

Registration System For COVID-19 Vaccination

Project Report

Submitted in the partial fulfillment for the award of the degree of

**BACHELOR OF ENGINEERING
IN
INFORMATION SECURITY**

Submitted by:

Pranav Gupta

20BCS3703

Under the Supervision of:

Gurpreet Singh Panesar



**CHANDIGARH
UNIVERSITY**

Discover. Learn. Empower.

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
APEX INSTITUTE OF TECHNOLOGY**

**CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,
PUNJAB**

July - 2021

DECLARATION

I, **Pranav Gupta**, student of '**Bachelor of Engineering in Information Security-CSE**', session: **2020 -2024**, Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Punjab, hereby declare that the work presented in this Project Work entitled '**Registration System for Covid-19 Vaccination**' is the outcome of our own bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

(Pranav Gupta)

UID: 20BCS3703

Date: 25 July 2021

Place: Chandigarh

Acknowledgement

Place: Chandigarh

Date: 25 July 2021

On this great occasion of accomplishment of our project on '**Registration System for Covid-19 Vaccination**',

We would like to sincerely express our gratitude to **Mr. Gurpreet Singh Panesar**,
who has been supported through the completion of this project.

We would also be thankful to Assistant Professor **Mrs. Akwinder Kaur** of
Chandigarh University for providing all the required facilities in completion of this project.

Finally, as one of the team members, I would like to appreciate all my group members for their
support and coordination, I hope we will achieve more in our future endeavors.

Pranav Gupta

- Abstract of the Project Registration System for Covid-19 Vaccination:-

Registration System for Covid-19 Vaccination is an online service that helps managing the status of Vaccinated and Unvaccinated Users.

- Admin has all the controls and can see details of registered users.
- It manages the records of Users registered with the portal.
- Provides with password security using encryption.
- Keeps track of Vaccinated and Unvaccinated users.
- Data can be easily update, delete and create easily.
- It's secure as admin can login only by using password.
- Changes can be done as per requirements.

Table of Contents

Title Page	i
Declaration of the Student	ii
Abstract	iii
Acknowledgement	iv
List of Figures	v
List of Tables (optional)	vi
Timeline / Gantt Chart	vii
1. INTRODUCTION*	1
Problem Definition	1
Project Overview/Specifications	2
Specifications	3
	4
	4
2. PROJECT SURVEY	5
Existing System	5
Proposed System	6
Feasibility Study	7
3. PROBLEM FORMULATION	9
4. OBJECTIVES	
5. METHODOLOGY	16
6. CONCLUSIONS AND DISCUSSION	18
7. REFERENCES	19
	22

Introduction to Project

The aim of this Registration System for Covid vaccination project is to build a system that will completely automate the process of new user registration for vaccination. The system will handle the details submission, vaccination appointment.

In this system ,we can login it is further classified into two types- login as user and login as Admin. Admin credentials are already pre-defined in this system , only if he enters same details as pre-defined he will be allowed to login. He will be then re-directed to database.

It facilitates users to be register themselves for immunization by providing Interactive interface. Moreover, it stores all the pertinent details that can only be accessed by Admin. The system has two implementations i.e client side (user) and server side (Admin).

PROJECT SURVEY

Existing System:

The current registration system used by the Government has a huge amount of data, slow update speed, and longer data processing Operation, the security of the system is not that much efficient .

Due to the increasing amount of stored data, although the existing system can adapt to the current huge amount of data, the machine load is too large which adds mess to lot of files which adds complexity .

Proposed System:

The new system adopts advanced database technology, so that the accuracy and safety of data has been greatly improved and user interface has also been improved for better experience.

Project Feasibility:

Technical Feasibility:

- ✓ Project is based on C++ program which uses various concepts of Qt libraries.
- ✓ Program includes various functions to fulfil the user requirements.
- ✓ It is solving various problems which arise due to the existing system.

