VACCINATION APPLICATION

A Course Project Report in the course

Problem Solving with Programming

Computer Science & Engineering

By[TEAM: -7]

Roll.No:- 2103A52010	Name:- CH.RISHIKA.
Roll.No:- 2103A52011	Name:- CH.ALEKHYA.
Roll.No:- 2103A52018	Name:-G.SANJANA.
Roll.No:- 2103A52014	Name:-E.SAIKRISHNA.
Roll.No:- 2103A52017	Name:-G.MOHITH.
Roll.No:- 2103A52003	Name:-B.SHIVARAM

Under the Guidance of

Dr. J SRINIVAS

Assistant.Prof

Submitted to
July , 2022

PSP-PROJECT

TOPIC-VACCINATION APPICATION
SECTION-G1 (BATCH-7)

SR UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE

CERTIFICATE

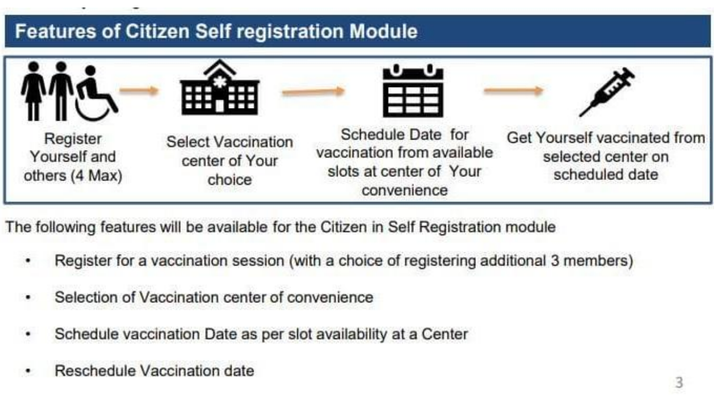
This is to certify that the Course Project Report entitled " VACCINATION APPLICATION" is a record of bonafide work carried out by the student(s) Ch.Rishika,Ch.Alekhya,E.Saikrishna,G.Mohith,G.Sanjana, B.Shivaram bearing Roll No(s) 2103A52010, 2103A52011, 2103A52014, 2103A52017, 2103A52018, 2103A52003 during the academic year 2021-22.

Supervisor **Head of the Department**

External Examiner

VACCINATION APPLICATION

Abstract: -



Features of Citizen Self registration Module

Register Yourself and others (4 Max) → Select Vaccination center of Your choice → Schedule Date for vaccination from available slots at center of Your convenience → Get Yourself vaccinated from selected center on scheduled date

The following features will be available for the Citizen in Self Registration module

- Register for a vaccination session (with a choice of registering additional 3 members)
- Selection of Vaccination center of convenience
- Schedule vaccination Date as per slot availability at a Center
- Reschedule Vaccination date

3

In this project we are making a registration program on vaccination. As the name suggests, the vaccine management system is software that handles all the data related to vaccination. The data contains the name of the persons who have taken the vaccine and also tells the current status of vaccine availability. Previously the task of handling the vaccination data was very difficult, so there was a need for software that can handle all the vaccination data.

This function will take necessary details of every candidate like their name, id card type & number, gender, etc. And it will store each data in a node like in a linked list. And we have to give details like state, city, hospital name where we want to take the vaccine.

After filling in the details it will generate all details in a receipt format.

Therefore the vaccine management system was designed. so that the stress and workload of employees were absolutely easier and it was also time-wasting for the employees to handle the whole vaccination data with the help of a notebook.

