ONLINE VOTING SYSTEM REPORT

1. DESCRIPTION:

Describe about the project:

The multifaceted online e-voting system. The proposed system is capable of handling electronic ballots with multiple scopes at the same time,e.g., presidential, municipal, parliamentary, amongst others.The system caters for integrity of an election process in termsof the functional and non-functional requirements.The functional requirements embedded in the design of the proposed system warrant well-secured identification and authentication processes for the voter through the use of combined simple biometrics.The design of the system guarantees that no votes in favor of agiven candidate are lost, due to improper tallying of the voting counts, with the proper incorporation of system FLAG's. Transparency of voting follows through in all phases of an election process to assure the voter that his/her vote went in favor of his/her candidate of choice. Of utmost importance are the requirements forcorrectness, robustness, coherence, consistency, and security.

Requirements:

High Level Requirements:

Features of my project:

The incorporation of biometric technologies can be as simple as using a single biometric. However, a single biometric measure is always subject to security breaches, if not properly attended and administered. This naturally includes security passwords, fingerprints, and signatures, all of which can be spoofed when applied in a non properly attended environment. This is significantly alleviated and system security enhanced with the proper application of combined simple biometric measures.

Low Level Requirements:

How each feature is implemented:

The application of combined weak biometrics leads to systems that are less complex and more robust in terms of the security levels attained. There are strong single biometric measures which involve retinal and iris scans that are rather hard, if not impossible, to breach, but usually lead to more complex systems which, in turn, slow down the underlying biometric matching process due to the amount of data processing involved. For these reasons, amongst others, the type of biometrics addressed in this work is of the former type that involves combined biometrics of the weak types. This will be elaborated upon in the succeeding sections

Swot Analysis:

STRENGTHS

Mobile voting as an aspect of electronic voting cannot be separated from electronic voting. It is an inclusion to electronic voting platform to support mobile devices in voting.

WEAKNESSES

The weakness to the integration and total deployment of mobile technology in voting begins from awareness, policy and technology.

OPPORTUNITIES

Both supporters and opponents of e-voting offer arguments justifying which method is better suited for the electoral process.

4W'S and 1-H

What- voter can use hisher voting right online without any difficulty.

why- While the main perceived benefit to online voting is increasing voter participation.

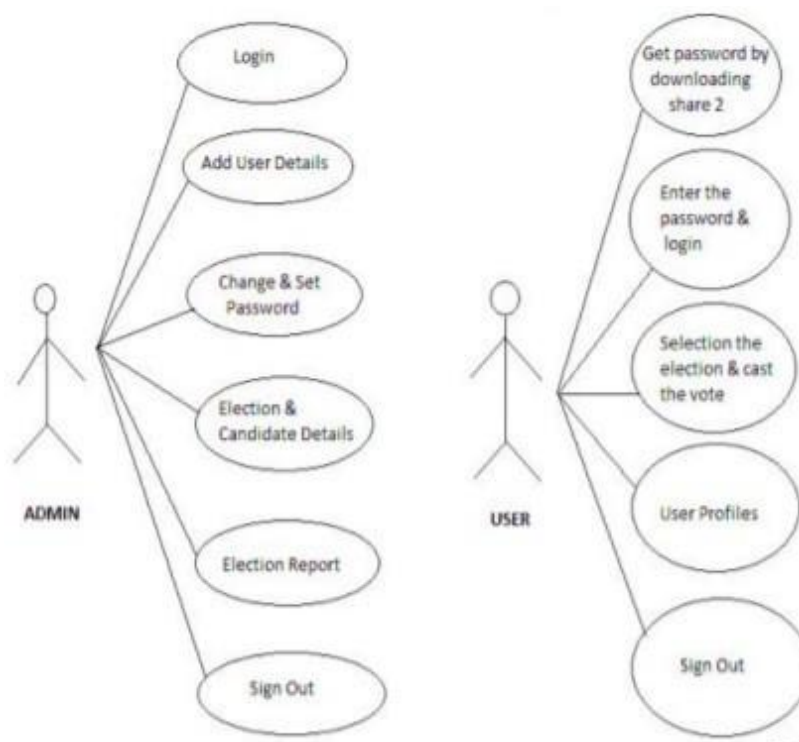where- voters will be able to instantly access the voting system online.

when- in elections.

