmentId	Varchar(100)	

IsActive	Int(11)	Define as 1
insertedBy	Varchar(10)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(10)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master - Doctor

Fields	Datatype	Description
Id	Int(11)	AI
Doctored	Varchar(6)	PrimaryKey
doctorName	Varchar(200)	
Gender	Varchar(6)	
Designation	Varchar(60)	
DOB	Date	
mobileNumber	Varchar(10)	
Email	Varchar(200)	
StateId	Int(11)	Foreignkey
DistrictId	Int(11)	Foreignkey
CityId	Int(11)	Foreignkey
Pin	Varchar(6)	
Local_Address	Text()	
Parmanent_Address	Text()	

Joining_Date	Date	
Image	Varchar(100)	
DepartmentId	Varchar(60)	
IsActive	Int(11)	Define as 1
insertedBy	Varchar(10)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(10)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master - Department

Fields	Datatype	Description
departmentId	Int(11)	Primarykey, Auto Increment
departmentName	Varchar(60)	
departmentLocation	Varchar(20)	
Flag	Int(11)	Define as 1
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master - State

Fields	Datatype	Description
StateId	Int(11)	Primarykey, Auto Increment
stateName	Varchar(60)	
Flag	Int(11)	Define as 1
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master - District

Fields	Datatype	Description
destrictId	Int(11)	Primarykey, Auto Increment
destrictName	Varchar(60)	
StateId	Int(11)	Foreignkey
Flag	Int(11)	Define as 1
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	
updatedDate	Datetime	

updatedIp	Varchar(20)	
-----------	-------------	--

Master - City

Fields	Datatype	Description
cityId	Int(11)	Primarykey, Auto Increment
cityName	Varchar(60)	
destrictId	Int(11)	Foreignkey
Flag	Int(11)	Define as 1
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Master - Designation

Fields	Datatype	Description
designationId	Int(11)	Primarykey, Auto Increment
designationName	Varchar(60)	
destrictId	Int(11)	Foreignkey
Flag	Int(11)	Define as 1
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	

updatedBy	Varchar(20)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Tables : – Vaccination Schedule

Fields	Datatype	Description
vaccineScheduleId	Int(11)	Primarykey, Auto Increment
childRegistrationNumber	Varchar(200)	Foreignkey
vaccineId	Int(11)	Foreignkey
dateOfVaccination	Date	
Flag	Int(11)	Define as 0
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Tables : – Vaccinated

Fields	Datatype	Description
vaccinatedId	Int(11)	Primarykey, Auto Increment
vaccineId	Int(11)	Foreignkey
vaccinationStaus	Int(11)	Defined as 0
childRegistrationNumber	Varchar(200)	Foreignkey

dateOfVaccination	Date	
DoctorId	Int(11)	Foreignkey
staffId	Int(11)	Foreignkey
vaccineScheduleId	Int(11)	Foreignkey
insertedBy	Varchar(20)	
insertedDate	Datetime	
insertedIp	Varchar(20)	
updatedBy	Varchar(20)	
updatedDate	Datetime	
updatedIp	Varchar(20)	

Database Name - Baby Immunization System

1. User/Admin Login-

The screenshot shows the 'login' table structure in the 'babymunizationsystemdb' database. The table has 10 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Sno	int(11)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	userId	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	userType	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
4	userPassword	longtext	utf8mb4_general_ci		No	None			Change Drop More
5	changedUserPassword	longtext	utf8mb4_general_ci		No	'0'			Change Drop More
6	sessionId	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
7	isActive	int(11)			No	0			Change Drop More
8	isAlreadyLogin	int(11)			No	0			Change Drop More
9	currentLoginTime	datetime			No	None			Change Drop More
10	lastLoginTime	datetime			No	None			Change Drop More

2. User Login Logout DateTime Table -

The screenshot shows the 'userloginlogoutime' table structure in the 'babymunizationsystemdb' database. The table has 6 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	userType	int(11)			No	None			Change Drop More
3	userId	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
4	loginDateTime	datetime			No	None			Change Drop More
5	logoutDateTime	datetime			No	None			Change Drop More
6	ipAddress	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More

3. Hospital Registration :-

The screenshot shows the phpMyAdmin interface for the 'babyimmunizationsystemdb' database. The 'hospitalregistration' table is selected. The table structure is as follows:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	hosptialId	int(11)	utf8mb4_general_ci	No	None		AUTO_INCREMENT		Change Drop More
2	hospitalUserId	varchar(6)	utf8mb4_general_ci	No	None				Change Drop More
3	hospitalRegistrationNumber	varchar(100)	utf8mb4_general_ci	No	None				Change Drop More
4	hospitalName	text	utf8mb4_general_ci	No	None				Change Drop More
5	stateId	int(11)	utf8mb4_general_ci	No	None				Change Drop More
6	districtId	int(11)	utf8mb4_general_ci	No	None				Change Drop More
7	cityId	int(11)	utf8mb4_general_ci	No	None				Change Drop More
8	hospitalAddress	text	utf8mb4_general_ci	No	None				Change Drop More
9	contactPerson	varchar(50)	utf8mb4_general_ci	No	None				Change Drop More
10	contactNumber	varchar(10)	utf8mb4_general_ci	No	None				Change Drop More
11	pincode	varchar(6)	utf8mb4_general_ci	No	None				Change Drop More
12	hospitalEmail	varchar(100)	utf8mb4_general_ci	No	None				Change Drop More
13	insertedBy	int(10)	utf8mb4_general_ci	No	None				Change Drop More
14	insertedDateTime	datetime	utf8mb4_general_ci	No	None				Change Drop More
15	insertedIp	varchar(20)	utf8mb4_general_ci	No	None				Change Drop More
16	updatedBy	int(10)	utf8mb4_general_ci	No	None				Change Drop More
17	updatedDateTime	datetime	utf8mb4_general_ci	No	None				Change Drop More
18	updatedIp	varchar(20)	utf8mb4_general_ci	No	None				Change Drop More
19	flag	int(11)	utf8mb4_general_ci	No	1				Change Drop More

4. State -

The screenshot shows the phpMyAdmin interface for the 'babymunizationsystemdb' database. The left sidebar lists various tables: New, bloodgroup, childregistration, city, department, designation, district, doctor, hospitalregistration, login, staff, state, vaccinated, vaccine, and vaccineschedule. The 'state' table is currently selected. The main area displays the 'Table structure' for the 'state' table, which has 9 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	stateId	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	stateName	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
3	insertedBy	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
4	insertedDateTime	datetime			No	None			Change Drop More
5	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
6	updatedBy	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
7	updatedDateTime	datetime			No	None			Change Drop More
8	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
9	flag	int(11)			No	1			Change Drop More

Below the table, there are buttons for 'Check all', 'With selected:', and actions like 'Browse', 'Change', 'Drop', 'Primary', 'Unique', 'Index', 'Spatial', and 'Fulltext'. A link 'Remove from central columns' is also present.

5. District -

The screenshot shows the phpMyAdmin interface for the 'babymunizationsystemdb' database, specifically the 'district' table. The left sidebar shows other tables: New, bloodgroup, childregistration, city, department, designation, district, doctor, hospitalregistration, login, staff, state, vaccinated, vaccine, and vaccineschedule. The 'district' table is selected. The main area displays its structure, which includes 10 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	districtId	int(10)			No	None		AUTO_INCREMENT	Change Drop More
2	districtName	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
3	stateId	int(10)			No	None			Change Drop More
4	insertedBy	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
5	insertedDateTime	datetime			No	None			Change Drop More
6	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
7	updatedBy	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
8	updatedDateTime	datetime			No	None			Change Drop More
9	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
10	flag	int(11)			No	1			Change Drop More

6. City -

The screenshot shows the phpMyAdmin interface for the 'city' table in the 'babymedicinesystemdb' database. The table has 10 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	cityId	int(11)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	cityName	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
3	districtId	int(11)	utf8mb4_general_ci		No	None			Change Drop More
4	insertedBy	int(10)	utf8mb4_general_ci		No	None			Change Drop More
5	insertedDateTime	datetime	utf8mb4_general_ci		No	None			Change Drop More
6	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
7	updatedBy	int(10)	utf8mb4_general_ci		No	None			Change Drop More
8	updatedDatetime	datetime	utf8mb4_general_ci		No	None			Change Drop More
9	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
10	flag	int(11)	utf8mb4_general_ci		No	1			Change Drop More

7. Department -

The screenshot shows the phpMyAdmin interface for the 'department' table in the 'babymedicinesystemdb' database. The table has 9 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	departmentId	int(11)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	departmentName	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
3	insertedBy	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
4	insertedDateTime	datetime	utf8mb4_general_ci		No	None			Change Drop More
5	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
6	updatedBy	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
7	updatedDateTime	datetime	utf8mb4_general_ci		No	None			Change Drop More
8	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
9	flag	int(11)	utf8mb4_general_ci		No	1			Change Drop More

8. Designation -

The screenshot shows the 'designation' table structure in phpMyAdmin. The table has 9 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	designationId	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	designationName	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
3	insertedBy	int(11)			No	None			Change Drop More
4	insertedDate	date			No	None			Change Drop More
5	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
6	updatedBy	int(11)			No	None			Change Drop More
7	updatedDate	date			No	None			Change Drop More
8	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
9	flag	int(11)			No	1			Change Drop More

9. Blood Group -

The screenshot shows the 'bloodgroup' table structure in phpMyAdmin. The table has 9 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	bloodGroupId	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	bloodGroupName	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
3	flag	int(11)			No	1			Change Drop More
4	insertedBy	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
5	insertedDateTime	timestamp			No	current_timestamp()	ON UPDATE CURRENT_TIMESTAMP()		Change Drop More
6	insertedIp	varchar(11)	utf8mb4_general_ci		No	None			Change Drop More
7	updatedBy	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
8	updatedIp	varchar(11)	utf8mb4_general_ci		No	None			Change Drop More

10. Child Registration -

Table: childregistration									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	childId	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	childRegistrationNumber	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
3	childName	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change Drop More
4	childMotherName	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
5	childFatherName	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
6	gender	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
7	bloodGroupId	int(11)			No	None			Change Drop More
8	dateOfBirth	date			No	current_timestamp()			Change Drop More
9	timeOfBirth	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
10	contactNumber	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
11	email	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
12	stateId	int(11)			No	None			Change Drop More
13	districtId	int(11)			No	None			Change Drop More
14	cityId	int(11)			No	None			Change Drop More
15	pincode	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
16	address	text	utf8mb4_general_ci		No	None			Change Drop More
17	admissionDate	date			No	None			Change Drop More
18	referringDoctorName	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change Drop More
19	consultantDoctorName	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
20	birthHospitalName	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
21	vaccineId	int(11)			No	None			Change Drop More
22	insertedBy	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
23	insertedDateTime	datetime			No	None			Change Drop More
24	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
25	updatedBy	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
26	updatedDateTime	datetime			No	None			Change Drop More
27	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
28	isActive	int(11)			No	1			Change Drop More

11. Staff -

Server: 127.0.0.1:3325 » Database: babymedicationsystemdb » Table: staff

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	slid	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	staffId	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
3	staffName	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
4	staffFatherName	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
5	staffGender	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
6	designationId	int(11)			No	None			Change Drop More
7	dateOfBirth	date			No	None			Change Drop More
8	contactNumber	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
9	email	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
10	stateId	int(11)			No	None			Change Drop More
11	districtId	int(11)			No	None			Change Drop More
12	cityId	int(11)			No	None			Change Drop More
13	pincode	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
14	localAddress	text	utf8mb4_general_ci		No	None			Change Drop More
15	permanentAddress	text	utf8mb4_general_ci		No	None			Change Drop More
16	joiningDate	datetime			No	None			Change Drop More
17	image	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
18	departmentId	int(11)			No	None			Change Drop More
19	isActive	int(11)			No	1			Change Drop More
20	insertedBy	int(10)			No	None			Change Drop More
21	insertedDateTime	datetime			No	None			Change Drop More
22	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
23	updatedBy	int(10)			No	None			Change Drop More
24	updatedDateTime	datetime			No	None			Change Drop More
25	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More

12. Doctor -

Server: 127.0.0.1:3325 » Database: babymunizationsystemdb » Table: doctor

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	doctorId	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
3	doctorName	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
4	gender	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
5	designationId	int(11)			No	None			Change Drop More
6	qualification	varchar(60)	utf8mb4_general_ci		No	None			Change Drop More
7	dateOfBirth	date			No	None			Change Drop More
8	contactNumber	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
9	email	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
10	stateId	int(11)			No	None			Change Drop More
11	districtId	int(11)			No	None			Change Drop More
12	cityId	int(11)			No	None			Change Drop More
13	pincode	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
14	localAddress	text	utf8mb4_general_ci		No	None			Change Drop More
15	permanentAddress	text	utf8mb4_general_ci		No	None			Change Drop More
16	joiningDate	date			No	None			Change Drop More
17	image	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
18	departmentId	int(11)			No	None			Change Drop More
19	isActive	int(11)			No	1			Change Drop More
20	insertedBy	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
21	insertedDateTime	datetime			No	None			Change Drop More
22	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
23	updatedBy	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
24	updatedDateTime	datetime			No	None			Change Drop More
25	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More

13. Vaccine -

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	serialNumber	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	vaccineId	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
3	vaccineName	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
4	prevent	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
5	physicalForm	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
6	composition	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
7	brand	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
8	packagingType	varchar(60)	utf8mb4_general_ci		No	None			Change Drop More
9	doses	int(11)			No	None			Change Drop More
10	flag	int(11)			No	1			Change Drop More
11	packagingDose	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
12	typeOfMedicine	varchar(60)	utf8mb4_general_ci		No	None			Change Drop More
13	insertedBy	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
14	insertedDateTime	datetime			No	None			Change Drop More
15	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
16	updatedBy	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
17	updatedDateTime	datetime			No	None			Change Drop More
18	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More

14. Vaccination Schedule –

The screenshot shows the phpMyAdmin interface for the 'vaccineschedule' table in the 'babymedicationsystemdb' database. The table has 11 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	vaccineScheduledId	int(11)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	childRegistrationNumber	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
3	vaccineId	varchar(6)	utf8mb4_general_ci		No	None			Change Drop More
4	dateOfVaccination	date			No	None			Change Drop More
5	flag	int(11)			No	0			Change Drop More
6	insertedBy	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
7	insertedDateTime	datetime			No	None			Change Drop More
8	insertedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
9	updatedBy	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
10	updatedDateTime	datetime			No	None			Change Drop More
11	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More

Buttons at the bottom include: Check all, With selected:, Browse, Change, Drop, Primary, Unique, Index, Spatial, Fulltext, and Remove from central columns.