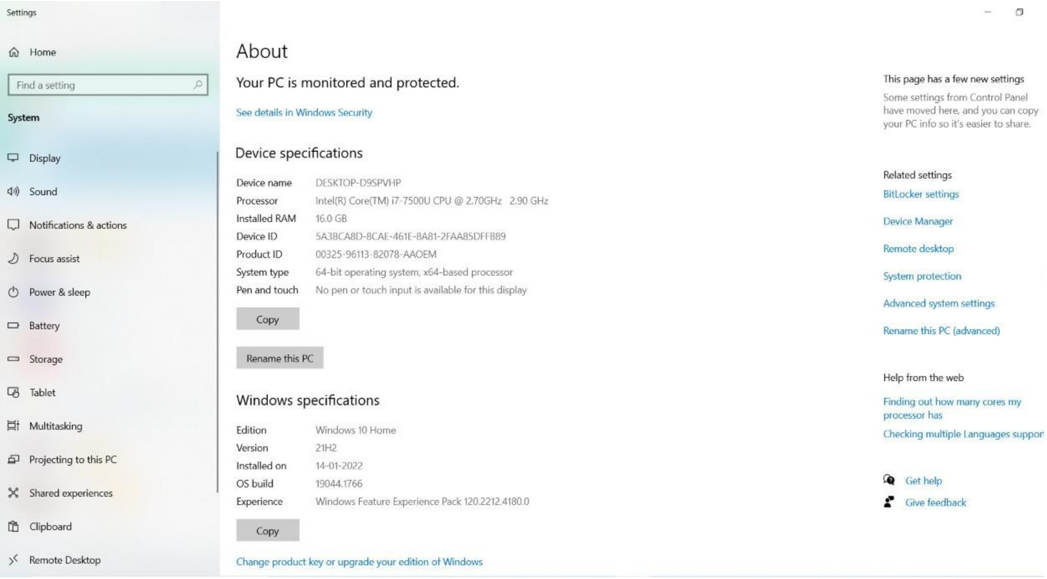
This is based on concept to maintain and generate vaccination registration.

This includes,

1. Name.
2. Gender.
3. Id type.
4. Id number.
5. Mobile number.
6. Co-Morbidity Status.

This code provides simplest and best way to register for vaccination.

PROJECT REQUIREMENTS:



About

Your PC is monitored and protected.

See details in Windows Security

Device specifications

Device name	DESKTOP-D9SPVHP
Processor	Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
Installed RAM	16.0 GB
Device ID	5A38CA8D-8CAE-461E-8A81-2FAA85DFFB89
Product ID	00325-96113-82078-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Windows specifications

Edition	Windows 10 Home
Version	21H2
Installed on	14-01-2022
OS build	19044.1766
Experience	Windows Feature Experience Pack 120.2212.4180.0

Change product key or upgrade your edition of Windows

This page has a few new settings

Some settings from Control Panel have moved here, and you can copy your PC info so it's easier to share.

Related settings

BitLocker settings

Device Manager

Remote desktop

System protection

Advanced system settings

Rename this PC (advanced)

Help from the web

Finding out how many cores my processor has

Checking multiple Languages suppor

Get help

Give feedback

HARDWARE:

SOFTWARE: DEV C++

Dev-C++ is a free full-featured integrated development environment distributed under the GNU General Public License for programming in C and C++. It is written in Delphi. It is bundled with, and uses, the MinGW or TDM-GCC 64bit port of the GCC as its compiler.

Written in: Object Pascal
Developer(s): Bloodshed Software until 2005, Orwell (Johan Mes) since 2011
License: GNU General Public License.
Stable release: 5.11 / April 27, 2015; 3 years ago.
Operating system: Microsoft Windows, Linux (alpha only)

Project Documentation:

<include>:
The #include directive tells the C preprocessor to include the contents of the file specified in the input stream to the compiler and then continue with the rest of the original file.

Stdio . h :
The header file stdio . h stands for Standard Input Output. It has the information related to input/output functions.

void main:
void main means return the type of main function is void which means it will return nothing. void means emptyness . void main means that the functions main () does not return any value.

header file:
A header file is a file with extension H which contains C function declarations and macro definitions to be shared between several source files. There are two types of header files: the files that the programmer writes and the files that comes with your compiler.

