PSP [20ES104] COURSE PROJECT REPORT

On

"Online Voting System"

Developed By:

H.T.NO	STUDENT NAME
2203A54014	M. SUHAS
2203A51598	CH. THARUN
2203A54006	K. HARSHAVARDAN
2203A54012	M. ABHIRAM

Under the Guidance of

Mr. Kothakonda Chandhar, M.Tech

Senior Assistant Professor

Submitted to

Department Computer Science and Artificial Intelligence

SR University

Ananthasagar(V), Hasanparthy(M), Hanamkonda(Dist.) – 506371

www.sru.edu.in

June 2023

Department of Computer Science and Artificial Intelligence

CERTIFICATE

This is to certify that the PSP course project report entitled **"Online Voting System"** is a record of bonafide work carried out by the student(s) M.Suhas, CH.Tharun, K.Harshavardhan & M.Abhiram bearing roll number(s) 2203A54014, 2203A51598, 2203A54006 & 2203A54012 of Computer Science and Artificial Intelligence department during the academic year 2022-23.

Supervisor

(Kothakonda Chandhar)

INDEX

PROBLEM STATEMENT:

Develop a C application for an Online Voting System that allows voters to cast their votes and provides functionality for managing voter details and vote statistics.

The application should provide the following features:

1. Registration:
   Voters should be able to register by providing their name, surname, phone number, and password.
   Each voter should be assigned a unique voter ID.
2. Voter Login:
   Registered voters should be able to log in using their voter ID and password.
3. Casting Votes:
   After logging in, voters should be able to cast their votes by selecting a candidate from a list of candidates.
   The application should keep track of the number of votes received by each candidate.
   Invalid votes should also be counted separately.
4. Voting Statistics:
   The application should provide functionality to display the current vote counts for each candidate and the number of invalid votes.
5. Leading Candidate:
   The application should determine and display the leading candidate based on the vote counts.
6. Voter List:
   The application should allow the returning officer to view the list of registered voters.

7. Returning Officer Login:
   The returning officer should be able to log in using a password to access administrative functionalities.
8. Password Protection:
   The application should enforce password protection to ensure the security of voter information and prevent unauthorized access.
9. Data Persistence:
   The voter details and voting statistics should be stored in files to ensure data persistence across multiple sessions of the application.

The application should be implemented in C and use dynamically allocated memory to store voter details. It should provide a user-friendly interface for voters and the returning officer

MODULES:

1. File Handling:

   The code uses file handling operations to read and write voter information, vote counts, and statistics. You should be familiar with file operations in C, such as opening, reading, writing, and closing files.

2. String Manipulation:

   The code uses string functions from the `<string.h>` library to handle and manipulate voter information and passwords. Knowledge of string functions like `strcmp`, `strcpy`, and `strlen` would be necessary.

3. Standard Input and Output:

   The code uses functions like `printf`, `scanf`, and `getchar` for input and output operations. Understanding how to use these functions effectively is important.

4. Conditional Statements and Loops:

   The code includes conditional statements (`if-else`) and loops (`while`, `do-while`) to control the program flow and make decisions. You should have a good understanding of these control structures.

5. Structures:

   The code defines a structure (`struct vote`) to store voter information. Knowledge of structures in C and how to declare, define, and use them is necessary.

6. Random Number Generation:

   The code uses the `rand` function from the `<stdlib.h>` library to generate a random voter ID during registration. Understanding how to generate random numbers in C is important.

KNOWLEDGE REQUIRED TO DEVELOP THIS APPLICATION

- Control Statements (if, if-else, switch)
- Loop Statements (while/do while, for)
- Arrays (1D/2D-arrays)
- Strings (Strings and Table of strings) and its functions
- Functions (Any type of user defined functions)
- Structure (structures and nested structures)
- Files (stores data)

SOURCE CODE [.C FILE]:

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include <time.h>
#include <conio.h>
// Constants for candidate names, voter list, and vote statistics files
#define CANDIDATE1 "Vladimir Putin"
#define CANDIDATE2 "Kim Jong-un"
#define CANDIDATE3 "Narendra Modi"
#define CANDIDATE4 "Joe Biden"
#define CANDIDATE5 "NOTA"
#define VOTER_LIST "voters.txt"
#define VOTE_STATISTICS "votestatistics.txt"