15. Vaccinated –

The screenshot shows the phpMyAdmin interface for the 'vaccinated' table in the 'babymedicationsystemdb' database. The table has 12 columns:

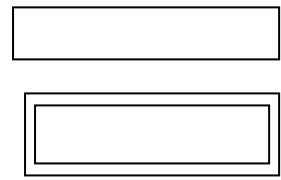
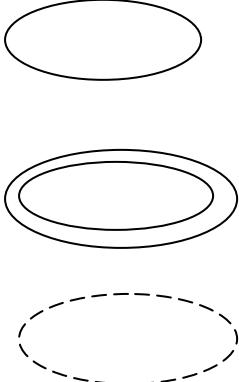
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	vaccinatedId	int(11)	utf8mb4_general_ci		No	None		AUTO_INCREMENT	Change Drop More
2	vaccinationStatus	int(11)			No	None			Change Drop More
3	childId	int(11)			No	None			Change Drop More
4	vaccinatedDate	date			No	None			Change Drop More
5	doctorId	int(11)			No	None			Change Drop More
6	reason	text	utf8mb4_general_ci		No	None			Change Drop More
7	insertedBy	int(11)			No	None			Change Drop More
8	insertedDateTime	datetime			No	None			Change Drop More
9	insertedIp	int(20)			No	None			Change Drop More
10	updatedBy	int(11)			No	None			Change Drop More
11	updatedDateTime	datetime			No	None			Change Drop More
12	updatedIp	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More

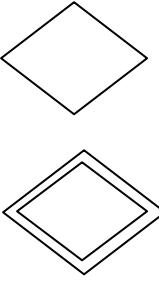
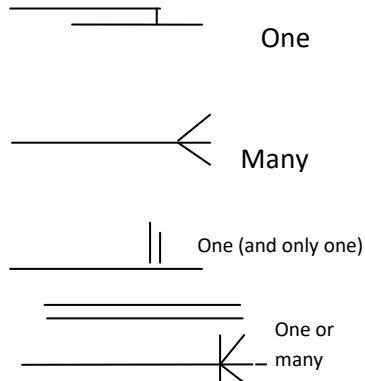
Buttons at the bottom include: Check all, With selected:, Browse, Change, Drop, Primary, Unique, Index, Spatial, Fulltext, and Remove from central columns.

6.3 E-R DIAGRAM

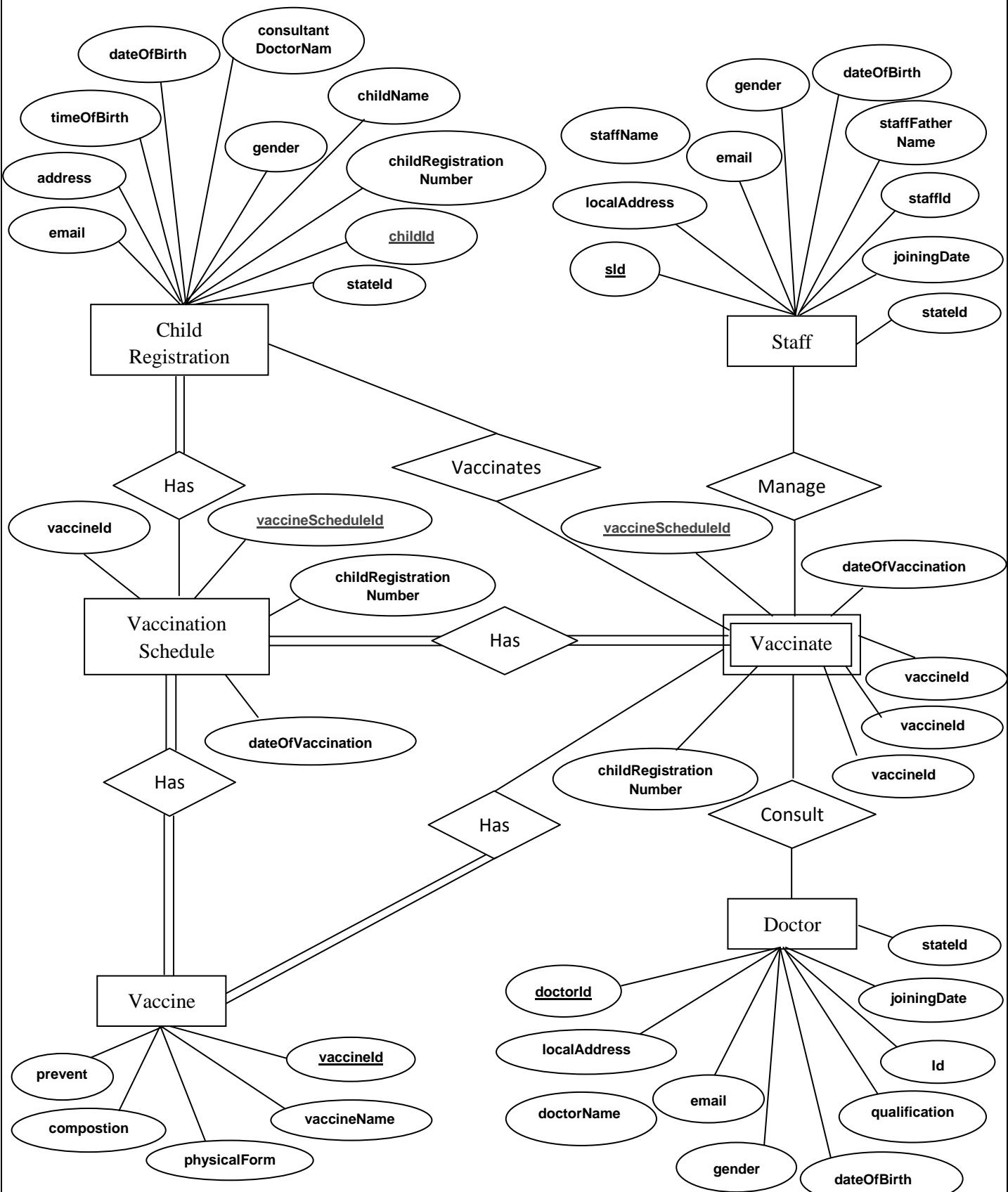
In software engineering, an entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modelling is a method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams, ER diagrams, or ERDs.

The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modelling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in turn is mapped to a physical model during physical design. Note that sometimes, both of these phases are referred to as "physical design"

Entities	<ol style="list-style-type: none">Strong Entity: exist independently from other entity types.Weak Entity: depend on some other entity type.Associative Entity: are entities that associate the instance of one or more entity types.	
*Attributes	<ol style="list-style-type: none">Attributes are characteristics of either an entity, a many-to-many relationship, or a one-to-one relationship.Multivalued attributes are those that are capable of taking on more than one value.Derived attributes are attributes whose value can be calculated from related attributes values.	

Relationship	<p>Relationship are meaningful associations between or among entities. They are usually verbs, e.g. assign, associate, or track. A relationship provides useful information that could not be discerned with just the entity types.</p> <p>Weak Relationship, or identifying relationships are connections that exist between a weak entity type and its owner.</p>	
Link	<p>It refers to the maximum number of times an instance in one entity can be associated with instances in the related entity, and the maximum number of times an instance in one entity can be associated with an instance in the related entity. Cardinality and Ordinality are represented by the styling of a line and its endpoint, denoted by the chosen notation style.</p>	

E-R DIAGRAM FOR PROJECT:-

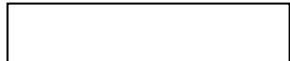


6.4 DATA FLOW DIAGRAM

A **data flow diagram (DFD)** is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing(structured design).

A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel unlike a flowchart which also shows this information.

Context Diagram (Data Flow Diagram)

Process	A process is some form of processing or data transformation which takes data as input, does something to it, and provides Outputs.	
Data Store	A data store is where a process stores data between process for later retrieval By that same process or other one. Files and tables are considered data store.	
Entity	The entity symbol represents source of data to the system or destination of data from the system. It represents by a rectangle, entities include End User, purchasing department, and Inventory system.	
Data Flow	Data flow represents with a line an arrow Head on one end. A fork in a data flow means that the same data goes to two separates destinations. The same data coming from several locations can also be joined. Flows define the interfaces between the components within the system, and its external entities.	

6.4.1 Context Level DFD:

The major objective of a context diagram is to provide graphical representation of the whole system. As stated earlier it consists of all the system boundaries, all the external entities that interact with the system and major data flow between the entities in the system.

Components of DFD

The Data Flow Diagram has 4 components:

- **Process**

Input to output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle or a circle. The process is named a short sentence, in one word or a phrase to express its essence

- **DataFlow**

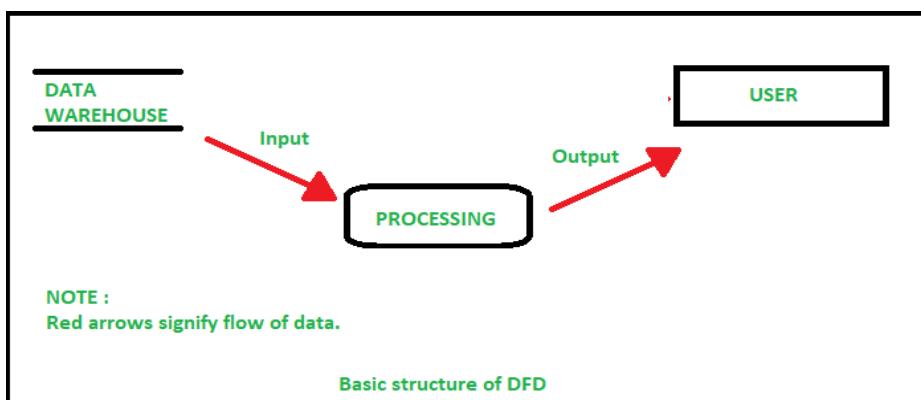
Data flow describes the information transferring between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved. Data flow also represents material along with information that is being moved. Material shifts are modeled in systems that are not merely informative. A given flow should only transfer a single type of information. The direction of flow is represented by the arrow which can also be bi-directional.

- **Warehouse**

The data is stored in the warehouse for later use. Two horizontal lines represent the symbol of the store. The warehouse is simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, a filing cabinet. The data warehouse can be viewed independent of its implementation. When the data flow from the warehouse it is considered as data reading and when data flows to the warehouse it is called data entry or data updation.

- **Terminator**

The Terminator is an external entity that stands outside of the system and communicates with the system. It can be, for example, organizations like banks, groups of people like customers or different departments of the same organization, which is not a part of the model system and is an external entity. Modeled systems also communicate with terminator.



Rules for creating DFD

The name of the entity should be easy and understandable without any extra Request For Login

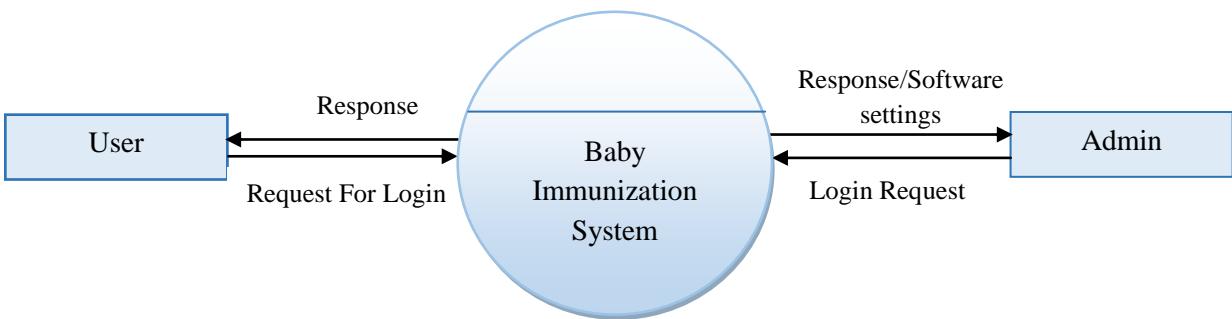
- assistance(like comments).
- The processes should be numbered or put in ordered list to be referred easily.
- The DFD should maintain consistency across all the DFD levels.
- A single DFD can have maximum processes upto 9 and minimum 3 processes.

Levels of DFD

DFD uses hierarchy to maintain transparency thus multilevel DFD's can be created. Levels of DFD are as follows:

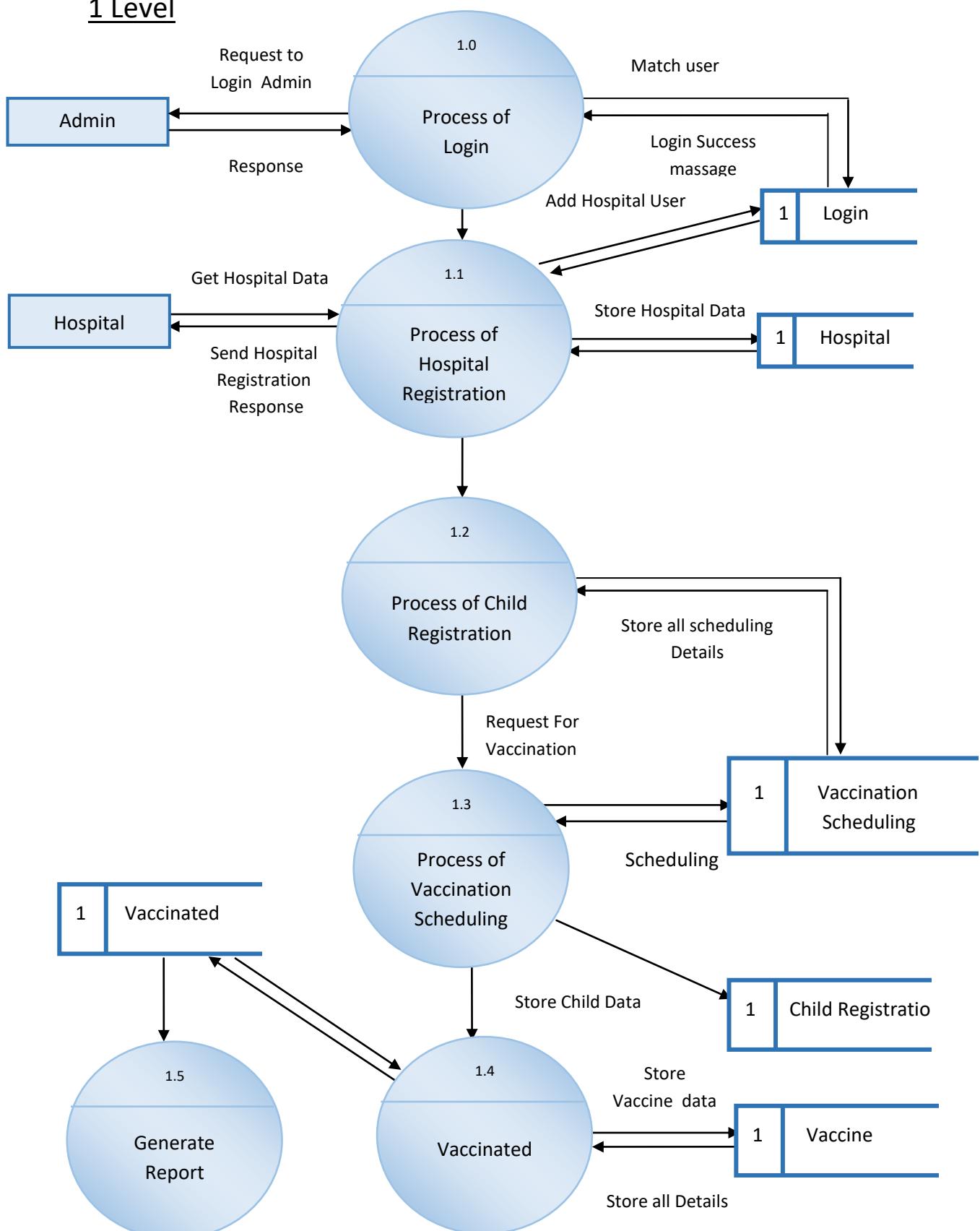
- 0-level DFD
- 1-level DFD:
- 2-level DFD:

6.4.2. ZERO Level Data Flow Diagram for Project View

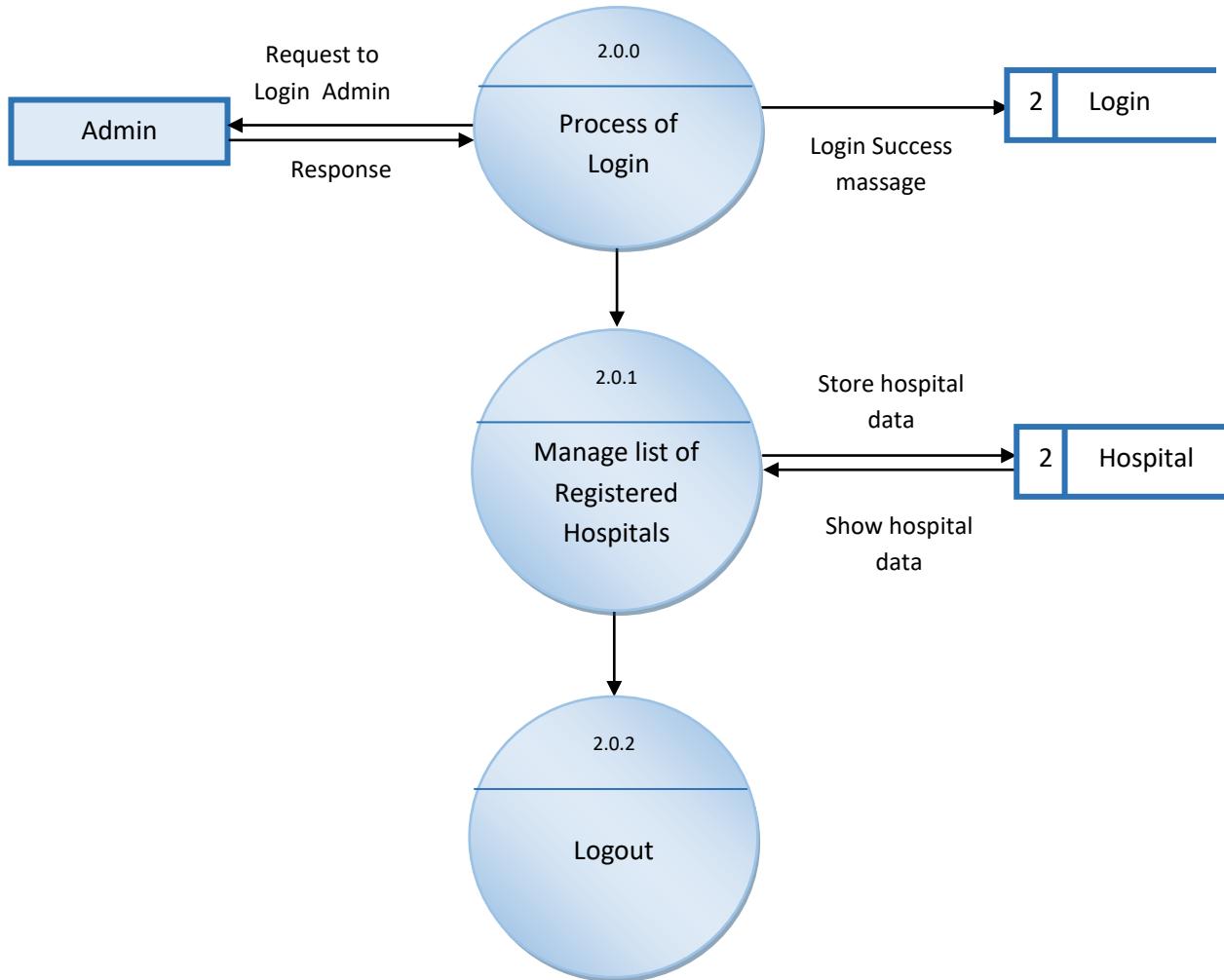


Level 1 Diagram For User

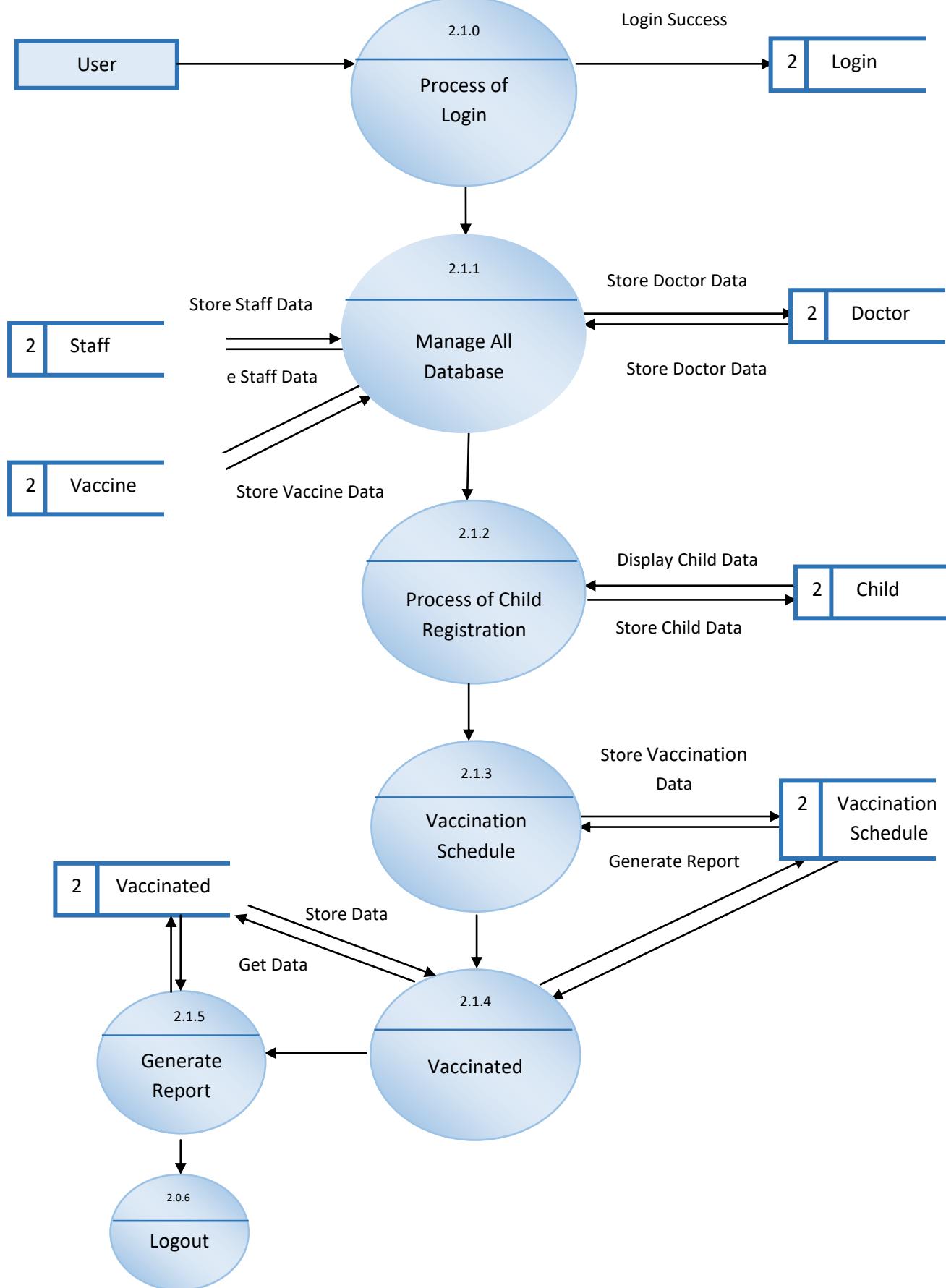
1 Level



Level 2 Diagram For Admin



Level 2 Diagram For User



6.5 USE CASE DIAGRAM

6.5.1 Introduction - A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Website. Use case diagrams are employed in UML (Unified Modelling Language), a standard notation for the modelling of real-world objects and systems. A use case diagram contains four components.

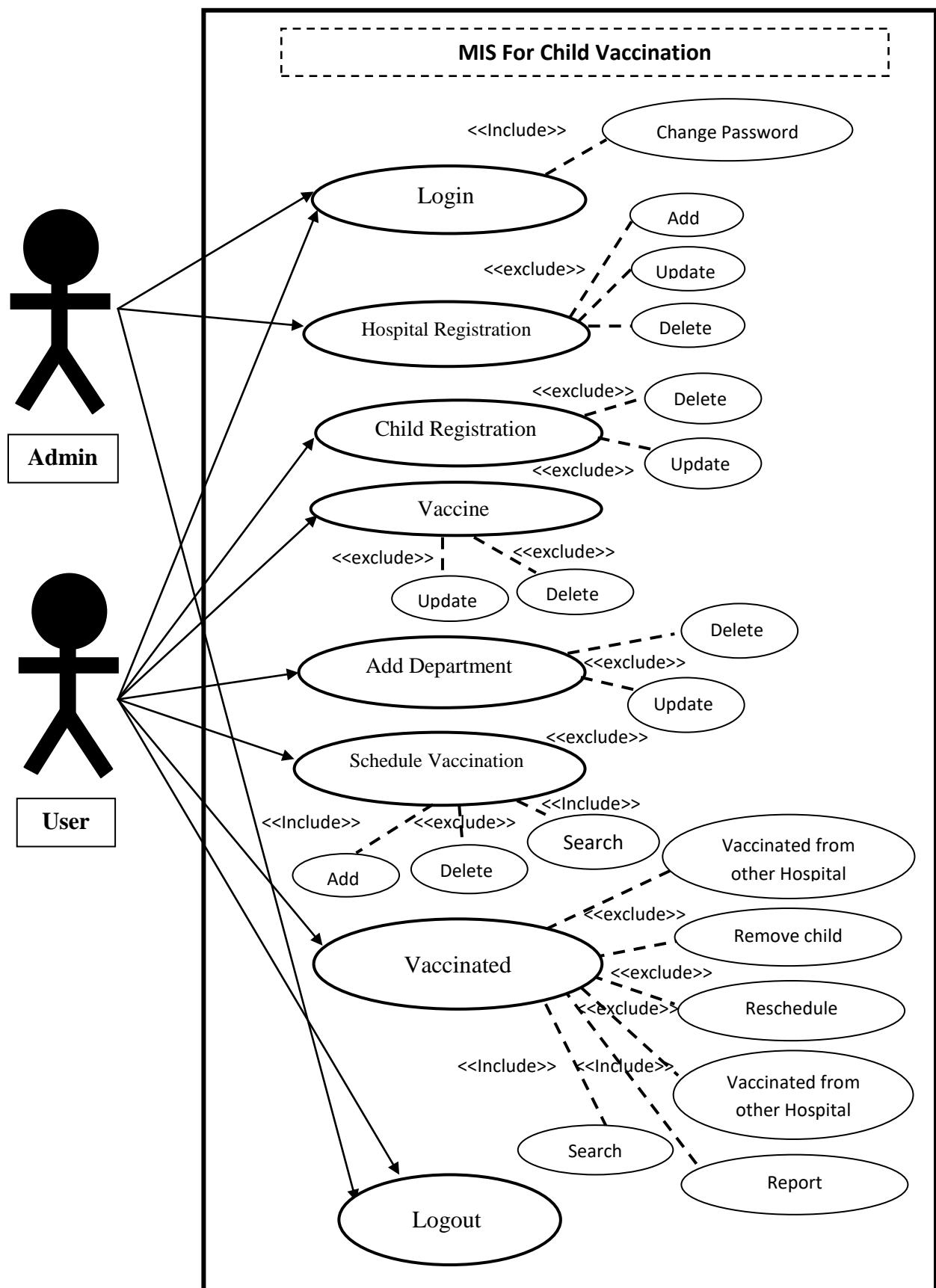
- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which the specific roles are played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

Use Case Diagram

Actor	An actor represents a role that an outsider takes on when interacting with the business system. For instance, an actor can be a customer, a business partner, a supplier, or another business system.	 Actor
Association	An association is the relationship between an actor and a business use case. It indicates that an actor can use a certain functionality of the business system-the business use case:	<hr/> <hr/>
Business Use Case	A business use case describes the interaction between an actor and a business system, meaning it describes the functionality of the business system that the actor utilizes.	 Business Use Case

Subject	A subject describes a business system that has one or more business use cases attached to it.	Subject
---------	---	---------

6.5.2 USE CASE DIAGRAM FOR USER



6.6 FLOW CHART

A flow chart is a visual representation of the sequence of steps and decisions needed to perform a process. Each step in the sequence is noted within a diagram shape. Steps are linked by connecting lines and directional arrows. This allows anyone to view the flowchart and logically follow the process from beginning to end.

A flowchart is a powerful business tool. With proper design and construction, it communicates the steps in a process very effectively and efficiently.

Common Flowchart Symbols

Different flow chart symbols have different meanings. The most common flow chart symbols are:

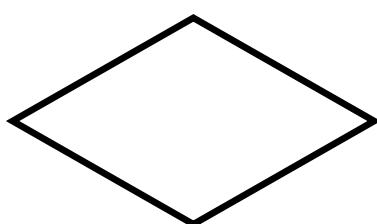
1. **Terminator:** An oval flow chart shape indicating the start or end of the process.