Strings:

Strings are defined as an array of characters. The difference between a character array and a string is the string is terminated with a special character '\0'. Declaration of strings: Declaring a string is as simple as declaring a one-dimensional array. Below is the basic syntax for declaring a string.

The file string. h is used in C programs to get access to a variety of functions for manipulating these arrays of characters.

The strcpy() function copies the string pointed by source (including the null character) to the destination. The strcpy() function also returns the copied string.

Typedef:

The typedef is a keyword used in C programming to provide some meaningful names to the already existing variable in the C program. It behaves similarly as we define the alias for the commands. In short, we can say that this keyword is used to redefine the name of an already existing variable.

Self-referential structures:

The self-referential structure is a structure that points to the same type of structure. It contains one or more pointers that ultimately point to the same structure. Structures are a user-defined data structure type in C and C++.

Self-Referential structures are those structures that have one or more pointers which point to the same type of structure, as their member. In other words, structures pointing to the same type of structures are self-referential in nature.

Global variables:

The variables that are declared outside the given function are known as global variables. These do not stay limited to a specific function- which means that one can use any given function to not only access but also modify the global variables.

Pointers:

The pointer in C language is a variable which stores the address of another variable. This variable can be of type int, char, array, function, or any other pointer. The size of the pointer depends on the architecture. However, in 32-bit architecture the size of a pointer is 2 bytes.

Function Calling:

A function call is an important part of the C programming language. It is called inside a program whenever it is required to call a function. It is only called by its name in the main() function of a program. We can pass the parameters to a function calling in the main() function.

Called function:

When a program calls a function, the program control is transferred to the called function. A called function performs a defined task and when its return statement is executed or when its function-ending closing brace is reached, it returns the program control back to the main program.

Structures :

Structures (also called structs) are a way to group several related variables into one place. Each variable in the structure is known as a member of the structure. Unlike an array, a structure can contain many different data types (int, float, char, etc.).

Char in data type:

The CHAR data type stores character data in a fixed-length field. Data can be a string of single-byte or multibyte letters, numbers, and other characters that are supported by the code set of your database locale.