How-To register to vote, voters are added to an electoral roll of eligible voters, as with any election.
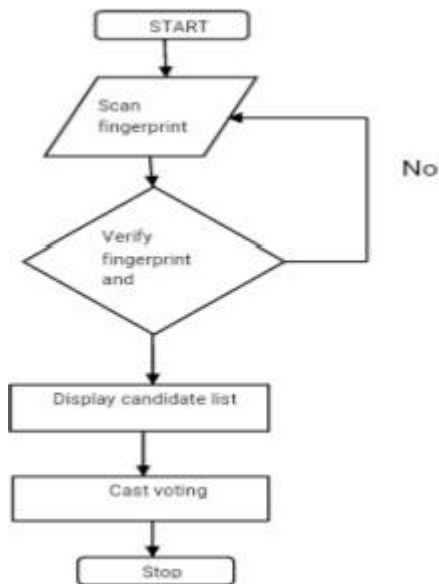
2.DESIGN:

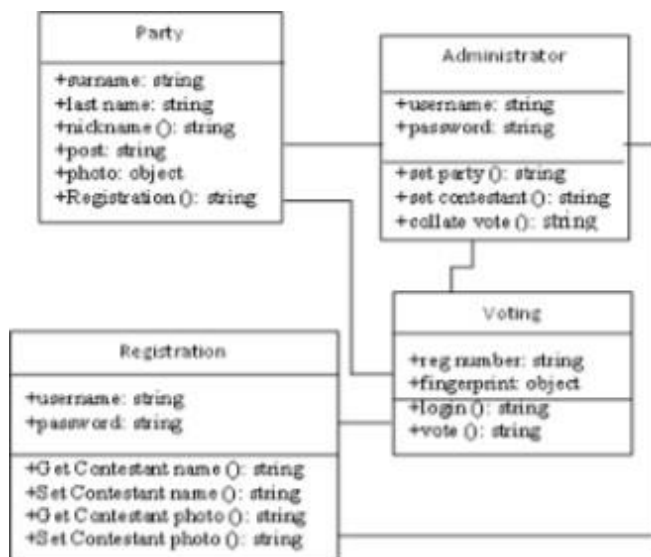**Behavioural Diagram:**

High Level Use Case Behavioural Diagram:
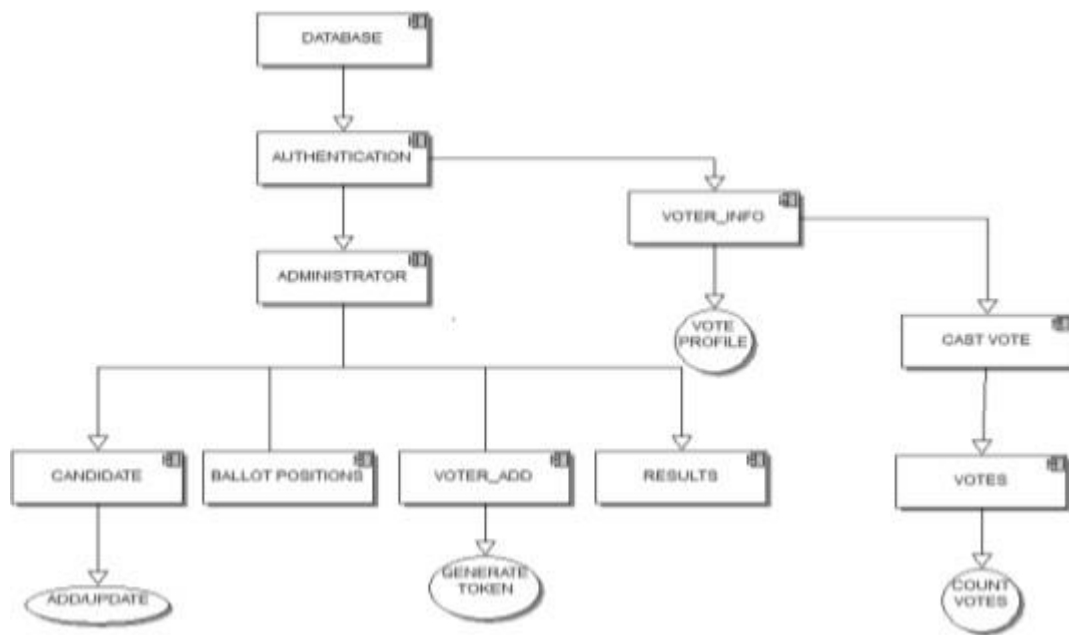
Low Level flow chart Behavioural Diagram:

Structural Diagram:

High Level UML class Structural Diagram:

Low Level UML component Structural Diagram:

3. IMPLEMENTATION:

Code:

```c
#include<stdlib.h>

#include<stdio.h>

#include<string.h>

void printMessageCenter(const char* message)

{

    int len =0;

    int pos = 0;

    //calculate how many space need to print

    len = (78 - strlen(message))/2;

    printf("\t\t\t");

    for(pos =0 ; pos < len ; pos++)

    {

        //print space

        printf(" ");
```

```c
    }
    //print message
    printf("%s",message);
}

void main()
{

    char username[15];
    char password[12];


    printf("Enter your username:\n");
    scanf("%s",&username);


    printf("Enter your password:\n");
    scanf("%s",&password);


    if(strcmp(username,"group6")==0){
        if(strcmp(password,"123")==0){


        printf("\nWelcome.Login Success!");
        welcomeMessage();
        int n,n1,n3,n4,n5,maxi,sum,pos,flag,k=0,co=0,br=0;
        char filename[10];
        char st[100][100],vote[100][100];
        char sy[100],cp[20],t[100];
        int count[100]={0};charn2,a;
        printf("\n");
        printf("\n");
        printf("\n");
```

```c
    printf("\n\n\n");

    headMessage("CANDIDATES");

    printf("\nEnter the number of candidates participating in the election :\n");//entering candidate details

    scanf("%d",&n5);

    int i = 0,j;

    while(i<n5)

    {

        int temp=0;

        printf("\n\n\n");

        printf("==========for %d candidate=========\n",i+1);

        printf("\n\n");

        printf("EntertheNameofthecandidate:\n");

        scanf("%s",st[i]);

        printf("\nEntertheSymbolofthecandidate:\n");

        scanf(" %c",&a);

        for(j = 0; j<i;j++)

        {

            if(sy[j] ==a)

            {

                printf("\n\aThe symbol is already taken.\n\n");

                printf("Pleasere-enterthedetailsofthecandidate.\n");

                temp =1;

                break;

            }

        }

        if (temp == 0)

        {

            sy[i] = a;

            i++;
```

```c
        }


}
printf("\n\n\n");
printf("Enter 1 to vote or any other key to exit the poll :\n");
scanf("%d",&n1);
if(n1!=1)
    exit(1);
while(n1==1)
{printf("\n\n\n");printf("\n\n\n");
    printf("Press entertocontinue.... \n");


    char *date = malloc(sizeof(char) * 8);
    headMessage("VOTER DETAILS");
    printf("\nEnter your voter ID\n");
    scanf("%s",date);
    strcpy(vote[k],date);
    k++;


    if(strcmp(t,date)==0)
        co=-1;


    for(i=0;i<k+1;i++)
    {
        if(strcmp(date,vote[i])==0)
            co++;
    }
    if(co==1)
        flag=0;
    else
        flag=1;
```

```c
        co=0;
if(flag==0)
{


    FILE *fp = fopen("voters.csv", "r");


    if (!fp) {
        printf("Can't open file\n");


    }




    char buf[1024];//Buffer of 1 kilo byte; Can be varied accordingly
    int field_count = 0;


    int entries = 0;
    fgets(buf, 1024, fp);//Skip heading row
    char *field = strtok(buf, ",");
    while(fgets(buf, 1024, fp))
    { field_count = 0;
        field = strtok(buf, ",");//Store next kilobyte into buffer;
        if(strcmp(field,date) == 0)//Check if date matches input
        {
            while(field){
                if (field_count == 0) {
                    printf("\nVOTER ID:\t");
                }
                if (field_count == 1) {
                    printf("NAME:\t");
```

```c
        }
        if (field_count == 2) {
            printf("AGE:\t");
        }


        printf("%s\n", field);
        field = strtok(NULL, ",");


        field_count++;
      }