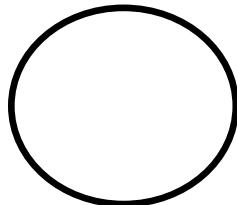
2. **Process:** A rectangular flow chart shape indicating a normal process flow step.



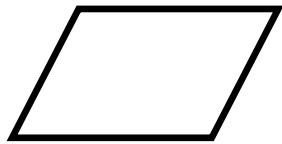
3. **Decision:** A diamond represents a decision or branching point. Lines coming out from the diamond indicate different possible situations, leading to different sub-processes.



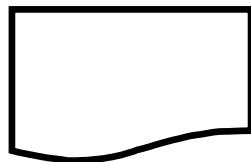
Connector: A small, labeled, circular flow chart shape used to indicate a jump in the process flow.



Data: A parallelogram that indicates data input or output (I/O) for a process.



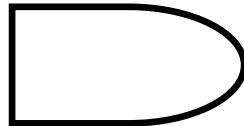
Document: This represents a printout, such as a document or a report.



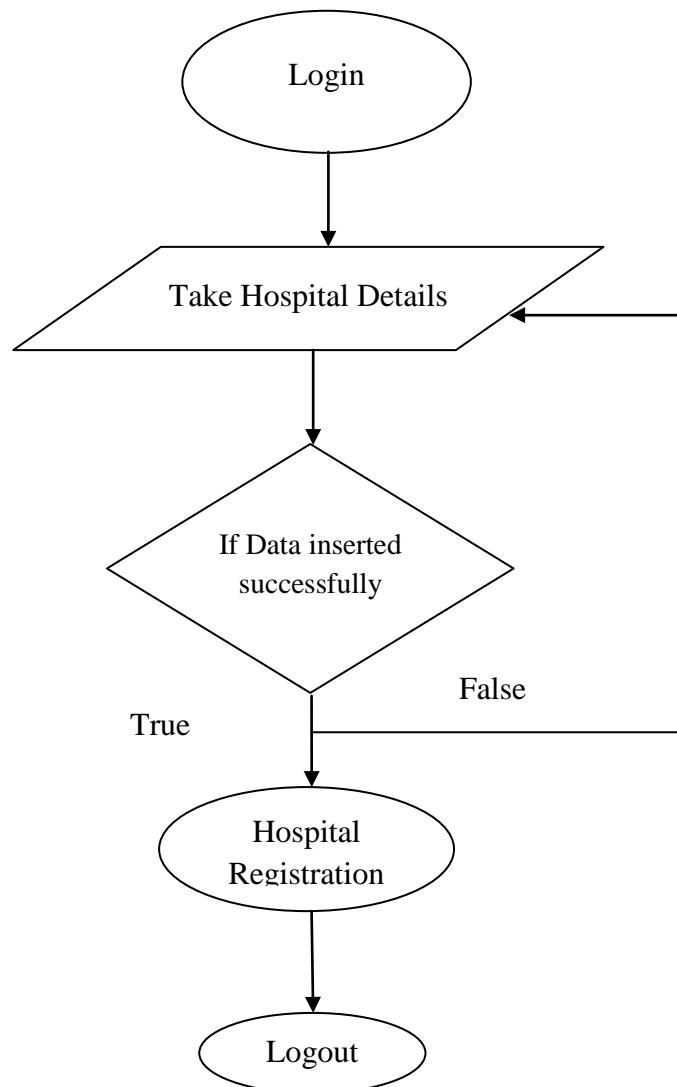
Flow line: Lines represent flow of the sequence and direction of a process.



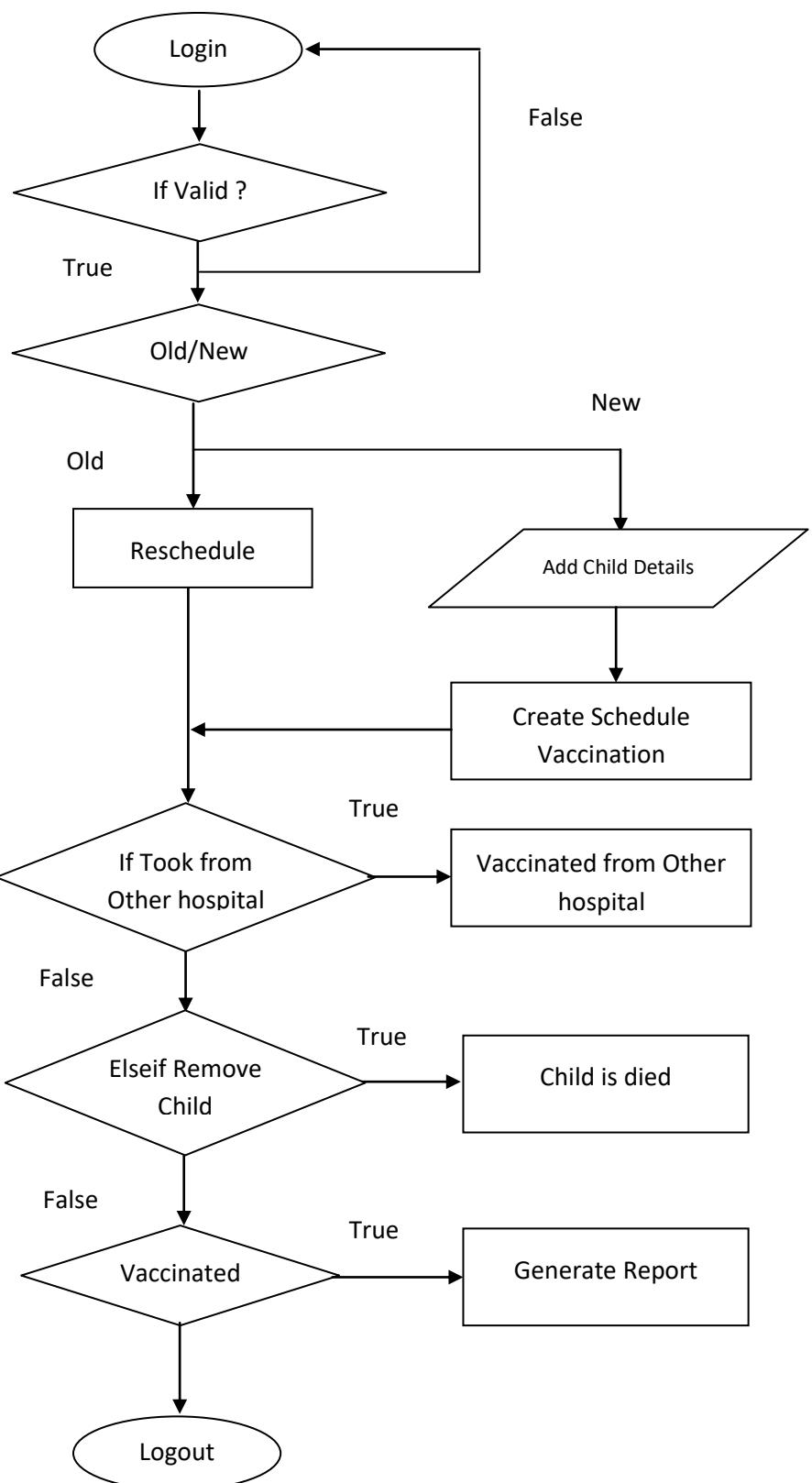
Delay or Bottleneck: Identifies a delay or a bottleneck.



6.6.1 FLOW CHART FOR ADMIN –



6.6.2 FLOW CHART FOR USER -



6.7 ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements.

The purpose of an activity diagram can be described as –

- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system.

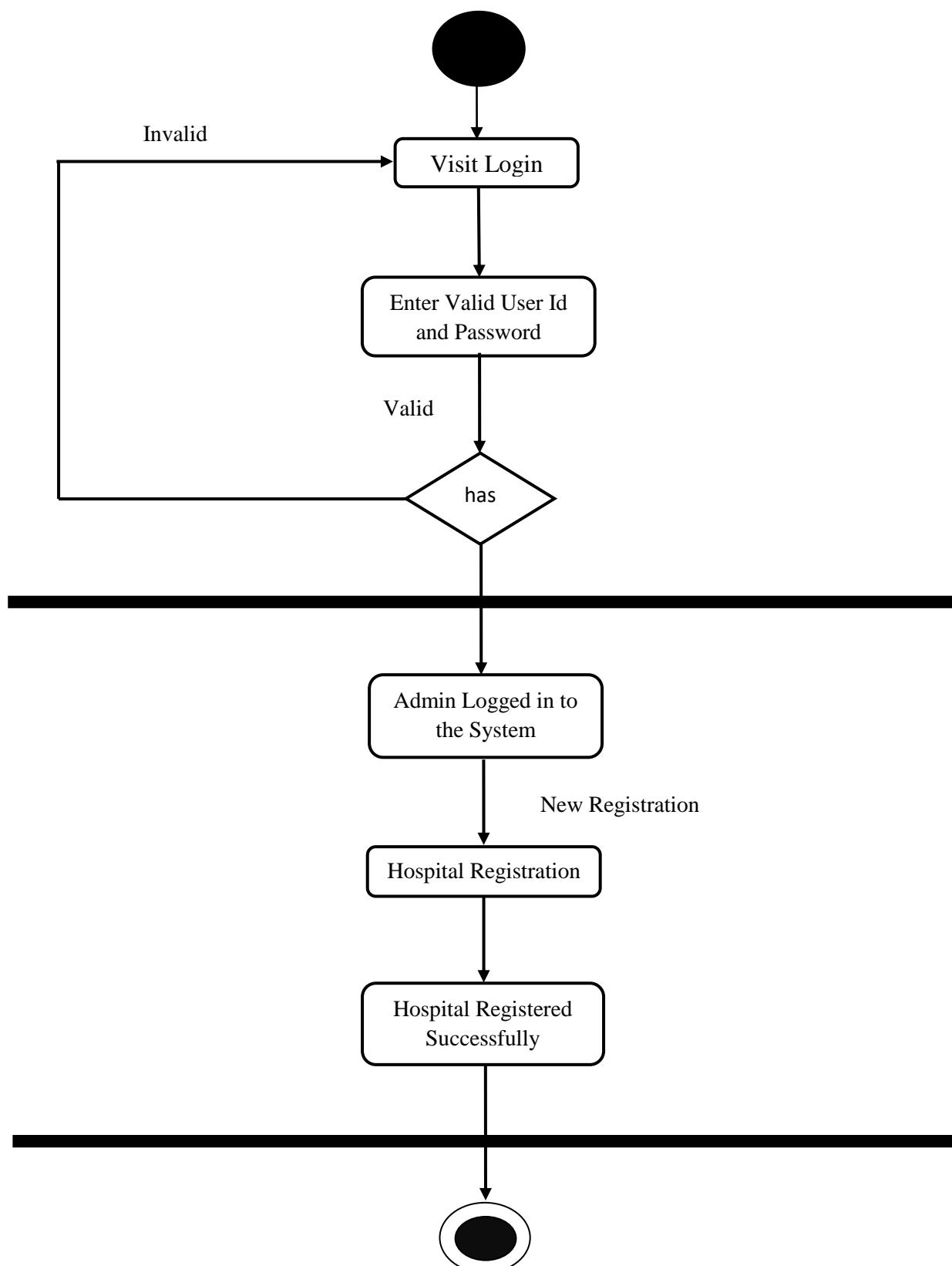
Activity diagrams are mainly used as a flowchart that consists of activities performed by the system. Activity diagrams are not exactly flowcharts as they have some additional capabilities. These additional capabilities include branching, parallel flow, swimlane, etc.

Before drawing an activity diagram, we must have a clear understanding about the elements used in activity diagram. The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities, we need to understand how they are associated with constraints and conditions.

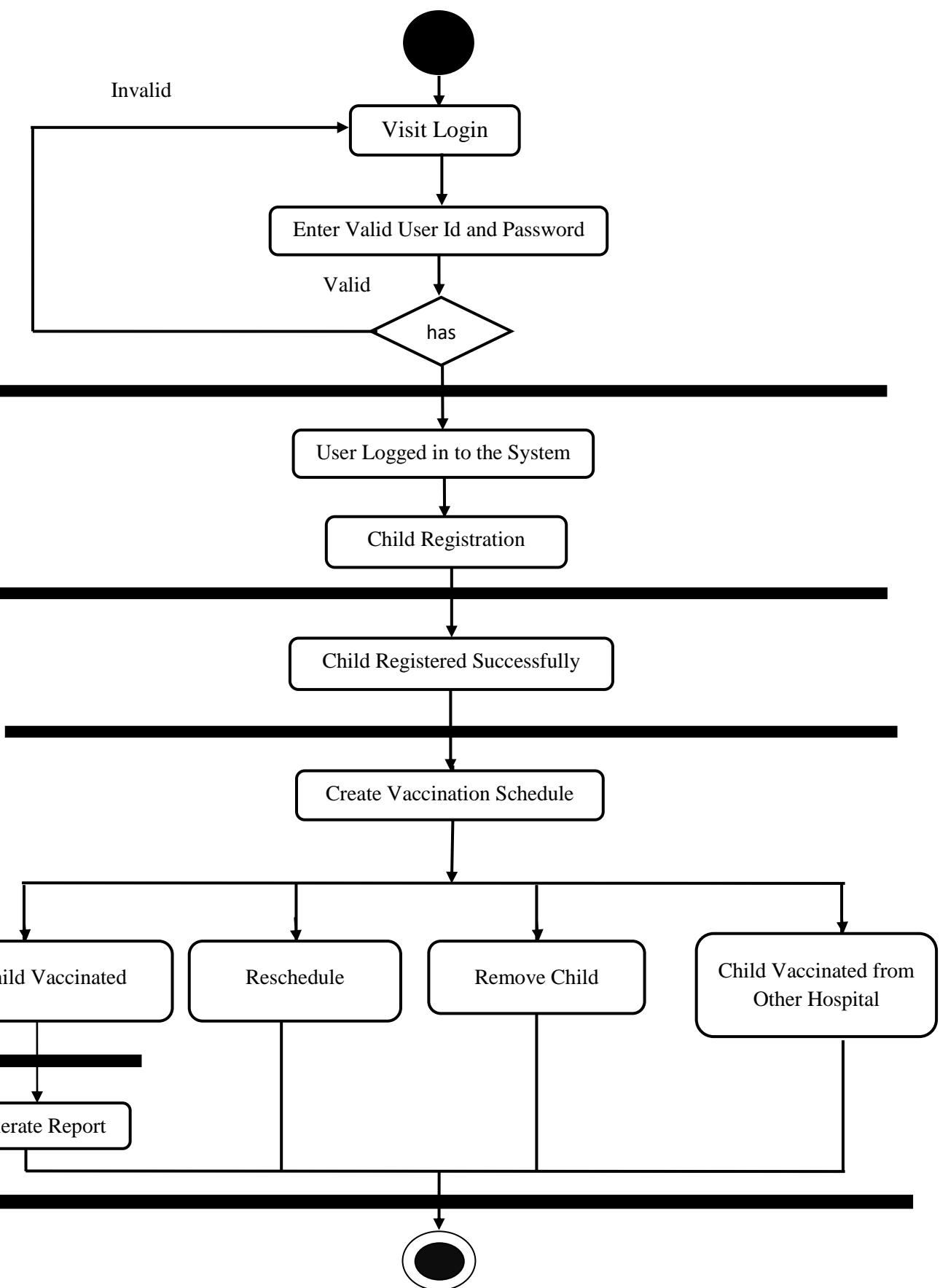
Before drawing an activity diagram, we should identify the following elements –