// Variables to store vote counts and invalid votes
int votesCount1=0, votesCount2=0, votesCount3=0, votesCount4=0,
votesCount5=0;
int inValidvotes=0;
static int i=0;

struct vote          // Structure to store voter information
{
  int voter_id;
  char voter_name[50];
  char voter_surname[50];
  char voter_phone[15];
  char voter_password[30];
  int voter_check;
}v[50];

void interface1()  // Function to display the initial interface
{
```

```
printf("\n\t\t=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*");
   printf("\n\t\t=*=*=*=*=*     \033[1;36m ONLINE VOTING SYSTEM
\033[0m     =*=*=*=*=*");

printf("\n\t\t=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*");
   printf("\n");
}

void interface2()  // Function to display the welcome interface
{

printf("\n\t\t=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=");
   printf("\n\t\t=*=*=*=*=*          \033[1;36m WELCOME \033[0m
=*=*=*=*=*");
   printf("\n\t\t=*=*=*=*=*           \033[1;36m TO \033[0m
=*=*=*=*=*");
   printf("\n\t\t=*=*=*=*=*       \033[1;36m ONLINE ELECTION \033[0m
=*=*=*=*=*");

printf("\n\t\t=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*");
   printf("\n");
   printf("\033[0m\n\n");
   getch();
}

void check_required()    // Function to initialize voter check status
{
   int noo=0;
   for(noo=0;noo<50;noo++)
   {
      v[noo].voter_check=0;
   }
   getchar();
}
```

```
void castVote()     // Function to cast a vote
{
   system("cls");
   interface1();
   interface2();
   int choice;
   int var;
   int hi = 0, ji = 0;
   int hii = 0;
   char passwerd[30];
   printf("\n\t\t Enter your voter id : ");
   scanf("%d", &var);

   FILE* f = fopen(VOTER_LIST, "r");
   if (f == NULL)
   {
      printf("\n\t\t Error: Could not open file for reading");
      return;
   }
   // Read voter information from file
   while (fscanf(f, "%d,%[^,],%[^,],%d,%s\n", &v[hi].voter_id,
v[hi].voter_name, v[hi].voter_surname, &v[hi].voter_phone,
v[hi].voter_password) != EOF)
         {
      if (var == v[hi].voter_id)
      {
         fclose(f);
         do {
            printf("\n\t\t Enter your password: ");
            scanf("%s", passwerd);
            if (strcmp(v[hi].voter_password, passwerd) == 0)
                                 {
               if (v[hi].voter_check == 0)
                                       {
```

```
            printf("\n\t\t Voter name : %s", v[hi].voter_name);
            printf("\n\t\t ### Please choose your Candidate ####");    //
Display the candidates for voting
            printf("\n\t\t 1. %s", CANDIDATE1);
            printf("\n\t\t 2. %s", CANDIDATE2);
            printf("\n\t\t 3. %s", CANDIDATE3);
            printf("\n\t\t 4. %s", CANDIDATE4);
            printf("\n\t\t 5. %s", CANDIDATE5);
            printf("\n\n Input your choice (1 - 5) : ");
            scanf("%d", &choice);

            switch (choice)
            {
               case 1: votesCount1++; break;
               case 2: votesCount2++; break;
               case 3: votesCount3++; break;
               case 4: votesCount4++; break;
               case 5: votesCount5++; break;
               default: inValidvotes++;
            }
            v[hi].voter_check++;
            saveVotingStatistics(); // Update voting statistics
            printf("\033[32m");
            printf("\n\t\tThank you for voting");
            printf("\033[0m\n\n");
         }
         else       // If the voter ID is not found or the vote has already been
cast
         {
            printf("\033[1;31m");
            printf("\n\t\t Sorry you have already voted!!!");
            printf("\033[0m\n\n");
         }
         hii = 0;
```

```
            }
            else {
               printf("\033[1;31m");
               printf("\n\t\tYou have entered the password incorrectly!!!");
               printf("\n\t\t Please re-enter the password correctly.....");
               printf("\033[0m\n\n");
               hii++;
            }
         }while (hii != 0 && hii <= 3);
         break;
         system("cls");
      }
      hi++;
   }

   if (feof(f))
   {
      printf("\033[1;31m");
      printf("\n\t\t Sorry the voter id does not exist!!!");
      printf("\033[0m\n\n");
   }
   fclose(f);
   printf("\n\n Press any key to go back to the main menu.....");
   getch();
}

void votesCount()// Function to display individual votes
{
        system("cls");
        interface1();
        interface2();
        printf("\n\t\t ##### Voting Statics ####");
        printf("\n\t\t %s - %d ", CANDIDATE1, votesCount1);
```

