

**The Model City Vaccination Drive Application**

**Software Requirement Specification (SRS) Document**

**Sprint 2 Implementation**

**Project Timeline: 7.9.2022 to 29.9.2022**

**Index**

|  |  |
| --- | --- |
| Title | Page No |
| 1. Introduction | 1 |
| * 1. Project Purpose | 1 |
| * 1. Project Scope | 1 |
| * 1. Intended Use | 1 |
| * 1. Project Scope | 1 |
| * 1. Glossary and reference | 1 |
| * 1. Overview | 2 |
| 1. Overall Descriptions | 2 |
| * 1. User Needs | 2 |
| * + 1. Beneficiary | 2 |
| * + 1. Government Personnel | 2 |
| * 1. Assumptions and Dependencies | 2 |
| 1. Detailed Features and Requirements | 3 |
| * 1. Functional Requirements | 3 |
| * 1. Non-Functional Requirements | 5 |
| * + 1. Usability | 5 |
| * + 1. Performance | 5 |
| * + 1. Security | 6 |
| * + 1. Supportability | 6 |
| * + 1. Design Constraints | 6 |
| * + 1. Reliability & Availability | 6 |
| * 1. External Interface Requirements | 7 |
| * + 1. Tools Used | 7 |
| * + 1. Software Interface | 7 |
| * + 1. Hardware Interface | 7 |

1. **Introduction**

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **Model City Vaccination Drive Application** by defining the problem statement in detail. The purpose of this document is that the requirements mentioned in it should be utilized by software developer to implement the system. This application will manage vaccine registrations and the data involved in the vaccination drive. The detailed requirements of the **Model City Vaccination Drive Application** are provided in this document.

* 1. **Project Purpose**

The purpose of this document is to describe the requirements to track all information related to vaccination drive**.** This online application would track all the new users, schedule vaccination slots for registered users and update all the data related to registered users.Secondly, the application also allows only the authorized city government employees to update the registered user’s database, track vaccine units in various centre and view specific required information related to vaccination drive.

* 1. **Intended Audience**

This document is intended to be read by, Client.

The intended audience for this application is the citizen that want to get vaccinated whereas the application is to be supervised by government personnel.

* 1. **Intended Use**
* Development Team
* Maintenance Team
* Clients
  1. **Project Scope**

The Model City Vaccination drive is specifically developed for the vaccination of the beneficiaries. This application allows the beneficiaries to register for the vaccination drive at any of the 3 centres available and also book their slots. Secondly, the application allows only authorised personnel to delete the fully vaccinated beneficiaries record. Also, he can create a vaccine file containing all the 3 centre codes and number of units of vaccine available across the centres just before the vaccination begins. He can also get the list of all the beneficiaries who got vaccinated today.

* 1. **Glossary and reference**
* <https://www.softwaretestinghelp.com/linked-list/>
* <http://www.trytoprogram.com/cplusplus-programming/multiple-inheritance/>
* <https://www.simplilearn.com/tutorials/cpp-tutorial/classes-in-cpp>
  1. **Overview**

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. General description of the project is discussed in section 2 of this document.

Section 3 gives the functional requirements, system features and constraints made while designing the system. Section 3 also discusses the external interface requirements and gives detailed description of functional requirements.

1. **Overall Descriptions**
   1. **User Needs**
      1. Beneficiary

* Aadhar Number (Unique 6-digit number).
* Name and Gender.
* Beneficiary age ( Should be >=18).
* Medical Certificate ( Only for user under medication & age > 35 ).
* Auto allotted registration ID.
  + 1. Government personnel
* Vaccine file database.
* Password.
* Centre codes.
* Registered application database.
  1. **Assumptions and Dependencies**

The assumptions are :

* Beneficiary has the latest version of Ubuntu Linux installed.
* Client has either an 4GB or more RAM.
* The service is used preferably on a desktop or laptop
* The information of all users, vaccine units and centres must be stored in a database.

The dependencies are:

* Beneficiarymust have their correct registration ID while the government personals must have correct password.
* The information of all the beneficiaries must be stored in a database that is accessible by the MVDA

1. **Detailed Features and Requirements**
   1. **Functional Requirements**
      1. **MVD\_01** è menu():

The user will be encountered with two options after starting the application: 1.Beneficiary Login 2. Admin Login. Beneficiaries are the ones who want to apply for vaccination drive whereas the admin login is for government personnel's who would supervise the vaccination drive.

* + 1. **MVD\_02** è user\_login():

There would be two choices for beneficiary login: 1.New 2.Registered. The registered beneficiaries can access the application by entering the allocated register id. Whereas the new beneficiaries have to enter the required details and is redirected to Registered choice.

* + 1. **MVD\_03** ènew\_user():

After choosing the new option in the previous login, user will be given two options: 1. Create Account 2.View Details.

* + 1. **MVD\_04** ècreate\_account():

Details are to be entered for creating a new account.  
Format of new user entries -  
Name, Aadhar number, Age, Gender.  
Example sample entries -  
123456 , Twinkle Dk , 22 , F  
568912 , Vaastav , 62 , M  
These details are first crosschecked with Aadhar File and only then they will be stored in Registered Applicant file.

* + 1. **MVD\_05** èview\_details():

Details of the created account are viewed in the format:  
Name : Nitika Nikam  
Aadhar no : 456789  
Gender : Female  
Age : 23.