- Activities
- Association
- Conditions
- Constraints

6.7.1 ACTIVITY DIAGRAM FOR ADMIN



6.7.2. ACTIVITY DIAGRAM FOR USER



6.8. SEQUENCE DIAGRAM

6.8.1 SEQUENCE DIAGRAM FOR ADMIN

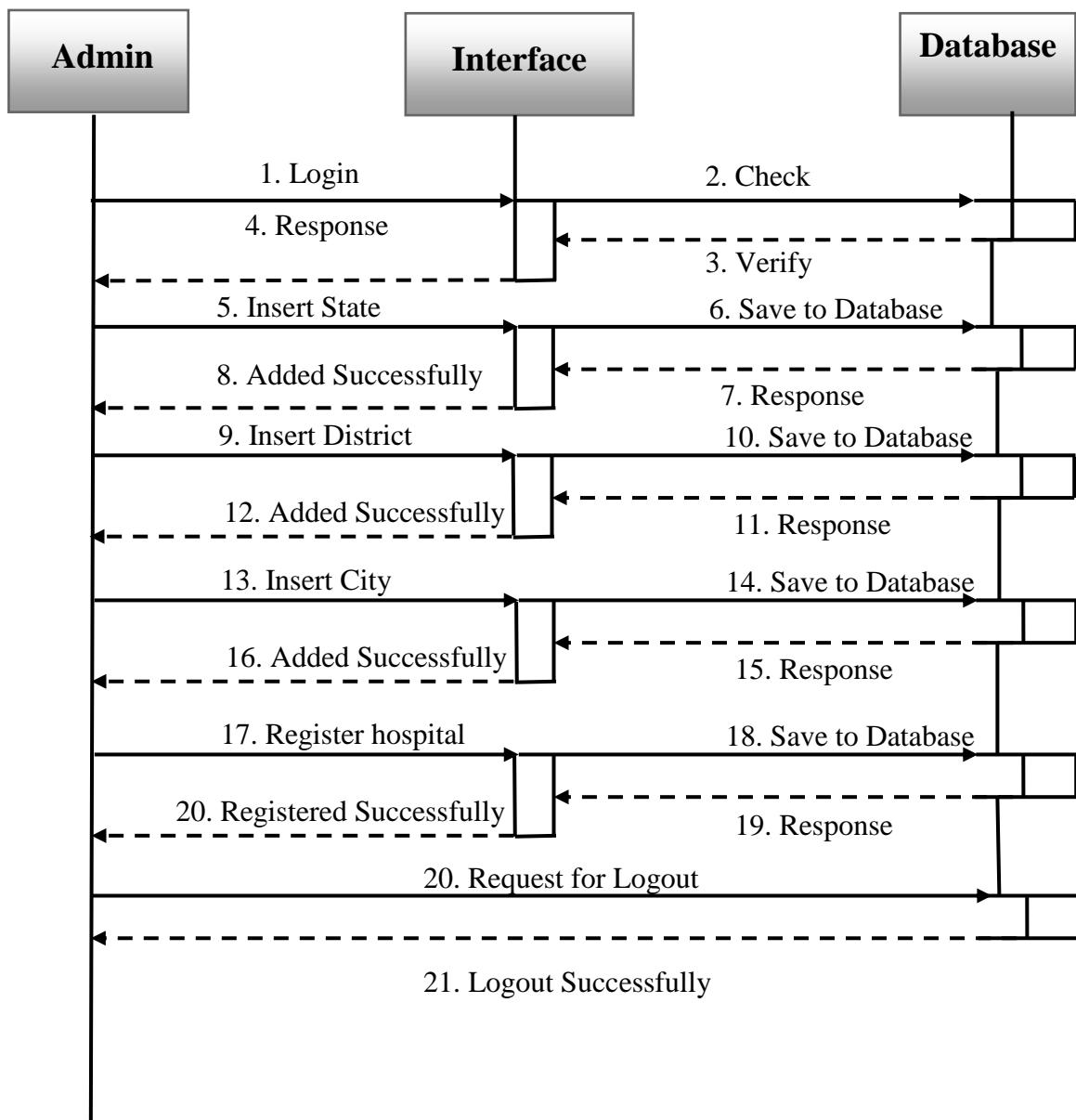


Figure 6.8.1 Sequential Diagram

6.8.2 SEQUENCE DIAGRAM FOR USER

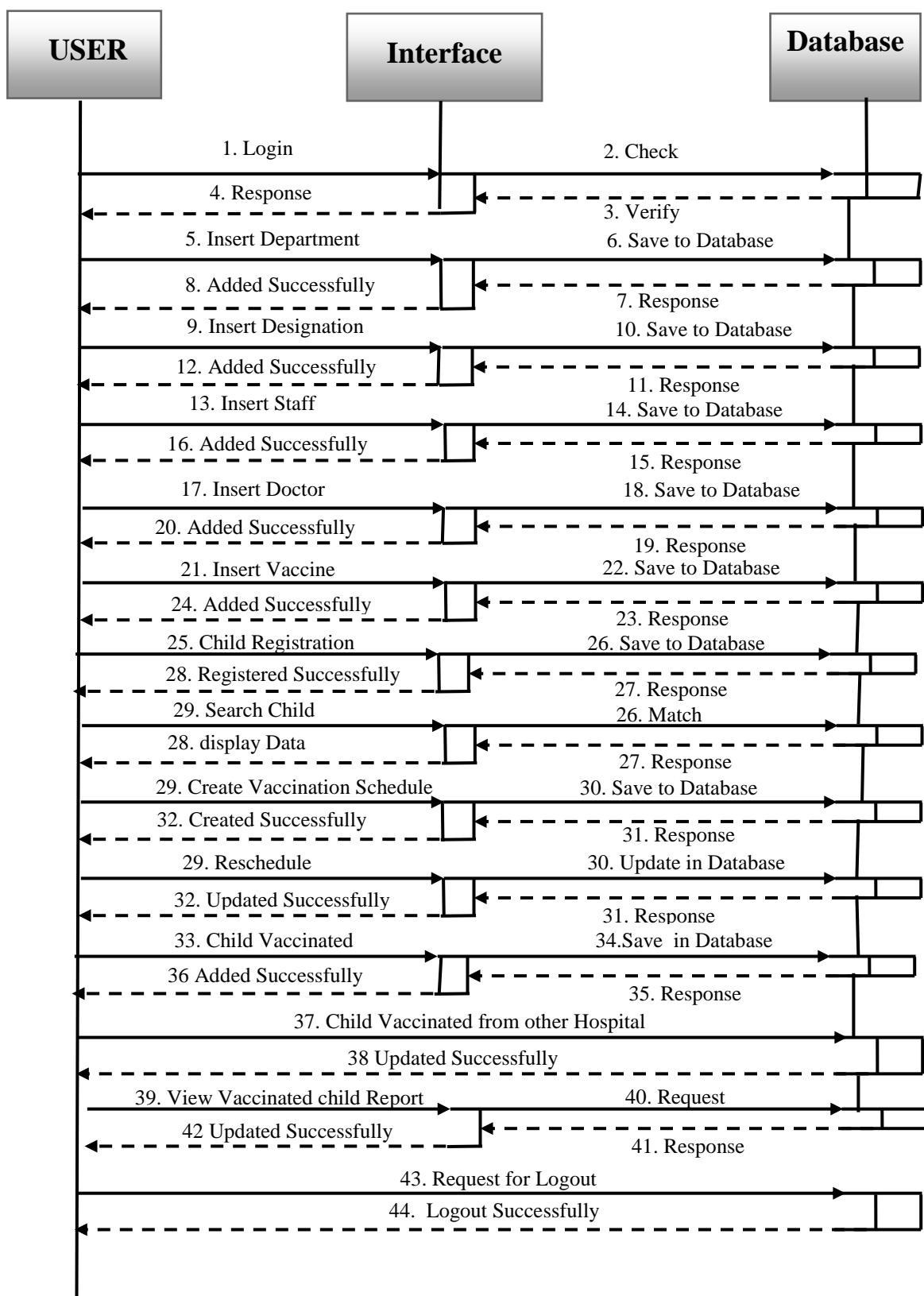


Figure 6.8.2 Sequential Diagram

SYSTEM TESTING AND QUALITY MEASURMENT

CRITERION

7.1 Introduction:

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software and application.

Psychology of Testing

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Testing is the process of executing a program with the intent of finding errors.

Testing Objectives:

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time.

Testing is a process of executing a program with the intent of finding error. A successful test is one that uncovers an as yet undiscovered error.

A good best case is one that has a high probability of finding error, if it exists. The tests are inadequate to detect possibly present errors.

The software more or less confirms to the quality and reliable standards.

Testing case design:

A rich variety of test case design methods have evolved for software. These methods provide the developer with a systematic approach to testing. More important, methods provide the highest likely hood of uncovering errors in software.

Any engineered product can be tested in one of the two ways:

1. Knowing the specified function in the application has been designed to perform.
2. Knowing the internal workings of an application.



Client Needs



Acceptance Testing



Requirements



System Testing



Design



Integration Testing



Code



Unit Testing



Black Box Testing:

Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. Black box testing, which is also known as behavioural, opaque-box, closed-box, specification-based or eye-to-eye testing, is a Software Testing method that analyses the functionality of a software/application without knowing much about the internal structure/design of the item that is being tested and compares the input value with the output value.

Types of Black Box Testing

Practically, there are several types of black box testing that are possible but if we consider the major variant of it then below mentioned are the two fundamental ones.

1) Functional Testing

This type deals with the functional requirements or specifications of an application. Here, different actions or functions of the system are being tested by providing the input and comparing the actual output with the expected output.

Functional Testing Types

It has many categories and these can be used based on the scenario.

13. Unit Testing: Unit testing is usually performed by the developer who writes different code units that could be related or unrelated to achieve a particular functionality.

Therefore, this usually entails writing unit tests which would call the methods in each unit and validate that when the needed parameters are passed, its return value is as expected. Code coverage is an important part of unit testing where test cases need to exist to cover the below three:

- Line coverage
- Code path coverage
- Method coverage

What is Non-Functional Testing?

Non-functional testing is done to verify the non-functional requirement i.e. Performance, Usability etc. It verifies if the behaviour of the system is as per the requirement or not.

Checklist

A checklist is used to ensure that no important aspect is left without testing.

Checklist for Performance Testing:

14. Response time of the application should be verified i.e. how long does it take to load the application, any input given to the application provides the output in how much time, refreshing the browser etc.
15. Throughput should be verified for the number of transactions completed during a load test.

16. Environment set up should be the same as the live environment or else the results would not be the same.
17. Process time – Process activities like import & export of excel, any calculations in the application should be tested.
18. Interoperability should be verified i.e. a software should be able to inter-operate with the other software's or systems.
19. ETL time should be verified i.e. time taken in extracting, transforming and loading the data from one database to another.
20. Increasing Load on the application should be verified.

Checklist for Security testing:

21. Authentication: Only the authentic user should be able to Log in.
22. Authorized: User should be able to log into those modules only for which he is authorized or for which the user has been provided access to.
23. Password: Password requirement should be verified i.e. password should be as per how the requirement defines i.e. length, special characters, numbers etc.
24. Timeout: If the application is inactive then it should timeout in a specified time.
25. Data Backup: Data backup should be taken at a specified time and should be copied to a secured location.
26. Internal links to the web application should not be accessible if placed directly in the browser.
27. All the communication should be encrypted.

Checklist for Documentation Testing:

28. User & System documentation.
29. Documents for training purpose.

Black Box Testing Techniques

1) Equivalence Partitioning:

This technique is also known as Equivalence Class Partitioning (ECP). In this technique, input values to the system or application are divided into different classes or groups based on its similarity in the outcome.

For Example: Equivalence Class Partitioning

Phone Number

*Accept only 10 numbers

INVALID

<=11

VALID

==10

2) Error Guessing:

This is a classic example of experience based testing.

Few common mistakes that developers usually forget to handle:

30. Divide by zero.
31. Handling null values in text fields.
32. Accepting Submit button without any value.
33. File upload without attachment.
34. File upload with less than or more than the limit size.

Advantages:

35. The tester need not have a technical background. It is important to test by being in the user's shoes and think from the user's point of view.
36. Testing can be started once the development of the project/application is done. Both the testers and developers work independently without interfering in each other's space.
37. It is more effective for large and complex applications.
38. Defects and inconsistencies can be identified at the early stage of testing.

Disadvantages:

39. Without any technical or programming knowledge, there are chances of ignoring possible conditions of the scenario to be tested.
40. In a stipulated time there are possibilities of testing less and skipping all possible inputs and their output testing.
41. A Complete Test Coverage is not possible for large and complex projects.

White box testing

If we go by the definition, “White box testing” (also known as clear, glass box or structural testing) is a testing technique which evaluates the code and the internal structure of a program.

White box testing involves looking at the structure of the code. When you know the internal structure of a product, tests can be conducted to ensure that the internal operations performed according to the specification.

Unit Testing:

Execution testing types of ‘Unit testing white box technique’.

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel. In this project each service can be thought of a module. There are modules like Accounts, Expenses Categories, Expenses, Incomes, Repeating Expenses, Repeating Incomes, Reports. Each module has been tested by giving different sets of inputs. The inputs are validated when accepting from the user.

Advantages of Unit Testing

1. Unit testing reduces the level of bugs in production code.
2. Unit testing saves you development time.
3. Automated tests can be run as frequently as required.

Test No.	User Id	Password	Actual Output	Expected Output
1	Wrong User Id	Wrong Password	Invalid Username And password	Show error message
2	Wrong User Id	Right Password	Invalid Username	Show error message
3	Right User Id	Wrong Password	Invalid Password	Show error message
4	Right User Id	Right Password	Login Successful	Success

System Testing

Test No.	Test Case Objective	Actual Output	Expected Output
1	Correct Work Flow?	Each module is correctly connected with each other.	Yes
2	Behaviour?	Behaviour of the system is user friendly.	User Friendly
3	Bug Free?	Changes can be made easily.	Yes

Acceptance Testing

Test No.	Test Case Objectives	Actual Output	Expected output
1	Admin Module	Working fully and correctly	Yes
2	Admin User Module (Website,Android & IOS application)	Working appropriately	User Friendly

SYSTEM IMPLEMENTATION

A crucial phase in the system life cycle is the successful implementation of the new design systems implementation. Implementation includes all those activities that take place to convert from the old system to the new system. The new system here is replacing an existing system. The proper implementation become necessary so that a reliable system based on the requirements of the organization can be provided. Successful implementation guarantees improvement in the organization working.