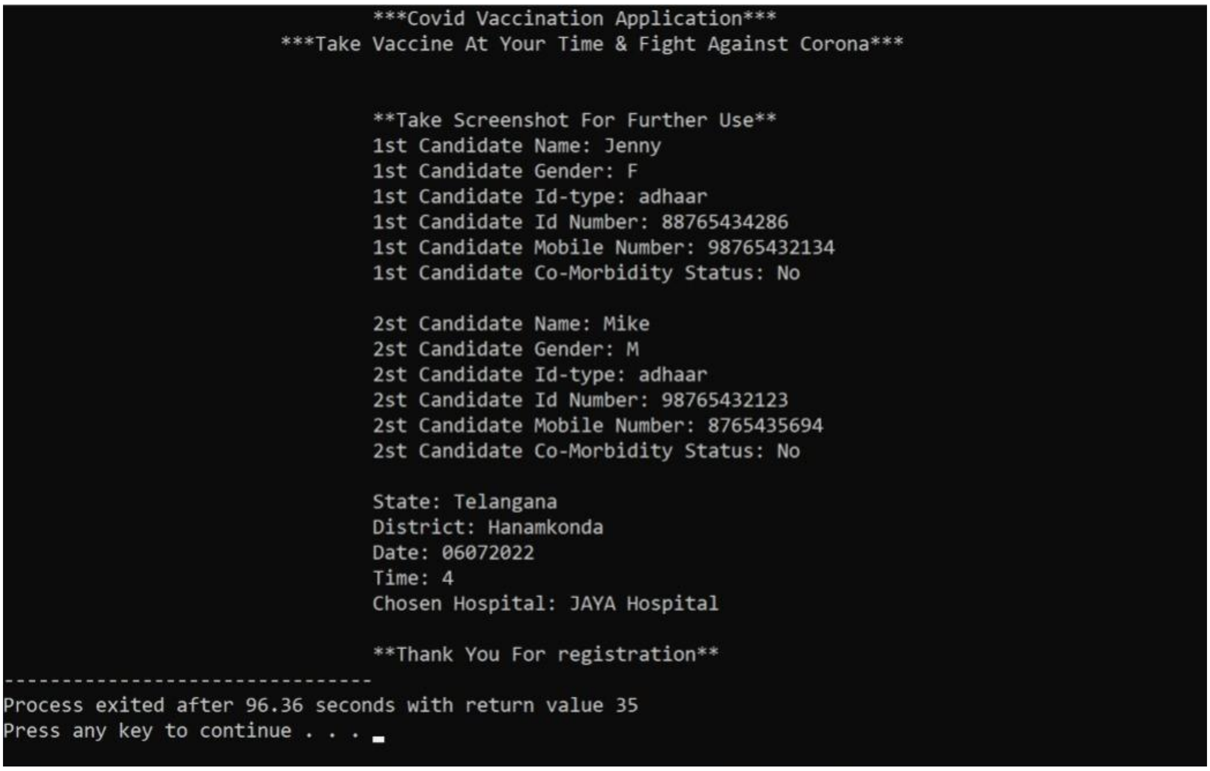
CODE:

```
 #include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Defining Structure
typedef struct mynode {
    char name[20];
    char gen[6];
    char idtype[40];
    char id[20];
    char mob[20];
    char comor[3];
    struct mynode* link;
} Node;
Node* start = NULL;

// Global Variables
int n;
char state[20], dis[20], hos[40], date[12], hour[6];

// Declaring Function Used In This Program
void heading();
void details();
void venue();
void receipt();

// Driver Code
void main()
{
    details();
    venue();
    receipt();
}

// Function To Take Candidate Numbers & Details
void details()
{
    int i;
    char a[20], b[6], c[40], d[20], e[20], f[3];
    heading();
    printf(
        "\t\t\t\tEnter Number of Candidate(s) applying for vaccination: ");
    scanf("%d", &n);
    for (i = 1; i <= n; i++) {
        system("cls");
        heading();
        printf("\t\t\t\tEnter The %dth Candidate Name: ",
               i);
        fflush(stdin);
        gets(a);
        printf("\t\t\t\tEnter The %dth Candidate Gender: ",
               i);
        fflush(stdin);
        gets(b);
        printf("\t\t\t\tEnter The %dth Candidate Id-Type: ",
               i);
        fflush(stdin);
        gets(c);
        printf(
            "\t\t\t\tEnter The %dth Candidate Id-Number: ",
               i);
        fflush(stdin);
        gets(d);
        printf("\t\t\t\tEnter The %dth Candidate Mobile "
               "Number: ",
              i);
        fflush(stdin);
        gets(e);
        printf("\t\t\t\tEnter The %dth Candidate "
               "Co-Morbidity Status (Yes or No): ",
               i);
        fflush(stdin);
        gets(f);

        // Calling Function addnode()
        addnode(a, b, c, d, e, f);
    }
}

// Function To Create Node & Insert Data Like Linked List
void addnode(char a[20], char b[6], char c[40], char d[20],
             char e[20], char f[3])
{
    Node *newptr = NULL, *ptr;
    newptr = (Node*)malloc(sizeof(Node));
    strcpy(newptr->name, a);
    strcpy(newptr->gen, b);
    strcpy(newptr->idtype, c);
    strcpy(newptr->id, d);
    strcpy(newptr->mob, e);
    strcpy(newptr->comor, f);
    newptr->link = NULL;
    if (start == NULL)
```

```
        start = newptr;
    else {
        ptr = start;
        while (ptr->link != NULL)
            ptr = ptr->link;
        ptr->link = newptr;
    }
}

// Function To Take Date & Time Details
void venue()
{
    int i, x = 0;

    // For Clear Screen
    system("cls");

    // Calling Heading() Function
    heading();
    printf("\t\t\t\tEnter State: ");
    gets(state);
    printf("\t\t\t\tEnter District: ");
    gets(dis);
    printf("\t\t\t\tEnter Date (DD-MM-YY): ");
    gets(date);
    printf("\t\t\t\tEnter Time (24 Hours): ");
    gets(hour);

    // For Clear Screen
    system("cls");

    // Calling Heading() Function
    heading();

    // List Of Hospitals Available
    printf("\t\t\t\t1. MGM Hospital\n");
    printf("\t\t\t\t2. JAYA Hospital\n");
    printf("\t\t\t\t3. ROHINI Hospital\n");

    // Taking Hospital Choice
    do {
        printf("\t\t\t\tEnter Choice: ");
        scanf("%d", &i);
        if (i == 1)
            strcpy(hos, "MGM Hospital");
        else if (i == 2)
            strcpy(hos, "JAYA Hospital");
        else if (i == 3)
            strcpy(hos, "ROHINI Hospital");
        else {
            printf("Enter Correct Choice...");
            x = 1;
        }
    } while (x);
}

// Function To Print Receipt
void receipt()
{
    int i;
    Node* ptr = start;

    // For Clear Screen
    system("cls");
    heading();
    printf(
        "\n\t\t\t\t*Take Screenshot For Further Use*\n");

    // Printing Candidate All Details
    for (i = 1; i <= n; i++) {
        printf("\t\t\t\t%dst Candidate Name: ", i);
        puts(ptr->name);
        printf("\t\t\t\t%dst Candidate Gender: ", i);
        puts(ptr->gen);
        printf("\t\t\t\t%dst Candidate Id-type: ", i);
        puts(ptr->idtype);
        printf("\t\t\t\t%dst Candidate Id Number: ", i);
        puts(ptr->id);
        printf("\t\t\t\t%dst Candidate Mobile Number: ", i);
        puts(ptr->mob);
        printf(
            "\t\t\t\t%dst Candidate Co-Morbidity Status: ",
            i);
        puts(ptr->comor);
        printf("\n");
        ptr = ptr->link;
    }
    printf("\t\t\t\tState: ");
    puts(state);
    printf("\t\t\t\tDistrict: ");
    puts(dis);
    printf("\t\t\t\tDate: ");
    puts(date);
    printf("\t\t\t\tTime: ");
    puts(hour);
    printf("\t\t\t\tChosen Hospital: ");
    puts(hos);
    printf("\n\t\t\t\t*Thank You For registration*");
}
```

```
// Function To Make Heading Of Portal
void heading()
{
   printf("\t\t\t\t**Covid Vaccination Application**\n");
   printf("\t\t\t***Take Vaccine At Your Time & Fight "
         "Against Corona***\n\n");
}
```