```
        printf("\n\t\t %s - %d ", CANDIDATE2, votesCount2);
        printf("\n\t\t %s - %d ", CANDIDATE3, votesCount3);
        printf("\n\t\t %s - %d ", CANDIDATE4, votesCount4);
        printf("\n\t\t %s - %d ", CANDIDATE5, votesCount5);
        printf("\n\t\t Invalid votes - %d ", inValidvotes);
        saveVotingStatistics();   // Update voting statistics
        fflush(stdin);
        getchar();
}

void getLeadingCandidate()    // Function to determine & display leading
candidate
{
   system("cls");
   interface1();
   interface2();
   printf("\n\n  #### Leading Candiate ####\n\n");
   if(votesCount1>votesCount2 && votesCount1>votesCount3 &&
votesCount1 >votesCount4)
   printf("[%s]",CANDIDATE1);
   else if (votesCount2>votesCount3 && votesCount2>votesCount4 &&
votesCount2 >votesCount1)
   printf("[%s]",CANDIDATE2);
   else if(votesCount3>votesCount4 && votesCount3>votesCount2 &&
votesCount3 >votesCount1)
   printf("[%s]",CANDIDATE3);
   else if(votesCount4>votesCount1 && votesCount4>votesCount2 &&
votesCount4 >votesCount3)
   printf("[%s]",CANDIDATE4);
   else
   printf("----- Warning !!! No-win situation----");
   fflush(stdin);
   getchar();
}
```

```
void saveVotingStatistics()              // Function saves the vote counts to a
file.
{
   FILE* file = fopen(VOTE_STATISTICS, "w");
   if (file == NULL)
   {
      printf("Error: Could not open file for writing\n");
      return;
   }
   fprintf(file, "%d\n", votesCount1);
   fprintf(file, "%d\n", votesCount2);
   fprintf(file, "%d\n", votesCount3);
   fprintf(file, "%d\n", votesCount4);
   fprintf(file, "%d\n", votesCount5);
   fprintf(file, "%d\n", inValidvotes);

   fclose(file);
}

void loadVotingStatistics()              // Function to load the vote counts from
a file.
{
   FILE* file = fopen(VOTE_STATISTICS, "r");
   if (file == NULL) {
      votesCount1 = 0;
      votesCount2 = 0;
      votesCount3 = 0;
      votesCount4 = 0;
      votesCount5 = 0;
      inValidvotes = 0;
      return;
   }

   fscanf(file, "%d", &votesCount1);
```

```
    fscanf(file, "%d", &votesCount2);
    fscanf(file, "%d", &votesCount3);
    fscanf(file, "%d", &votesCount4);
    fscanf(file, "%d", &votesCount5);
    fscanf(file, "%d", &inValidvotes);

    fclose(file);
}

void voterList()    // Function to displays the list of voters
{
    system("cls");
    interface1();
    interface2();
    printf("\n\n  #### Voter List ####\n\n");

    FILE* f = fopen(VOTER_LIST, "r");
    if (f == NULL)
    {
        printf("Error: Could not open file for reading\n");
        return;
    }
    printf("Voter ID\tName\t\tSurname\t\tPhone\n");
    printf("-------------------------------------------------\n");

    while (fscanf(f, "%d,%[^,],%[^,],%[^,],%s\n", &v[i].voter_id,
v[i].voter_name, v[i].voter_surname, v[i].voter_phone, v[i].voter_password)
== 5)
    {
        printf("%d\t\t%s\t\t%s\t\t%s\n", v[i].voter_id, v[i].voter_name,
v[i].voter_surname, v[i].voter_phone);
    }
    fclose(f);
    fflush(stdin);
```

```
   getchar();
}

void returningofficer()            //  Login function for the returning officer
{
   system("cls");
   interface1();
   interface2();
   char returning_password[]="password";
   char enter_password[30];
   printf("\n\t\tEnter the password: ");
   scanf("%s",enter_password);
    if(strcmp(returning_password,enter_password)==0)
    {
     int choice_officer=0;
     do{
      printf("\n\t\t 1. Find Vote Count");
      printf("\n\t\t 2. Find leading Candidate");
      printf("\n\t\t 3. Show voter's list");
                  printf("\n\t\t 0. Exit");
      printf("\n\t\t Please enter your choice : ");
      scanf("%d", &choice_officer);

      switch(choice_officer)
      {
      case 1: votesCount();break;
      case 2: getLeadingCandidate();break;
      case 3: voterList();break;
      default: printf("\n Please wait,you are being directed to home page");
      }
    }while(choice_officer!=0);
    }
    else
```