- ✓ Also it is efficient in terms of memory and time consumption.
- ✓ No high-level software or hardware requirements as such are required.
- ✓ So on the basis of the above points it can be concluded that Project is technically feasible.

Economic Feasibility:

As the existing government system requires huge expenditures for maintaining their portal so as to reduce it to certain limits ,proposed system came into existence i.e. less expenditure is spent on maintaining the data and many more. So the development of this project is highly feasible.

PROBLEM FORMULATION:

The new system adopts advanced database technology, so that the accuracy and safety of data has been greatly improved and user interface has also been improved for better experience.

OBJECTIVES:

Its purpose is to facilitate users to register themselves for immunization by providing interactive interface. Moreover, it stores all the pertinent details that can only be accessed by Admin. The system has two implementations i.e client side (user) and server side (Admin).

Source Code:

User's Registration

```
#include "secdialog.h"
#include "ui_secdialog.h"
#include "userlogin.h"

SecDialog::SecDialog(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::SecDialog)
{
    ui->setupUi(this);
    ui->adhaar->
>setPlaceholderText("Enter a valid Adhaar Card Number");
    ui->Age_3->setPlaceholderText("Enter your age");
    ui->Username_3->
>setPlaceholderText("Choose a unique Username");
    ui->Password->setPlaceholderText("Choose a strong Password");
    ui->label_2->setAlignment(Qt::AlignCenter);
    mydb = QSqlDatabase::addDatabase("QSQLITE");
    mydb.setDatabaseName("F:/SQL lite/register.db");
    if(!mydb.open()){
        ui->label_2->setText("Database not connected");
    }
    else{
        ui->label_2->setText("Database connected");
    }
}

SecDialog::~SecDialog()
{
    delete ui;
}

void SecDialog::on_pushButton_clicked()
{
    QSqlQuery qry,qry1;
```

```

QString adhar, age, username, date, password,status;
adhar = ui->adhaar->text();
age= ui->Age_3->text();
username = ui->Username_3->text();
date = ui->dateEdit->text();
password= ui->Password->text();

QByteArray encryp = password.toUtf8();
for (int j =0;j<encryp.size();j++ ) {
    encryp[j]= encryp[j]+4;
}
password = encryp;

status = "Unvaccinated";
if(!mydb.isOpen()){
    qDebug()<<"Database failed ";
}
if(qry1.exec("select * from register where username = '"+user
name+"'")){
    int count =0;
    while(qry1.next()){
        count++;
    }
    if(count==1){
        ui->label_2-
>setText("Deatils are required..! or Username already exists..!");
    }
    if(count<1){
        if(qry.exec("insert into register(adhar,age,username,d
ate,password,status)values('"+adhar+"','"+age+"','"+username+"','"
+date+"','"+password+"','"+status+"')")){
            ui->label_2-
>setText("Data entered you have been sucessfully registerd...!");
        }
    }
}
UserLogin user;
user.setModal(true);
user.exec();
}

```

User's login

```
#include "userlogin.h"
#include "ui_userlogin.h"
#include <QDesktopServices>
#include <QUrl>
UserLogin::UserLogin(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::UserLogin)
{
    ui->setupUi(this);
    ui->label_5->setAlignment(Qt::AlignCenter);
    mydb = QSqlDatabase::addDatabase("QSQLITE");
    mydb.setDatabaseName("F:/SQL lite/register.db");
    if(!mydb.open())
        ui->label_5->setText("Database failed connection");
    else
        ui->label_5->setText("Database connected...!");
}

UserLogin::~UserLogin()
{
    delete ui;
}

void UserLogin::on_pushButton_clicked()
{
    QString username , password;
    username = ui->lineEdit_2->text();
    password = ui->lineEdit_3->text();
    if(!mydb.isOpen()){
        qDebug()<<"Database failed...!";
        return;
    }
    QSqlQuery qry;
    if(qry.exec("select * from register where username = '"+username+"' and password = '"+password+"'")){
        int count =0;
        while(qry.next()){
            count++;
        }
        if(count==1){
```

```
        ui->label_5->setText("Log in sucessfully..!");
    }
    if(count<1){
        ui->label_5-
>setText("Deatails are Required..! or Username or password not cor
rect...!");
    }
}
QString link = "https://www.mygov.in/covid-19/";
QDesktopServices::openUrl(QUrl(link));
}
```