Finally, the implementation phase requires a lot of training to be given to the user through some means. This software fulfills the requirement of the online shopping app.

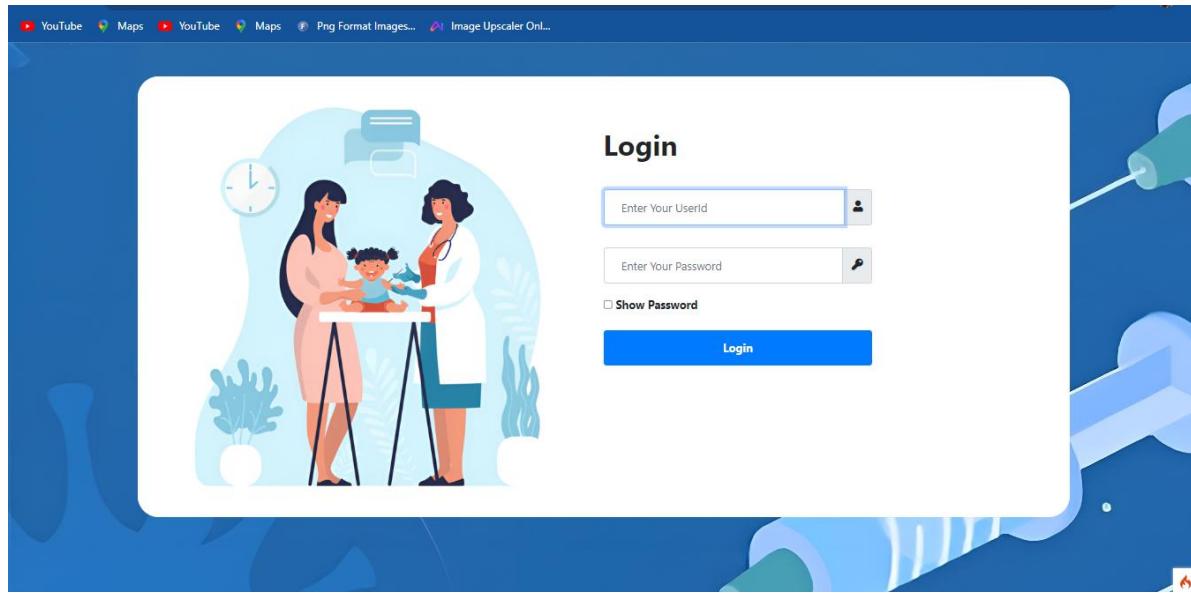
The system design of this app has been done keeping in mind all the things that used to happen in the market in the old times such as the system of storing account transactions, many customers' bills etc.

This system throw all registered User's record easily access. A user can easily login and place his order while viewing the product and the admin can easily view and update all the orders at once, delete them as well as track the delivery of order, keeping all these things in mind, the system has been implemented .Users will get better product at the **"Fruits Delivery and Management App"**.

All report publicly show.

INPUT & OUTPUT FORMS

1. Login Page –

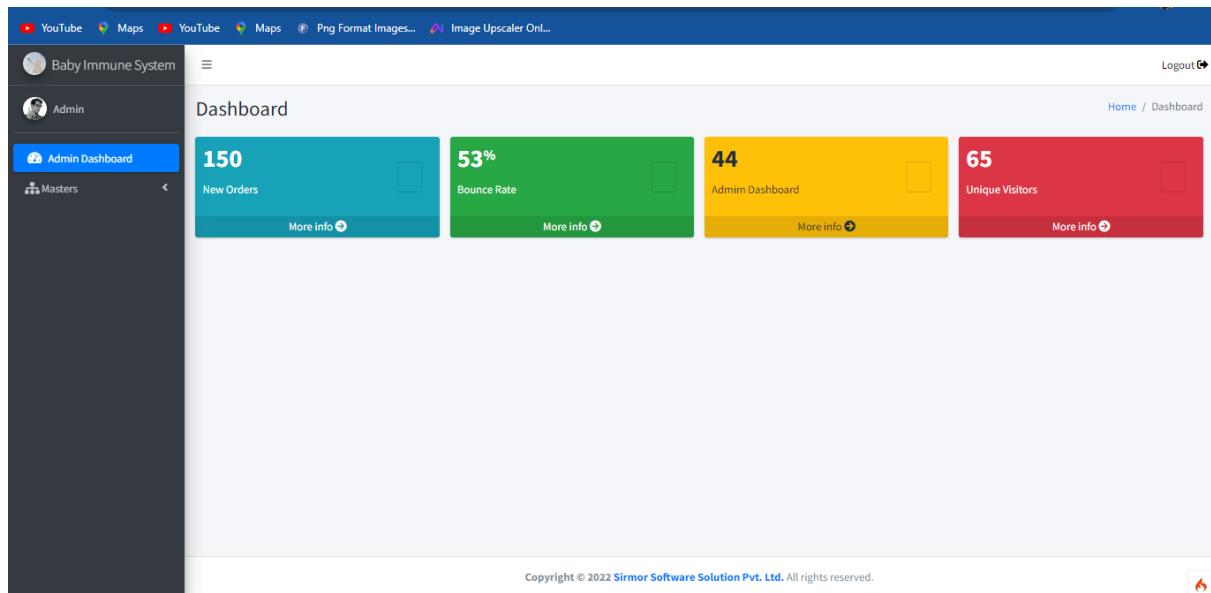


2. Change Password –

A screenshot of a 'Change User Password' page. The title 'Change User Password' is at the top. Below it is a message 'Please Update your Password'. There are two input fields: 'Enter New Password' and 'Confirm New Password', each with a key icon. A 'Show Password' checkbox is present. At the bottom is a large blue 'Submit' button.

Admin Module

3. Admin Dashboard –



4. State -

- Add State

The screenshot shows the 'Admin Dashboard' interface. On the left sidebar, there are links for 'Admin Dashboard' and 'Masters'. The main content area is titled 'State' and contains a sub-section 'Add State Master'. It features a text input field labeled 'Enter State' with the value 'Chhattisgarh'. Below the input are two buttons: 'Submit' (blue) and 'Cancel' (red). To the right of this form is a table titled 'State DataTable' with columns 'S.No.', 'State', and 'Action'. The table contains five rows with data: 1. Chhattisgarh, 2. Odisha, 3. Punjab, 4. Bihar, and 5. Andhra Pradesh. Each row has an 'Edit' icon (pencil) and a 'Delete' icon (trash can) in the 'Action' column. At the bottom of the page, there is a Windows taskbar with various icons and a system status bar showing '27°C Partly cloudy', '10:39 20-09-2022', and battery level.

- Update State

This screenshot shows the 'Admin Dashboard' with the 'State' section active. A modal window titled 'Update State' is open over the 'Add State Master' form. Inside the modal, there is an input field labeled 'State Name' containing 'Chhattisgarh'. Below the input are two buttons: 'Close' (white) and 'Update' (blue). In the background, the 'Add State Master' form is visible with its input field and buttons. Below the forms is the 'State DataTable' table, which has the same structure and data as the one in the previous screenshot.

- Delete State

This screenshot shows the 'Admin Dashboard' with the 'State' section. A confirmation dialog box is overlaid on the page. The dialog has a yellow circular icon with an exclamation mark. It contains the text 'Are you sure you want to delete state?' and 'You won't be able to revert this!'. At the bottom are two buttons: 'No, cancel!' (gray) and 'Yes, delete it!' (blue). In the background, the 'State DataTable' table is visible with rows for Chhattisgarh, Odisha, and Punjab. Each row has an 'Edit' icon and a 'Delete' icon in the 'Action' column.

5. District –

- Add District

The screenshot shows a web application interface for managing districts. At the top, there's a navigation bar with links to YouTube, Maps, and other system modules. On the left, a sidebar shows 'Baby Immune System' and 'Admin'. The main content area has a title 'District' and a sub-section 'Add District Master'. It contains two input fields: 'State' (dropdown menu) and 'District' (text input). Below these are 'Submit' and 'Cancel' buttons. A table titled 'District DataTable' lists five districts: Raipur, Chandigarh, Durg, Mahasamund, and Patna, each associated with a state like Chhattisgarh or Bihar. The bottom of the screen shows a Windows taskbar with various icons and system status.

- Update District

This screenshot shows a modal dialog box titled 'Update District'. Inside, there are two dropdown menus: 'State Name' (set to Chhattisgarh) and 'District Name' (set to Raipur). At the bottom right of the dialog are 'Close' and 'Update' buttons. The background shows the same 'Add District Master' page as the previous screenshot, with the district table visible.

- Delete District

This screenshot shows a confirmation dialog box with a large exclamation mark icon. The text asks 'Are you sure you want to delete district?' and includes a note: 'You won't be able to revert this!'. At the bottom are 'No, cancel!' and 'Yes, delete it!' buttons. The background shows the 'Add District Master' page with a district table containing three rows of data.

6. City

- Add City

S.No.	State	District	City	Action
1	Chhattishgarh	Durg	Durg	<input checked="" type="checkbox"/> <input type="button" value="Delete"/>
2	Punjab	Chandigarh	Chandigarh	<input checked="" type="checkbox"/> <input type="button" value="Delete"/>
3	Chhattishgarh	Durg	Durg	<input checked="" type="checkbox"/> <input type="button" value="Delete"/>
4	Chhattishgarh	Rajpur	Sarona	<input checked="" type="checkbox"/> <input type="button" value="Delete"/>
5	Chhattishgarh	Rajpur	Mowa	<input checked="" type="checkbox"/> <input type="button" value="Delete"/>

- Update City

- Delete City

7. Hospital Registration

The screenshot shows a web-based application for hospital registration. On the left, there's a sidebar with a user profile picture, the text "Baby Immune System", and "Admin". Below that is a "Masters" section with a "Hospital Dashboard" button. The main content area has a title "Hospital Registration" and a subtitle "Add Hospital Registration Master". The form contains fields for "Hospital Name" (with a placeholder "Enter Hospital Name"), "State" (with a dropdown placeholder "--Select State--"), "District" (with a dropdown placeholder "--Select District--"), "City" (with a dropdown placeholder "--Select City--"), "Hospital Address" (with a placeholder "Enter Hospital Address"), "Contact Person" (with a placeholder "Enter Contact Person"), "Contact Number" (with a placeholder "Please Enter 10 Digit Mobile Number"), "Pincode" (with a placeholder "Enter Pincode"), "Hospital Email" (with a placeholder "Enter Hospital Email"), and two buttons at the bottom: "Submit" (blue) and "Cancel" (red).

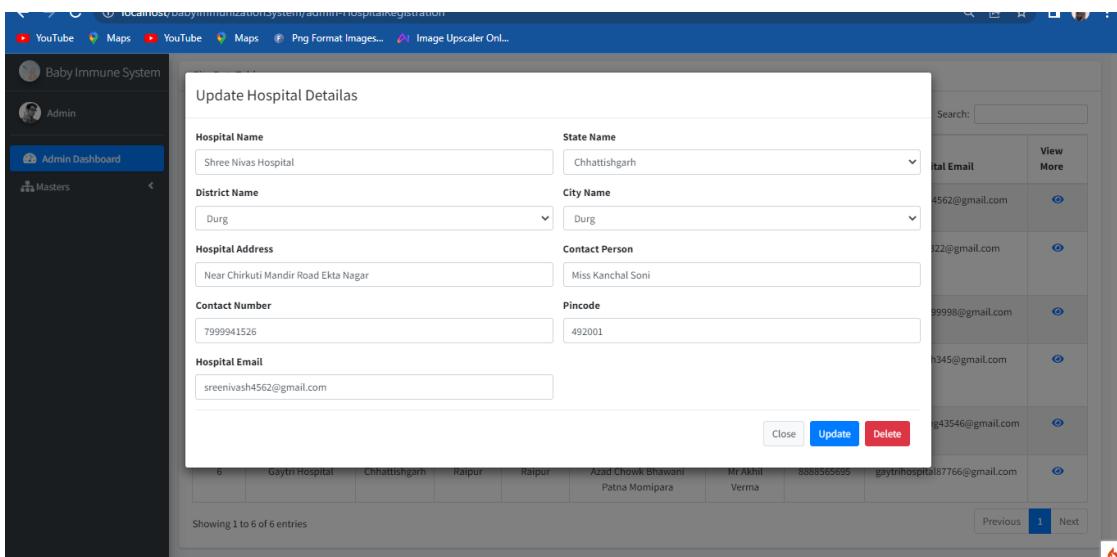
• Hospital Master

The screenshot shows a list of hospitals in a "City DataTable". The sidebar on the left is identical to the previous screenshot. The main content area has a title "City DataTable" and a table with the following data:

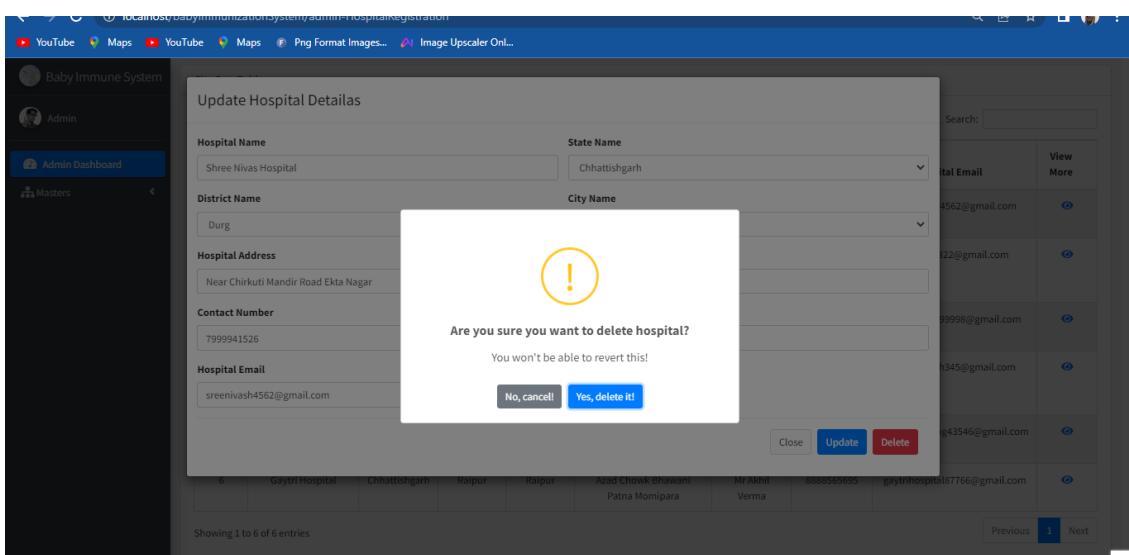
S.No.	Hospital Name	State	District	City	Hospital Address	Contact Person	Contact Number	Hospital Email	View More
1	Shree Nivas Hospital	Chhattisgarh	Durg	Durg	Near Chirkuti Mandir Road Ekta Nagar	Miss Kanchal Soni	7999941526	sreenivash4562@gmail.com	
2	Braj Women and Child Care Nursing Home	Bihar	Patna	Patna	Syamji Market Meena Bajar	Mrs Ayushi Chudhary	9899556688	brajorg4322@gmail.com	
3	District Civil City Hospital	Punjab	Chandigarh	Chandigarh	Piccadilly Rd City Center	Mr Sanjay Kumar	9998585858	dchospital99998@gmail.com	
4	MEDI Health Hospital	Chhattisgarh	Raipur	Raipur	Kukurbeda Amanaka Near pt Ravishankar Shukla University	Mr Rem Agrawal	8888885598	medihealth345@gmail.com	
5	Pandey Nursing home	Chhattisgarh	Raipur	Raipur	Sama Colony Main Road Sama Colony	Miss Shivani Saxena	9425514696	pandeynursing43546@gmail.com	
6	Gaytri Hospital	Chhattisgarh	Raipur	Raipur	Azad Chowk Bhawani Patna Mominpara	Mr Akhil Verma	8888565695	gaytrihospital87766@gmail.com	

At the bottom, it says "Showing 1 to 6 of 6 entries" and has navigation buttons for "Previous", "Next", and a page number "1". The status bar at the bottom right shows "27°C Partly cloudy", "11:55", "ENG", and the date "20-09-2022".