```
 {
      printf("\033[1;31m");
                printf("\n\t\t Please renter the password correctly");
      printf("\033[0m\n\n");
    }
fflush(stdin);
getchar();
}

void voterlogin()          // Login function for voters
{
   system("cls");
   interface1();
   interface2();
   int choice_voter;
   do
   {
      printf("\n\t\t 1.Cast your vote");
      printf("\n\t\t 0.Exit");
      printf("\n\t\t Please enter your choice:");
      scanf("%d",&choice_voter);

      switch(choice_voter)
      {
         case 1: castVote();break;
         default: printf("\n Please wait you are being directed to home page");
      }
   }
   while(choice_voter!=0);
   fflush(stdin);
   sleep(100);
}
```

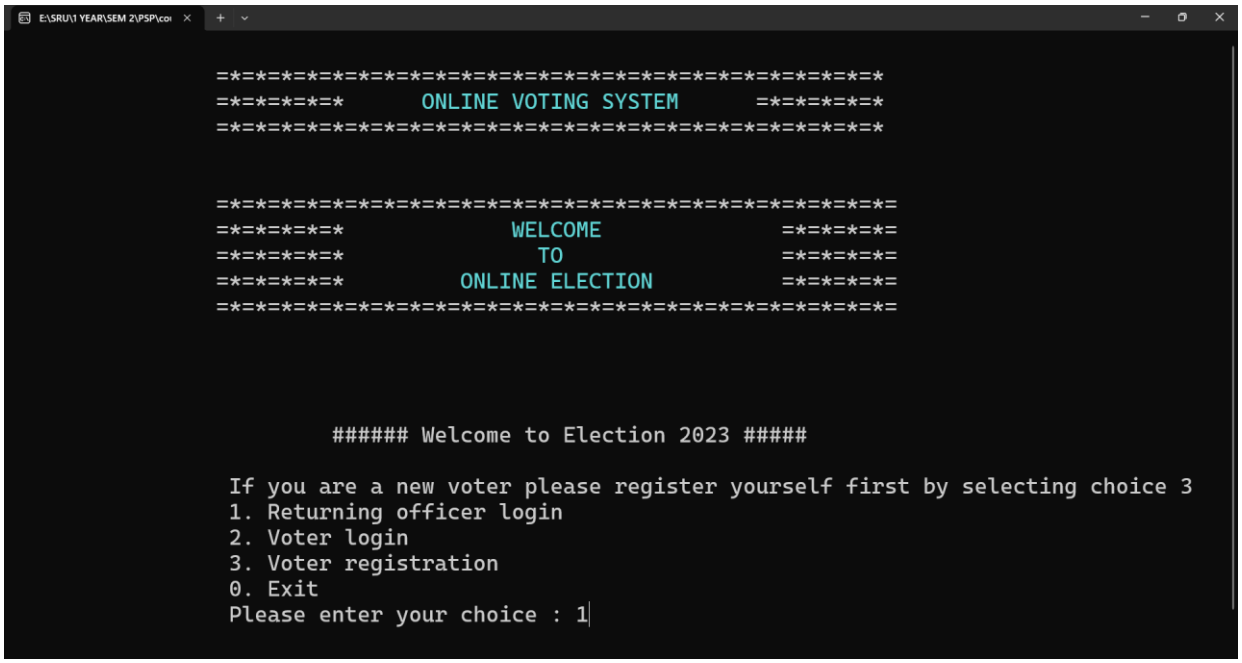
```
void voterregistration()          // Function for voter registration
{
   system("cls");
   interface1();
   interface2();
   srand(time(0));
   int id = rand();  // generates a random voter ID
   printf("Random number: %d\n", id);
   printf("\n\t\t Your voter id is %d", id);
   printf("\n\t\t Please enter the voter id allotted to you: ");
   scanf("%d", &v[i].voter_id);
   printf("\n\t\t Enter your name: ");
   scanf(" %s", v[i].voter_name);
   printf("\n\t\t Enter your surname: ");
   scanf(" %s", v[i].voter_surname);
   printf("\n\t\t Enter your phone number (10 digits): ");
   scanf(" %10s", v[i].voter_phone);
   printf("\n\t\t Enter your password: ");
   scanf(" %s", v[i].voter_password);
   printf("\033[32m");
   printf("\n\t\t You have successfully completed the registration!!!");
   printf("\033[0m\n\n");
   i++;
   id++;
   fflush(stdin);
   sleep(100);

   FILE* f = fopen(VOTER_LIST, "a");
   if (f == NULL)
   {
      printf("Error: Could not open file for writing\n");
      return;
   }
```

