# Online Medical Passport

## List Of Actions:

1)creating Pojos of Admin,Doctor and User.

2)Operation of insert ,update for Doctors

3)All Crud operations for Admin

4)Operation of select for User

## Purpose of project:

This document is meant to highlight the features of OMP, so as to serve as a guide to the developers

On one hand and a software validation document for the prospective client on the other The Online Medical Passport (OMP) web application is intended to provide complete medical history record for every single citizen using get-way through the internet. It will enable the government body to maintain a single database for keeping a health record of every individual online. The user can get a medical check-up through local medical authority in the city .

The administration module will take care of updating,deleting and creating new user record as well as adding new medical authority in a particular area.

## Scope of Project:

This Online Medical Passport system gives the information about medical history of single citizen using getway through internet .Doctors are allowed to use new Patient record or update the old patient record through health check up. Admin has all the authorities including deleting the patient record who has deceased.

## Definitions:

| | |
|---|---|
| OMP → | Online Medical Passport |
| SRS → | Software Requirement Specification |
| GUI→ | Graphical User Interface |
| Portal→ | Personalized Website |
| Stackholder→ | The person who will participate in the System. And Owner of system<br>Ex. Administrator , User ,Doctor |
| UML→ | Software Engineering Notation for visualising System in the form diagrams |
| SSL→ | Secure Socket Layer used for providing restricted access to application. |
| AM→ | Administration Management. |

RDBMS → Relational Database Management System.

## Additional Information:

The system works on internet server, so it will be operated by any end user to get history of his medical health .This system used is by the doctors to update the medical history by routine check up.

## General Description:

The Online Medical passport application helps to manage health record of every users and admins have authorities to update the medical history or delete the record from the database

## Functional Requirement:

This section provides requirement overview of the system. Various functional modules that can be implemented by the system will be-

Description:

User can get registered through Authorised Doctors Only .User cannot Register on site by its own

when he is registered on web portal the Doctors can fill the medical health information in user logged in .User can only show the medical history by logging in with his id and password. Doctor can update the user record data after every check up.

Users health check up report generation will be saved in data base, and after the check up user gets Successful health check up message on his Email-address.

The term client/server refers primarily to an architecture or logical division of responsibilities, the client is the application (also known as the front-end), and the server is the RDBMS (also known as the back-end).

A client/server system is a distributed system in which, Some sites are client sites and others are server sites.

All the data resides at the server sites.

All applications execute at the client sites.

Technical Issues:

This system will work on client-Server architecture. It will require an internet server.

The system should support some commonly used browser such as Chrome etc.

Interface Requirement Various interfaces for the product could be

1.Login Page,

There will be a screen displaying information about Medical information and gives users' medical history report after login. The Doctors may select the different options which will be open in another screen as

1.Login Page

2.Registration Form

3.Update Medical record.

4.history of the User.

5.Account Settings

Hardware Interface:

The System must run over the internet, all the hardware shall require to connect to internet will be hardware interface for the system.

e.g. modem, WAN, LAN

Specialized Server Infrastructure Hardware

The system should use distributed servers i.e. cloud for managing large amount of data so as to make it appear as single unit for end-user.

The system should have proper clusters for backup.

Software Interface:

The system is on server so it requires the any scripting language like JSP.

The system should be able to exchange data using XML, JASON or any advance technology. The system require Database also for storing any transaction of the system like MySQL .System also require DNS (Domain Name space) for the naming on the internet.

At the end user need web browser for interact with the system.

Performance Requirement:

There is no performance requirement in this system, because the server request and response to client is totally based on internet connection of end-user.

Design Constrains:

This system should be developed using Standard Web Page Development Tool , which conforms GUI standards such like HTML, XML, JSON,etc.

The system should support various RDMS and Cloud Technologies.

## Non-Functional Requirements:

1.Security:

SSL

The System use SSL (Secure Socket Layer) in all transactions that include any confidential customer information.

The system must automatically log out all customers after a period of inactivity. The system should not leave any cookies on the customer's computer containing users' password.

The system's back-end servers shall only be accessible to authenticated administrators. Sensitive data will be encrypted before being sent over insecure connections like internet.

The proper firewalls should be developed to avoid intrusions from the internal or external sources.

2.Reliability:

The system provides storage of all databases on redundant computers with automatic switchover.

The main pillar of reliability of the system is the backup of the database which is continuously maintained and update to reflect the most recent changes.

3: Availability:

The system should be available at all times. Meaning the user can access it using web browser,

only restricted by the down time of the server on which the system runs.

In case of a of a hardware failure or database corruption, a replacement page will be shown. uptime : It mean 24 * 7 availability

100%----------

99.9%

99.999%

99.9999%

4: Maintainability:

A commercial database is used for maintaining the database and application server takes care of the site.

The maintainability can be done efficiently.

5.Portability:

The application is HTML and scripting language based (JavaScript). So the end user part is fully portable and any system using any web browser should be able to use the features of the system, including any hardware platform that is available or will be available in the future.

An end-user is used this system on an OS. Either it is Windows or Linux.

The System shall run on PC, Laptops and PDA.etc.

The technology should be transferable to different environments easily.

6.Accessibility:

Only registered users should be allowed to process the orders after authentications.

Only GUI access of the system should be permitted to end users.

7.Policies:

The system should adhere to all the legal formalities of the particular countries.

The system should maintain security related to sensitive data.

8.Efficiency:

The system should provide good throughput and response to multiple users without burdening the system by using appropriate number of servers.

9.Safety:

Software should not harm ethical and environmental conditions of the end users machine.

10.Modulariy:

The system should have user friendly interface.

It should be easily updated, modified and reused.

Operational Scenario:

User Interaction

The User want to see his Medical health history after login and update the medical history by doctor after every medical check ups.

Doctor Interaction:

Medical Staff Interaction:

Admin interaction

**Preliminary Schedule:**

1.login.

2.Manage the users Medical history in database.

3.Add or update the description medical check up .

4. Manage User database.

5.Aprrove or reject the Doctors by admin.

6.Logout.

7.Visit Site

8.Create new account

9.View account details

10.Delete medical History and Delete User  Admin only.

11.Registration

12.Medical Team Support