      printf("\n");

    }

}


  fclose(fp);




  printf("\n\n\n");printf("\n\n\n");
  printf("Press any keytocontinue.......\n");
  getch();



printf("Cast your vote by selecting the symbol of your representative\n");
    headMessage("CASTING THE VOTE");
    printf("\n");printf("\n");printf("\n");
    printf("Enter the symbol of your Representative :\n");
    for(i=0;i<n5;i++)
    {


      printf("%d. %s (%c)\n",i+1,st[i],sy[i]);

    }
```

```c
int fl=0;

scanf(" %c",&n2);

for(i=0;i<n5;i++)

{


if(n2==sy[i])

    {fl=1;

    break;

    }

else

    fl=0;

}


if(fl==1)


{



for(i=0;i<n5;i++)

{


if(n2==sy[i])

    count[i]++;



}


printf("\aThank You ! Your vote has been casted.\n");printf("\n\n\n");printf("\n\n\n");

printf("Press any key to continue\n");

}

else{
```

```c
            strcpy(t,date);

            printf("\n\n\n");

            printf("\aWrong symbol is entered. Enter the correct symbol\n");

            printf("The details are to be re-entered.\n");

            printf("\n\n\n");

            printf("Press any key to continue\n");


            continue;
        }


    headMessage("MENU");


    while(n1==1)


    {
        printf("\n");printf("\n");printf("\n");


printf("Enter your choice \n1. Results\n2. Difference between the votes of any two candidates\n3.
Finish your voting\n4. EXIT the Poll\n ");
        scanf("%d",&n);


        if(n==1)
        {
            headMessage("Results");
            printf("\n\n\n");
            printf("The Result according to votes casted so far is as follows : \n");
            printf("\n");printf("\n");printf("\n");
            for(i=0;i<n5;i++)
            {
```

```c
        printf(" %s( %c): %d\n",st[i],sy[i],count[i]);

        }

    }
    else if(n==2)

    {

        headMessage("Difference");

        printf("\n\n\n");

        for(i=0;i<n5;i++)

    {


        printf("%d. %s (%c)\n",i+1,st[i],sy[i]);

    }

        printf("Enter the serial nos. of whose vote count differences are to be known : \n");


        scanf("%d%d",&n3,&n4);

        if(count[n3-1]==count[n4-1])

            printf("It's a Tie between both the candidates'\n");

        printf("\n\n\n");

        printf("The difference between %s and %s is %d",st[n3-1],st[n4-1],count[n3-1]-count[n4-1]);

    }
    else if(n==3)

    {

        break;

    }
    else if(n>4)

    {


        printf("\aWRONG OPTION ENTERED TRY AGAIN");

    }
```

```c
        else

            n1=0;


    }
}



    else
        printf("\a\a\a\a\aError!!!! You already casted your vote ");printf("\n\n\n");

        printf("press any key to continue\n");
}
headMessage(" FINAL RESULTS");
for(i=0; i<n5; i++)

    {


        /* File name for storing Candidate details */
        sprintf(filename, "file%d.apk", i+1);
        FILE * fPtr;


        /* If above operation is failed print error and exit */
        if(fPtr == NULL)
        {
            /* File not created hence exit */
            printf("Unable to create file.\n");
            exit(EXIT_FAILURE);
        }


        /* Writing data in the file */
        fPtr = fopen(filename, "w");
        printf("\nCandidate : %s",st[i]);
```

```c
            fprintf(fPtr,"candidate:%s\n",st[i]);


            printf("\t Symbol : %c",sy[i]);
            fprintf(fPtr,"symbol:%c\n",sy[i]);


            printf("\t Votes : %d\n",count[i]);
            fprintf(fPtr,"votes:%d\n",count[i]);


            /* Closing the file */
            fclose(fPtr);
        }



    }else{
    printf("\nwrong password\n");
    main();
}


    }else{
    printf("\nUser doesn't exist\n");
    main();
}
}
```