```
    fprintf(f, "%d,%s,%s,%s,%s\n", v[i - 1].voter_id, v[i - 1].voter_name, v[i - 1].voter_surname, v[i - 1].voter_phone, v[i - 1].voter_password);
    fclose(f);
}

int main()
{
    loadVotingStatistics();
    int choice;

    do {
        system("cls");
        interface1();
        interface2();
        printf("\n\t\t\t ###### Welcome to Election 2023 #####");
        Sleep(150);
        printf("\n\n\t\t If you are a new voter please register yourself first by selecting choice 3");
        Sleep(150);
        printf("\n\t\t 1. Returning officer login ");
        Sleep(150);
        printf("\n\t\t 2. Voter login ");
        Sleep(150);
        printf("\n\t\t 3. Voter registration");
        Sleep(150);
        printf("\n\t\t 0. Exit");
        Sleep(150);
        printf("\n\t\t Please enter your choice : ");
        scanf("%d", &choice);

        switch(choice)
        {
            case 1:
                vigilanceofficer();
```

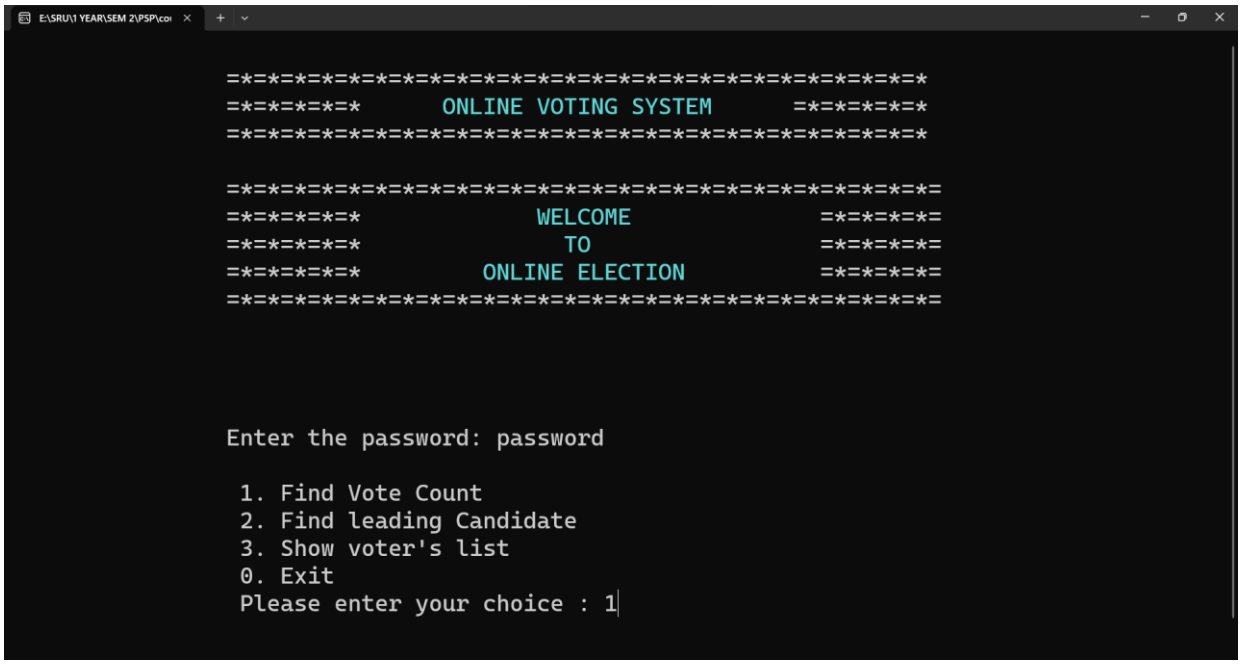
```
            break;
        case 2:
            voterlogin();
            break;
        case 3:
            voterregistration();
            break;
        default:
            Sleep(150);
            printf("\033[34m");
            printf("\n Thank you ");
            printf("\033[0m\n\n");
      }
   } while (choice != 0);
   fflush(stdin);
   sleep(100);
   return 0;
}
```

RESULTS:

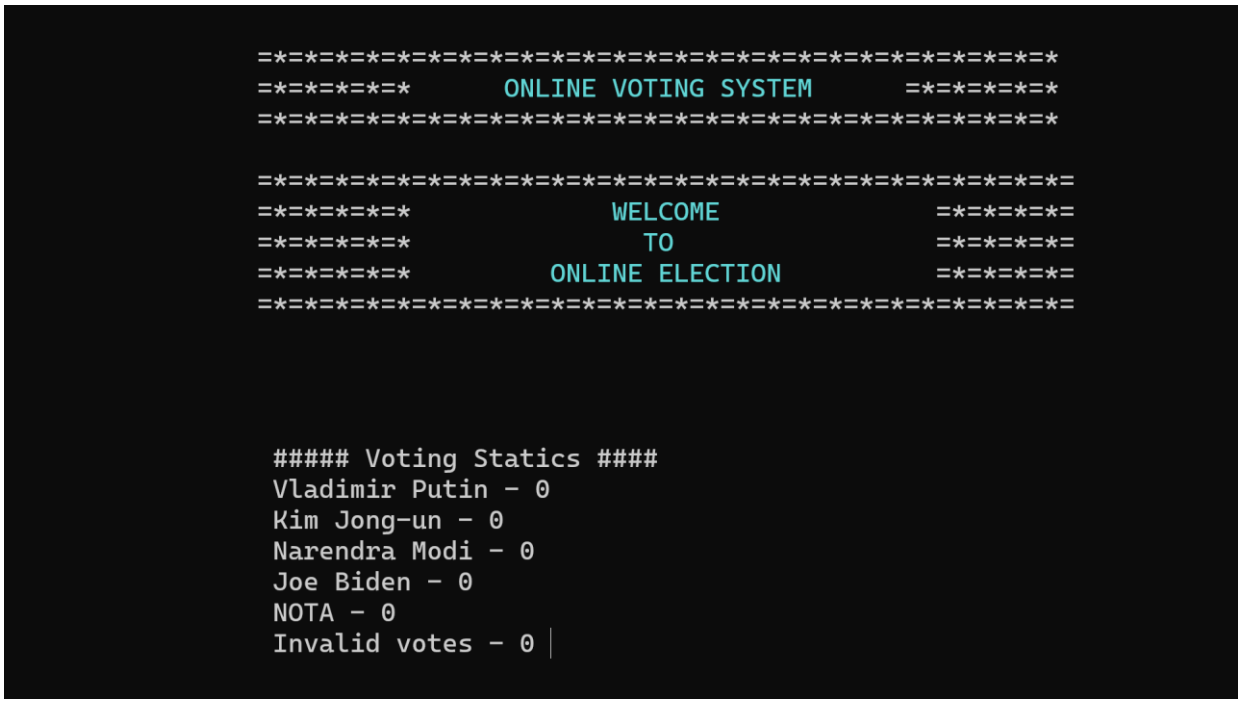
Home screen interface



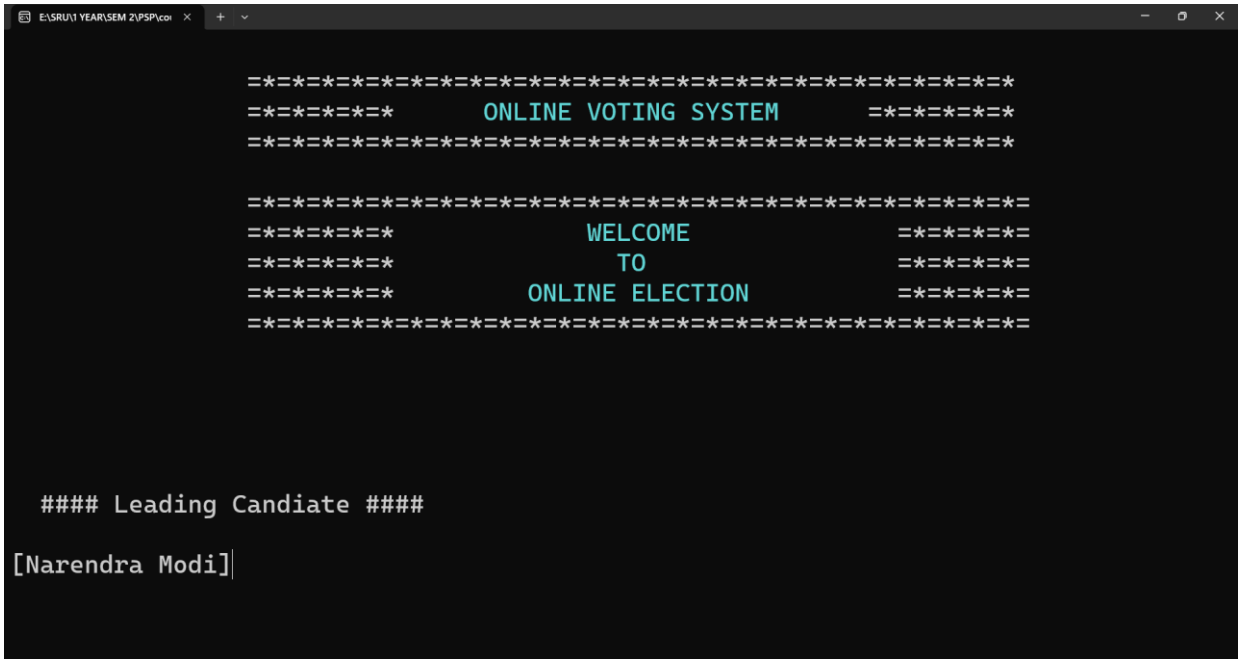
Returning officer login



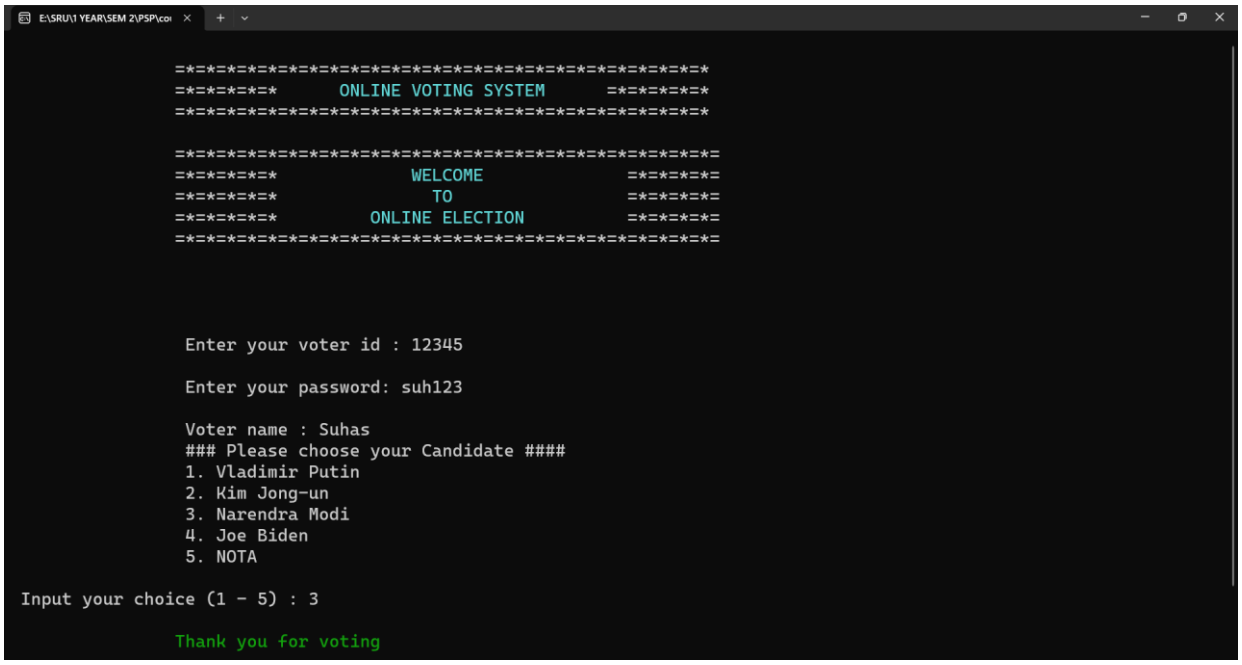
Vote Count

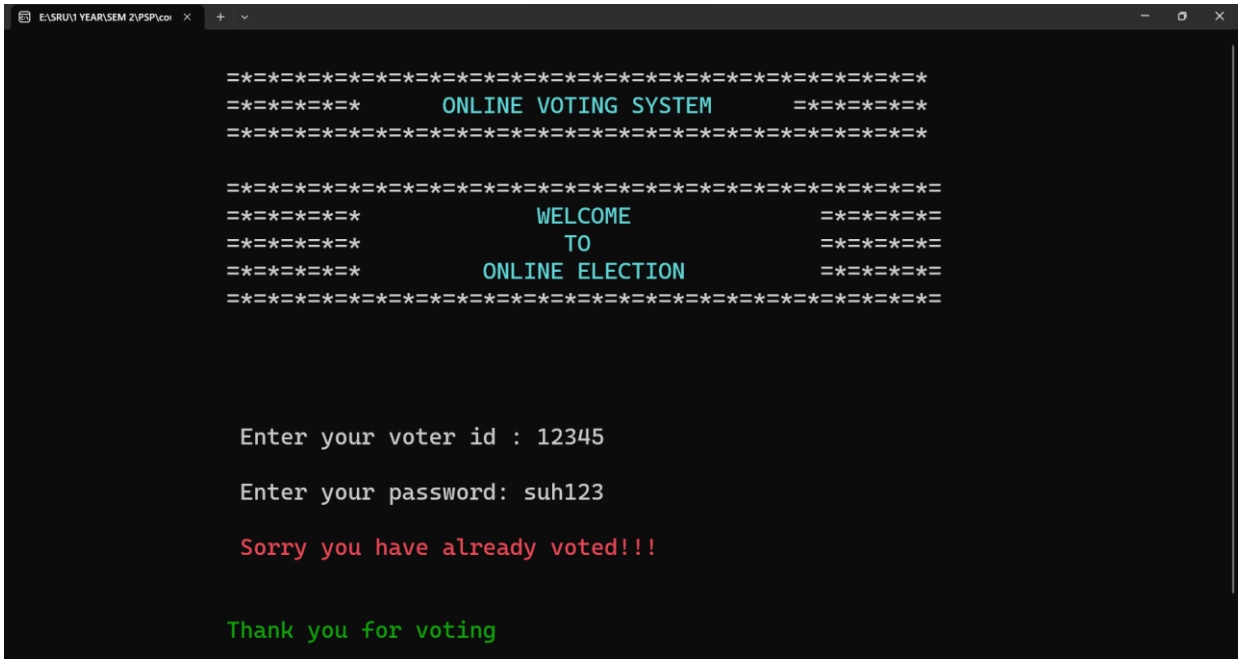


Leading candidate

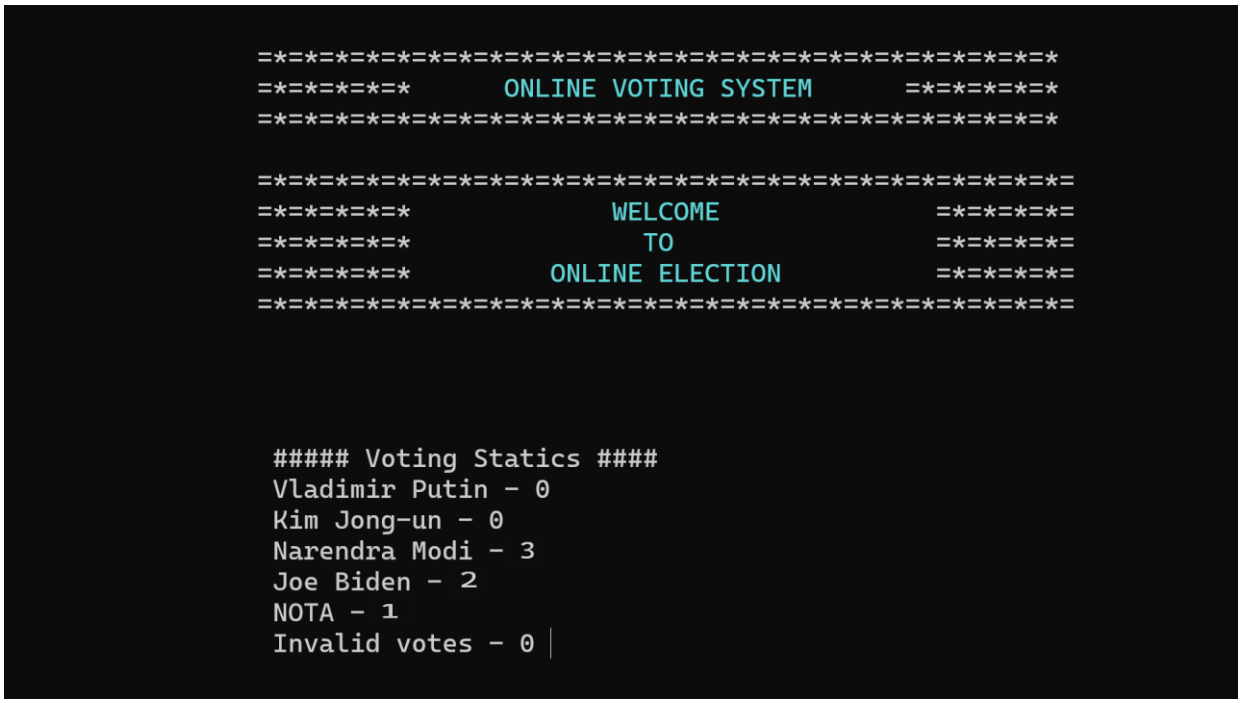


Voter login


E:\SRU\1 YEAR\SEM 2\PSP\co
ONLINE VOTING SYSTEM
WELCOME
TO
ONLINE ELECTION
Enter your voter id : 12345
Enter your password: suh123
Voter name : Suhas
Please choose your Candidate
1. Vladimir Putin
2. Kim Jong-un
3. Narendra Modi
4. Joe Biden
5. NOTA
Input your choice (1 - 5) : 3
Thank you for voting


E:\SRU\1 YEAR\SEM 2\PSP\co
ONLINE VOTING SYSTEM
WELCOME
TO
ONLINE ELECTION
Enter your voter id : 12345
Enter your password: suh123
Sorry you have already voted!!!
Thank you for voting

Voter registration



Exit Screen