OUTPUT:-



```
                                ***Covid Vaccination Application***
                        ***Take Vaccine At Your Time & Fight Against Corona***


                                **Take Screenshot For Further Use**
                                1st Candidate Name: Jenny
                                1st Candidate Gender: F
                                1st Candidate Id-type: adhaar
                                1st Candidate Id Number: 88765434286
                                1st Candidate Mobile Number: 98765432134
                                1st Candidate Co-Morbidity Status: No

                                2st Candidate Name: Mike
                                2st Candidate Gender: M
                                2st Candidate Id-type: adhaar
                                2st Candidate Id Number: 98765432123
                                2st Candidate Mobile Number: 8765435694
                                2st Candidate Co-Morbidity Status: No

                                State: Telangana
                                District: Hanamkonda
                                Date: 06072022
                                Time: 4
                                Chosen Hospital: JAYA Hospital

                                **Thank You For registration**
--------------------------------
Process exited after 96.36 seconds with return value 35
Press any key to continue . . .
```

CONCLUSION:-

Vaccination application gives the easy access to all the details regarding name,gender,id type,id number,co-morbidity status.
Therefore the vaccine management system was designed. so that the stress and workload of employees were absolutely easier and it was also time-wasting for the employees to handle the whole vaccination data with the help of a notebook.