* + 1. **MVD\_06** èregistered\_user():

After choosing the registered option, user will be given following options:  
1.Select Centre 2.Book Slot 3.View details.

* + 1. **MVD\_07** èselect\_centre():

Vaccination centres are selected by beneficiaries based on location and availability.

* + 1. **MVD\_08** èbook\_slot():

Slots are selected according to beneficiary's' convenience. Two choices for the specified date:

1. Morning (9:00 - 11:00) 2. Afternoon (13:00 - 16:00).
   * 1. **MVD\_09**èget\_vaccinated():

The beneficiary has to verify the token provided while filling the slot, only after this the vaccination is said to be successful. After completion of this stage, the count of the vaccination unit is decremented for that particular centre in the vaccination file.

* + 1. **MVD\_10** ècheckpassword():

A government personnel can login by entering the specified password so as to access all the functions related to the vaccination application. Only after the password is matched, the screen is directed to Admin menu.

* + 1. **MVD\_11** èadmin\_login():

User is given following options: 1. Display Beneficiary records 2. Add Vaccine Units 3. Display Vaccine Stock 4. Display Fully Vaccinated records 5. Delete Fully Vaccinated records 6. Display Non-Vaccinated Records 7. Display Today's vaccinated records. All these functions are accessed by Government personnels only.

* + 1. **MVD\_12** èshow\_bene\_data():

Data of all the beneficiary records are to be displayed in format:

1.234567 Anuja Jain Vcount = 2 F 26   
2. 567834 Vaidehii K Vcount =1 F 21.

* + 1. **MVD\_13** è add\_vaccine\_stock():

The user would be able to edit the of number of vaccine units allocated to each centre. These details are than updated in Vaccine file.

* + 1. **MVD\_14** èdisplay\_vaccine\_stock():

Displays the details of number of vaccine units in each allocated centres. These details are fetched from Vaccine file.

Example: centre name vaccine units

Hawkins 50

Central Perk 120

* + 1. **MVD\_15** èfully\_vaccinated():

Displays the list of beneficiary that have vaccine count equal to 3. These details are fetched from Registered Applicant file.

* + 1. **MVD\_16** èdelete\_fully\_vaccinated():

The user would be able to delete the record of the beneficiary that have reached the vaccine count to 3. These details are than updated in Registered Applicant file.

* + 1. **MVD\_17** ènon\_vaccinated():

Displays the list of beneficiaries that have vaccine count equal to 0. These details are fetched from Registered Applicant file.

* + 1. **MVD\_18** ètoday\_vaccinated():

Displays the list of beneficiary that have been vaccinated today. These details are fetched from Registered Applicant file.

* 1. **Non-Functional Requirements**
     1. **Usability**

The MVDA is designed for keeping essential digital records of beneficiary’s data. It executes the process of storing and tracking data of beneficiaries. The software would be able to handle basic information such as name, age, gender and number of vaccination dose.

* + 1. **Performance**

The proposed application that we are going to develop will be used as the Chief performance application within the different centres which interacts with the beneficiaries and the authorised government officials. Therefore, it is expected that the database would perform functionally according to all the requirements that are specified.

* The performance of the system should be fast and accurate.
* MVDA shall handle expected and non-expected errors in ways that would prevent loss in information and long downtime period. Thus, it should have inbuilt error testing to identify invalid Aadhar & password.
* The application should be able to handle large amount of data. Thus, it should accommodate high number of records of beneficiaries without any fault.
* The application will work on the user’s terminal. The performance shall depend upon hardware components of the beneficiaries/authorised personnel and the internet connection .
  + 1. **Security**
* Application will use a secured database.
* Beneficiary can just read information but they cannot edit or modify anything except their personal information.
* Application will have different types of beneficiary and specific beneficiary has specific access constraints.
* Proper user authentication is to be provided.
  + - 1. **Login**

The beneficiary can login the application to register for the vaccination drive at any of the three centres. After login the beneficiary can edit data and view his data. The authorised personnel can login with a fixed password to edit or delete vaccinated beneficiaries record and also add vaccine units for particular centre.

* + - 1. **Autogenerated unique PIN**

After the beneficiary book for vaccine then he/she acquire an autogenerated random pin and only then he/she can get vaccinated. Only after the pin is verified the details are updated.

* + 1. **Supportability**

The application is easy to maintain.

* + 1. **Design Constraints**

The application is built using only CPP language which puts certain limitation to the visual appeal of the software.

* + 1. **Reliability & Availability**

The application is available when the beneficiary and city government employee’s requests for service. The application is available for 24/7.

* 1. **External Interface Requirement**
     1. **Tools Used**
* CPP File Handling
* CPP Language
* Valgrind
* Splint
* Cppunit
* Gcov
* Linux on cloud
* Cppcheck
  + 1. **Software Interface**
* Any flavour of Linux/Unix operating system.
* Programming Language : C++ Language
* The final application must be packaged in a set up program, so that the application can be easily installed on machines. This application must be networked to corresponding vaccination centres.
  + 1. **Hardware Interface**

Since the application must run over the internet, all the hardware shall require to

Connect internet will be hardware interface for the system

Various interfaces for the product could be

* Touch screen/Monitor with 8 GB RAM
* Continuous battery backup