Admin's login

```
#include "admin_login.h"
#include "ui_admin_login.h"
#include <QDesktopServices>
#include <QUrl>
Admin_login::Admin_login(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::Admin_login)
{
    ui->setupUi(this);
    ui->label_2->setAlignment(Qt::AlignCenter);
}

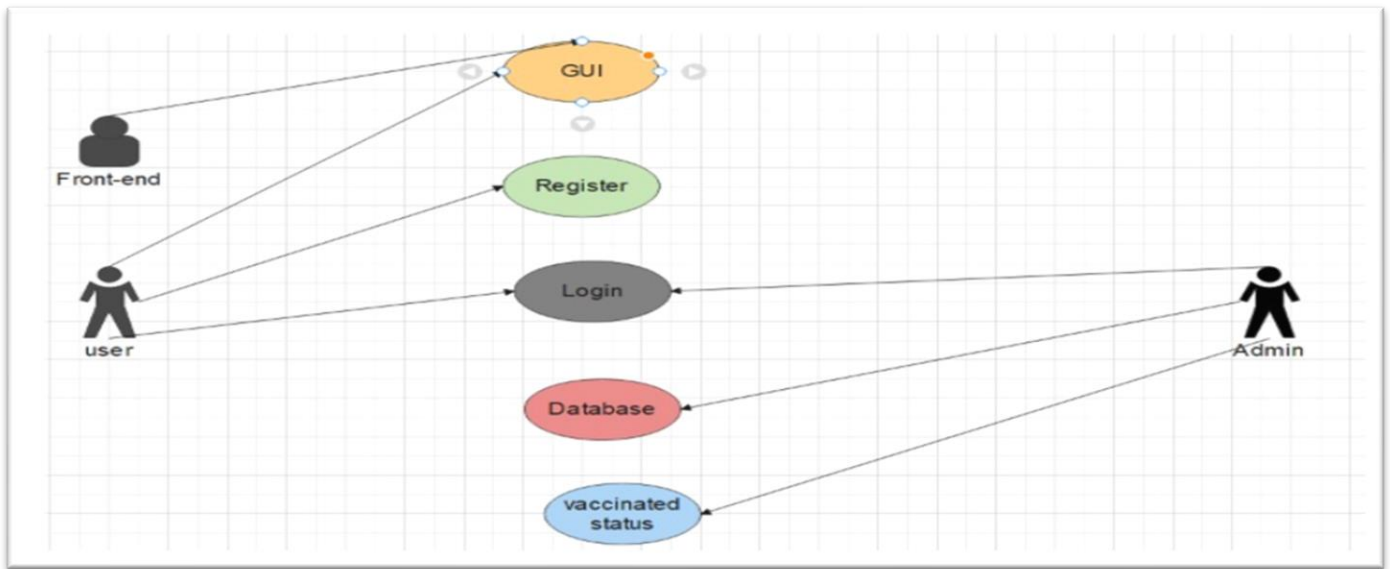
Admin_login::~Admin_login()
{
    delete ui;
}

QString Ausername = "admin", Apassword = "admin@123", Aadhar = "12
3456789";
void Admin_login::on_pushButton_clicked()
{
    QString username , password,aadhar;
    username = ui->USERNAME->text();
    password=ui->PASSWORD->text();
    aadhar = ui->ADHAAR->text();

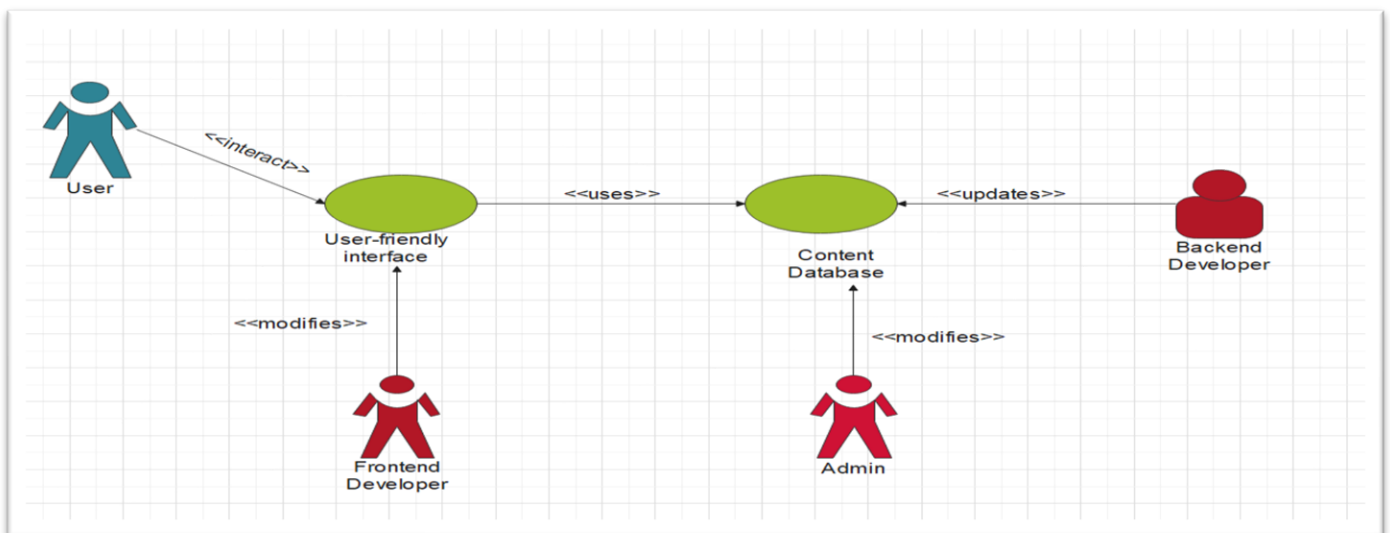
    if(username == Ausername && Apassword == password && aadhar== A
aadhar){
        ui->label_2->setText("Welcome admin...!");

        QString link = "https://filext.com/online-file-viewer.html";
        QDesktopServices::openUrl(QUrl(link));
    }
    else{
        ui->label_2->setText("No such data found...!");
    }
}
```

DATA FLOW DIAGRAM:



USE CASE DIAGRAM:



Conclusions and Discussion:

- Admin has all the controls and can see details of registered users.
- Helps managing the status of Vaccinated and Unvaccinated Users.
- It manages the records of Users registered with the portal.
- Provides with password security using encryption.
- Data can be easily update, delete and create easily.
- It's secure as admin can login only by using password.
- Changes can be done as per requirements.

Scope of project:

This system can be deployed using web based interface now it is working on Qt framework. Using Web Interface is definitely going to enhance user's experience. Moreover, another feature of downloading e-certificate can be integrated with this system and even the pitfall of this project can be resolved. This project is currently working on offline database that is working on a local machine only. Hence, for the future prospective we will use online compiler and load it on serves so as anyone can access it on their machines. This would further enhance the practicality of the product.

REFERENCES:

- www.google.com
- <https://doc.qt.io/qt.html>
- <http://www.codebind.com/cpp-tutorial/qt-tutorial/qt-tutorials-beginners-creating-simple-login-form-qt/>
- https://youtu.be/6_eY8O20I8
- <https://youtu.be/RL9nGmv3uSU>