- **Update Hospital**

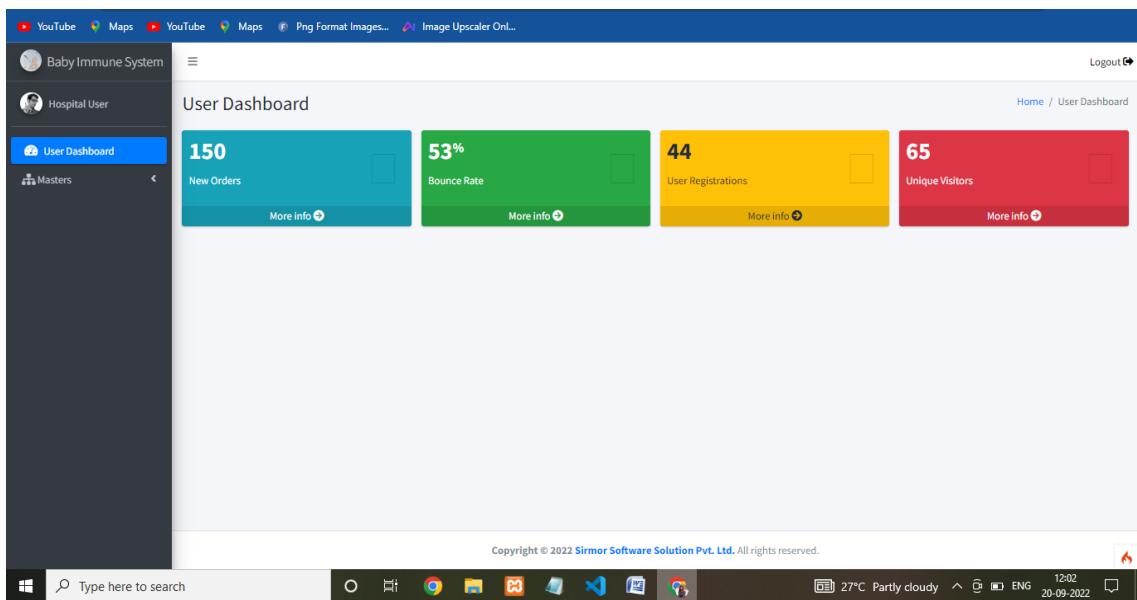


- **Delete Hospital**



User Module

8. User Dashboard



9. Department –

The screenshot shows a Windows desktop environment with a browser window open to the Baby Immune System application. The application's sidebar includes 'User Dashboard', 'Masters', and 'Logout'. The main content area is titled 'Department' and contains a 'Add Department Master' form with a 'Department Name' input field and 'Submit'/'Cancel' buttons. Below this is a 'Department DataTable' table with columns 'S.No.', 'Department', and 'Action'. The table lists five departments: General Surgery, Obstetrician, Schedule Injection, Consulting Department, and ENT Department, each with edit and delete icons in the 'Action' column. The system status bar at the bottom shows '27°C Partly cloudy', '12:04', 'ENG', and the date '20-09-2022'.

- **Update Department**

This screenshot shows the same application interface as above, but with a modal dialog box in the foreground titled 'Update Department'. It contains a 'Department Name' input field with 'General Surgery' typed in, a 'Close' button, and a prominent blue 'Update' button. The background 'Department DataTable' table is partially visible, showing the first row with '1' and 'General Surgery'.

10. Designation

The screenshot shows the 'Designation' master page in the Baby Immune System application. The top navigation bar includes links for YouTube, Maps, and various system functions. The left sidebar has icons for Baby Immune System, Hospital User, User Dashboard, and Masters. The main content area has a blue header 'Add Designation Master' and a sub-header 'Designation'. A search bar says 'Enter Designation'. Below is a table with columns S.No., Designation, and Action (checkboxes for edit and delete). The table contains five rows: 1. Recepnist, 2. Obstetrician gynecologist, 3. Physician Assistant, 4. Registered Nurse, and 5. Attending Doctor. At the bottom are 'Submit' and 'Cancel' buttons.

- **Update Designation**

The screenshot shows a modal dialog titled 'Update Designation' over the 'Designation' master page. The dialog has a 'Designation Name' input field containing 'Recepnist', a 'Close' button, and an 'Update' button. The background master page shows the same table as the previous screenshot, with the first row 'Recepnist' selected (indicated by a blue border).

11. Child Registration

The screenshot shows the 'Add Child Registration Master' form. It includes fields for Child Name, Father Name, Blood Group, Time Of Birth, Email, District, Pincode, Mother Name, Gender, Date Of Birth, Contact Number, State, City, and Address. The interface is a standard web-based application with a dark sidebar menu.

Child Name	Mother Name
Enter Child Name (Optional)	Enter Child's Mother Name
Father Name	Gender
Enter Child's Father Name	<input type="radio"/> Male <input type="radio"/> Female
Blood Group	Date Of Birth
--Select Blood Group--	Enter Date Of Birth
Time Of Birth	Contact Number
Enter Time Of Birth	Enter 10 digit Mobile Number
Email	State
Enter Email	--Select State--
District	City
--Select District--	--Select City--
Pincode	Address
Enter Pincode	Enter Home Address

- **Child Master**

The screenshot shows a data table titled 'Child DataTable' with columns: S.No., Child Name, Mother's Name, Father's Name, Gender, Blood Group, Date Of Birth, Time Of Birth, Admission Date, Consultant Doctor Name, and Action. The table contains 5 entries. The interface includes a sidebar menu for 'Masters' and a Windows taskbar at the bottom.

S.No.	Child Name	Mother's Name	Father's Name	Gender	Blood Group	Date Of Birth	Time Of Birth	Admission Date	Consultant Doctor Name	Action
1	Anu	Anjali	Samar	Male	AB+	17-Jun-2021	05:30 am	01-Jan-1970	Payal	
2	Ayushi Meshram	Santoshi Meshram	Shaman Meshram	Female	A-	01-Jan-1970	05:30 am	01-Jan-1970	Sandhya Pandey	
3	Ayush Narang	Ranu Naurang	Deepak Narang	Male	A-	09-Aug-2022	12:18 pm	01-Jan-1970	Arohi	
4		Renu Narang	Deepak Narang	Male	O+	09-Jun-2022	12:27 pm	09-Dec-2022	Shahil	
5	Sanu Kulkarni	Samita Kulkarni	Samrth Kulkarni	Female	AB-	01-Jan-1970	05:30 am	09-Oct-2022	Dr Nilam Deshmukh	

- **Update Child**

The screenshot shows a Windows desktop environment with a software application titled "Baby Immune System" running. The main window is titled "Update Child Detailas". It contains several input fields for child information:

- Child Name: Anu
- Mother Name: Anjali
- Father Name: Samar
- Gender: Male (selected)
- Blood Group: AB+
- Date Of Birth: 17-Jun-2021
- Time Of Birth: 05:30 am
- Contact Number: 8788787877
- Email: samae@gmail.com
- State Name: Chhattisgarh
- District Name: Raipur
- City Name: Raipur
- Pincode: 787878
- Address: Amapara
- Date Of Admission: 01-Jan-1970
- Referring Doctor Name: Priya

On the left side of the application, there is a sidebar with a navigation menu under "User Dashboard" and "Masters". The "Masters" section includes options like Department, Designation, Child Registration, Staff Master, Doctor Master, Vaccine Master, Vaccination Schedule, and Vaccinated.

On the right side, there is a sidebar titled "Consultant Doctor" with a table listing names and actions:

Consultant Doctor Name	Action
Payal	(Edit)
Sandhiya Pandey	(Edit)
Arohi	(Edit)
Shahil	(Edit)
Dr Nilam Deshmukh	(Edit)

The taskbar at the bottom of the screen shows various pinned icons and system status indicators.

12. Staff

The screenshot shows the 'Add Staff Master' form. It includes fields for Staff Name, Father Name, Gender (Male, Female, Other), Staff Designation, Date Of Joining, Contact Number, Email, State, District, City, Pincode, Local Address, Permanent Address, and Date Of Joining. The interface is a standard web-based form with dropdown menus and input fields.

● Staff Master

Staff DataTable									
S.No.	Staff Name	Gender	Designation	Date Of Birth	Contact Number	Email	Date Of Joining	Staff Department	Action
1	Kusum Shrivastava	Female	Nurse	22-09-2022	7896541236	kusum45@gmail.com	29-09-2022	General Surgery	
2	Suman Varma	Female	Obstetrician gynecologist	09-08-2022	4545454588	sukan245@gmail.com	01-01-1970	Consulting Department	
3	Sunita Yadu	Female	Receptionist	12-02-1998	7899456133	sunitayadu7665@gmail.com	12-12-2021	Obstetrician	

● Update Staff

The screenshot shows the 'Update staff Details' form. It includes fields for Staff Name, Father Name, Gender (Male, Female, Other), Staff Designation, Date Of Birth, Contact Number, Email, State Name, District Name, City Name, Pincode, Local Address, Permanent Address, Date Of Joining, Staff Image, and Staff Department. A sidebar on the right lists Staff Department and Action items. The interface is a standard web-based form with dropdown menus and input fields.

13. Doctor

The screenshot shows the 'Doctor Registration' page. It includes fields for Doctor Name, Gender (Male, Female, Other), Doctor Designation, Qualification, Date Of Birth, Contact Number, Email, State, District, City, Pincode, Permanent Address, Date Of Joinning, Doctor Image (with a placeholder 'IMAGE RESULT'), and Doctor Department.

• Doctor Master

The screenshot shows a 'Doctor DataTable' grid with columns: S.No., Doctor Name, Gender, Designation, Qualification, Date Of Birth, Contact Number, Email, Date Of Joinning, Doctor Department, and Action. There is one entry for Dr Sumita Datta.

S.No.	Doctor Name	Gender	Designation	Qualification	Date Of Birth	Contact Number	Email	Date Of Joinning	Doctor Department	Action
1	Dr Sumita Datta	Female	Obstetrician gynecologist	MD	01-01-1970	7878745821	dattasumita234@gmail.com	02-12-2020	Consulting Department	

• Update Doctor

The screenshot shows the 'Update Doctor Details' page. It includes fields for Doctor Name, Gender (Male, Female, Other), Doctor Designation, Qualification, Date Of Birth, Contact Number, Email, State Name, District Name, City Name, Pincode, Local Address, Permanent Address, Date Of Joinning, Doctor Image (with a placeholder 'IMAGE RESULT'), and Doctor Department.

14. Vaccine

The screenshot shows the 'Vaccine Registration' page. At the top, there are links to YouTube, Maps, and Image Upscaler Online. On the left, a sidebar includes 'BabyImmune System', 'Hospital User', 'User Dashboard', and 'Masters'. The main area has a blue header 'Add Vaccine Master'. It contains several input fields: 'Vaccine Name' (with placeholder 'Enter Vaccine Name'), 'Prevent' (with placeholder 'Vaccine Prevent'), 'Physical Form' (with placeholder 'Enter Physical Form(Optional)'), 'Composition' (with placeholder 'Enter Composition(Optional)'), 'Brand' (with placeholder 'Enter Brand(Optional)'), and 'Packaging Type' (with placeholder 'Enter PackagingType(Optional)'). Below these are 'Number of Doses' (with placeholder 'Enter Number of Dose of this Vaccine(Optional)') and two buttons: 'Submit' (blue) and 'Cancel' (red). The URL in the address bar is 'Home / Masters / Vaccine Registration'.

- **Vaccine Master**

The screenshot shows the 'Vaccine DataTable' page. The interface is similar to the previous one, with a sidebar and a blue header. The main content is a table titled 'Vaccine DataTable' with columns: S.No., Vaccine Name, Prevent, Dose, and Action. The data includes:

S.No.	Vaccine Name	Prevent	Dose	Action
1	BGC	TB and Bladder Cancer	1	<input checked="" type="checkbox"/> <input type="checkbox"/>
2	HepB	Hepatitis B	3	<input checked="" type="checkbox"/> <input type="checkbox"/>
3	Poliovirus	Polio	3	<input checked="" type="checkbox"/> <input type="checkbox"/>
4	DTP	Diphtheria Tetanus and Pertussis	5	<input checked="" type="checkbox"/> <input type="checkbox"/>
5	Hib	Infectious causes and Bacteria	4	<input checked="" type="checkbox"/> <input type="checkbox"/>
6	PCV	Pneumonia	4	<input checked="" type="checkbox"/> <input type="checkbox"/>
7	RV	Severe Diarrhoeal Disease	3	<input checked="" type="checkbox"/> <input type="checkbox"/>
8	Typhoid	Typhoid Fever Diarrhea	2	<input checked="" type="checkbox"/> <input type="checkbox"/>

At the bottom, it says 'Showing 1 to 8 of 8 entries' and 'Copyright © 2022 Sirmor Software Solution Pvt. Ltd. All rights reserved.' The taskbar at the bottom shows various icons and the system status: 27°C, Partly cloudy, 12:56, ENG, 20-09-2022.

- **Update Vaccine**

The screenshot shows the 'Update Vaccine Details' page. A modal dialog box is open over a background table. The dialog contains fields for 'Vaccine Name' (with values 'V00001' and 'BGC'), 'Prevent' (with value 'TB and Bladder Cancer'), 'Physical Form' (with placeholder 'Enter Physical Form(Optional)'), 'Composition' (with placeholder 'Enter Composition(Optional)'), 'Brand' (with placeholder 'Enter Brand(Optional)'), and 'Packaging Type' (with placeholder 'Enter PackagingType(Optional)'). Below these are 'Number of Doses' (with value '1') and two buttons: 'Close' (grey) and 'Update' (blue). The background table shows rows for 'BGC' (Dose 1, Prevent: TB and Bladder Cancer), 'Typhoid' (Dose 1, Prevent: Typhoid Fever Diarrhea), and 'Typhoid Fever Diarrhea' (Dose 2, Prevent: Typhoid Fever Diarrhea). The URL in the address bar is 'Home / Masters / Vaccine Registration'.

15. Vaccination Schedule

The screenshot shows a software application window titled "Vaccination Schedule". The left sidebar has icons for YouTube, Maps, Png Format Images..., and Image Upscaler Online. The main menu includes "Baby Immune System", "Hospital User", "User Dashboard", and "Masters". The "Masters" option is selected, showing a sub-menu with "Vaccination Schedule". The top right corner has a "Logout" link.

The "Add Vaccination Schedule" form is displayed. It includes fields for "Child Name" (CR00024), "Mother Name" (RENU NARANG), "Father Name" (DEEPAK NARANG), "Gender" (MALE), "Date Of Birth" (9/6/2022), and "Email" (ren232@gmail.com). A dropdown menu for "Add Vaccine" is open, showing "Select Vaccine--". Below this is a table with columns: S.No., Vaccine Name, Date Of Vaccination, and Action. Two rows are listed: row 1 (S.No. 1) has Vaccine Name HepB and Date Of Vaccination 09/14/2022; row 2 (S.No. 2) has Vaccine Name Poliovirus and Date Of Vaccination 11/09/2022. Each row has a small trash can icon in the Action column. At the bottom are "Submit" and "Cancel" buttons.

At the bottom of the screen, there is a Windows taskbar with the start button, a search bar, and various pinned icons. The system tray shows the date (20-09-2022), time (12:59), temperature (27°C), and weather (Partly cloudy).

16. Vaccinated

Vaccination

Child Name: RENU NARANG Mother Name: RENU NARANG Father Name: DEEPAK NARANG

Gender: MALE Date Of Birth: 9/6/2022 Email: ren232@gmail.com

VaccineSchedule DataTable

S.No.	Vaccine Name	Date Of Vaccination	Flag	Action
1	HepB	2022-09-22	5	[Action Icons]
2	Poliovirus	2022-09-29	5	[Action Icons]
3	DTP	2022-09-30	1	[Action Icons]

- Child Vaccinated

Insert Child Vaccinated Details

Vaccine Name : HEPB

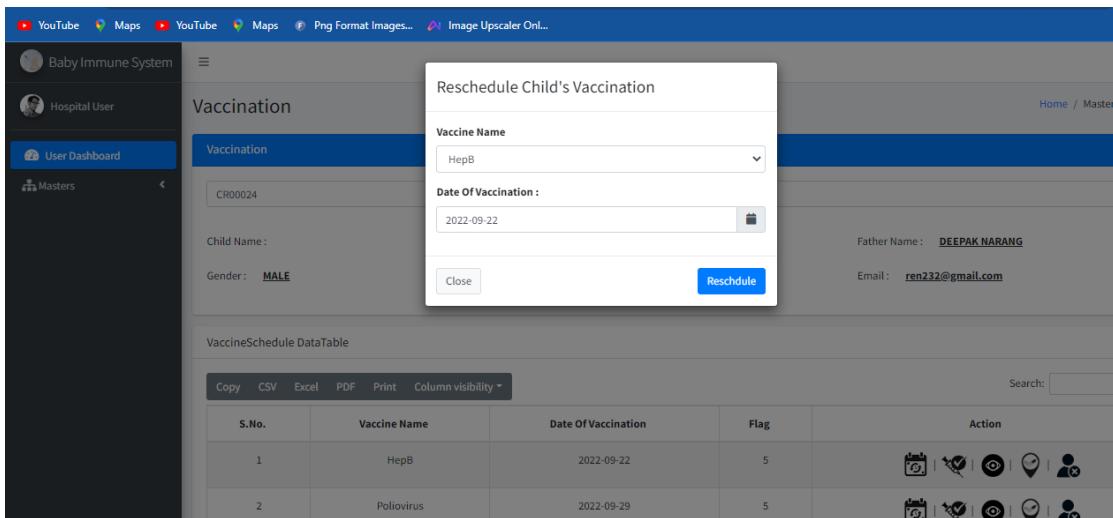
Date Of Vaccination : 2022-09-22

Doctor Name : Select Doctor

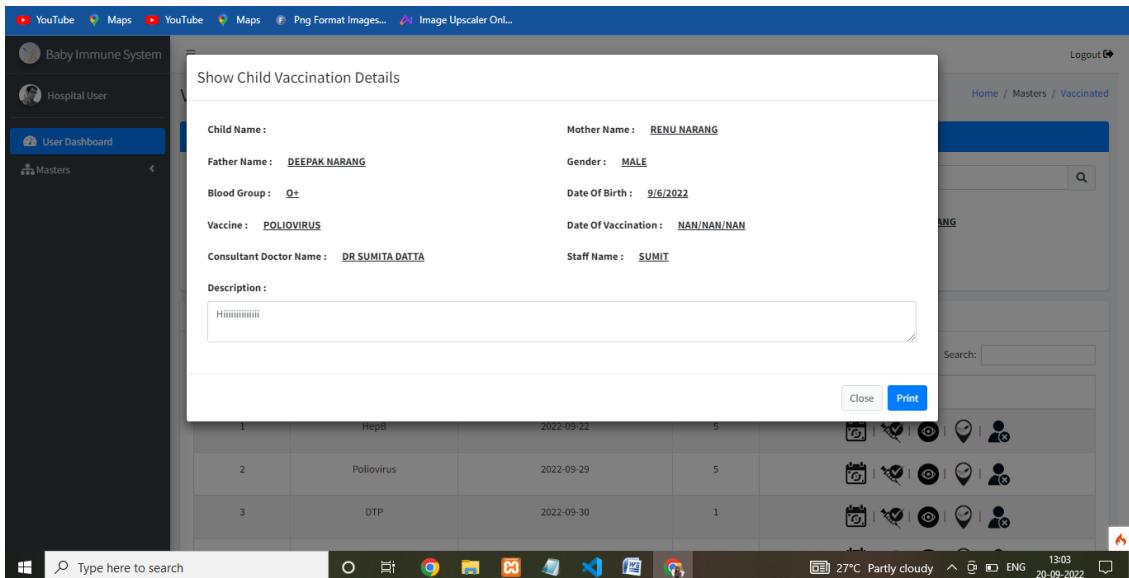
Staff Name : Select Staff

Description :

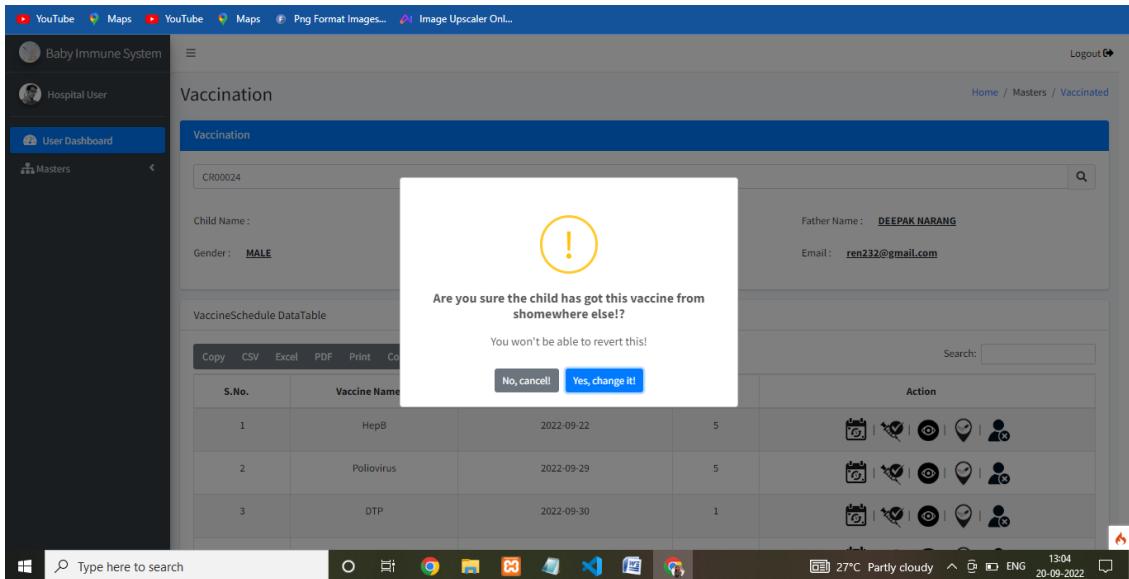
- **Reschedule**



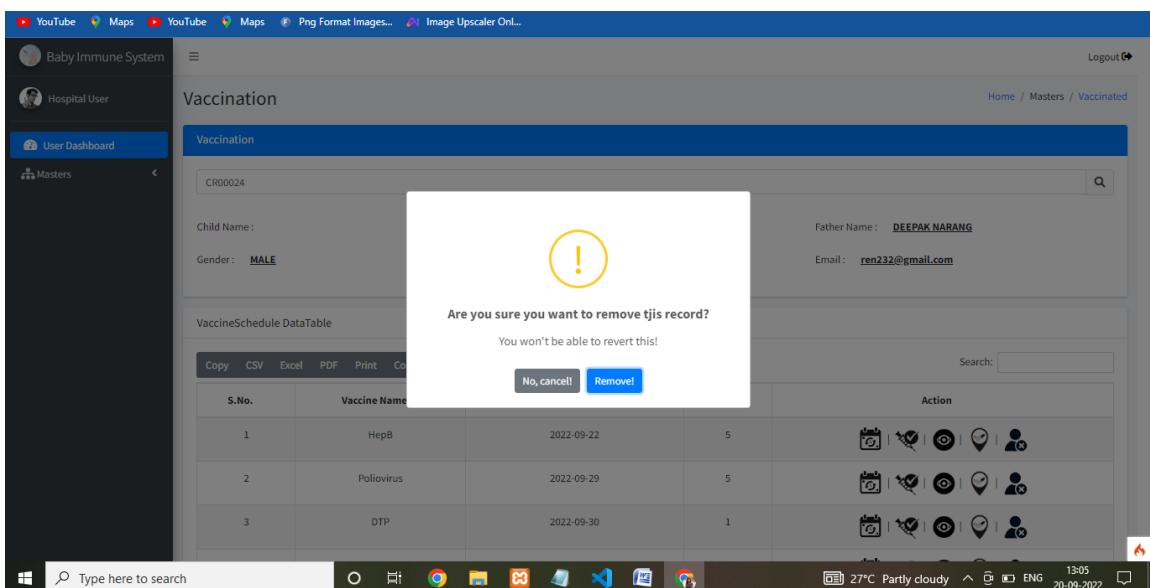
- **Child Vaccination Report**



- **Child Vaccinated from Other Hospital**



- **Remove Child If Child has died**



SYSTEM SECURITY

The protection of computer based resources that include hardware, software, data, procedures and people against unauthorized use or natural disaster is known as System Security.

System Security refers to the technical innovations and procedures applied to the hardware and operating systems to protect against deliberate or accidental damage from a defined threat.

Data Security is the protection of data from loss, disclosure, modification and destruction.

System Integrity refers to the proper functioning of hardware and programs, appropriate physical security, and safety against external threats such as eavesdropping and wiretapping.

Privacy defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

Confidentiality: It is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection.

In my project, Authentication in my project When a user creates his username and password then the password is stored in the database in encrypted form , which no one can see as a normal password and also when the id password is created by the user a token is also provided to him.

Security of Software System security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered and only valid operations are performed on the system.

CONCLUSION

The Fruits Delivery and Management System runs efficiently and Smoothly. we have developed secure, user-friendly Baby Immunization Management System. This System can take care of each member whether it is an Administrator or Customer .This System will help them to properly manage the orders of the customers, the delivery boy's data and help in growth without creating any hassle. This System is completely secure since every user is provided with user ID and Password so there is no chance of any unauthorised access . So, using this system will help in reducing the labour and provide more facility for Customer to like the services.

Finally it can be concluded that the overall “**BABY IMMUNIZATION MANAGEMENT SYSTEM**” - (USER) module of this Website or Web application (app) is implemented successfully.

- Now after the development of “**The MIS for Children Vaccination**”, maintaining data for Organization has become easier as compared to the earlier system.
- Fruit purchase has become easy now.
- The entry of data will become very easy with accuracy and without redundancy.
- User will save bill as pdf.

The overall management of “**The MIS for Children Vaccination**” is now easy to use and handle.

LIMITATION/FUTURE ENHANCEMENT

12.1 Limitation of system

- This is a web based application, so this need internet facility.
- Online payment method is not provided.
- Facility to order by direct clicking on any fruit is not provided.
- Add to cart facility is not provided
- This application needs high performance processor and RAM.
- Data processing can be slow if internet will be slow.

12.2 Future enhancement

It is not possible to develop a system that meets all the requirements of the user.

User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system are:

- **Online System.**
Most valuable thing in project is in online System.
- **Remote access.**
Database access from anywhere cloud services provide.
- **OTP/SMS Notification.**
New system provides new functionality.
- **Online Payment Method.**
Online payment can be done from anywhere with any method.

The above mentioned points are the enhancements which can be done to increase the ability and usage of this project.

BIBLIOGRAPHY

While developing this project, I have used some books and websites which helped me in our development process.

REFERENCE BOOKS

- Gud Apps Provided By SIRMOR

WEB REFERENCES

- <https://www.w3schools.com/angular>
- <https://ionicframework.com/docs>
- <https://nodejs.org/en/docs/>
- <https://www.tutorialspoint.com/typescript/index.htm>
- <https://www.tutorialspoint.com/ionic/index.htm>
- <https://github.com/>
- <https://www.tutorialspoint.com/angular>
- <https://stackoverflow.com>
- <https://www.bezkoder.com/>