Online Voting System

USING

PHP with MySQL

Project work submitted to the

BHARATHIDASAN UNIVERSITY, TIRUCHIRAPALLI

In partial fulfillment of the requirement for the award of the degree of

B.Sc., INFORMATION TECHNOLOGY

(BACHELOR OF INFORMATION TECHNOLOGY)

(2022-2025)

Submitted by

NAME	REG.NO
M.RAJASEKAR	CB22S619321
G.SABARISH	CB22S619322
P.SANJAI	CB22S619324

Under the Guidance of

Prof.S.KAVIBHARATHI, M.C.A.,

April 2025

DEPARTMENT OF COMPUTER SCIENCE

SRINIVASAN COLLEGE OF ARTS & SCIENCE

(Affiliated to Bharathidasan University, Tiruchirappalli)
(Recognized under the Sections 2(f) & 12B of the UGC Act, 1956)
PERAMBALUR-621 212

DEPARTMENT OF COMPUTER SCIENCE

SRINIVASAN COLLEGE OF ARTS & SCIENCE

PERAMBALUR-621 212

CERTIFICATE

This is to certify that the project entitled "Online Voting System" using PHP with MySQL is a bonafide record work done by the following members.

NAME	REG.NO
M.RAJASEKAR	CB22S619321
G.SABARISH	CB22S619322
P.SANJAI	CB22S619324

In partial fulfillment of the requirement for the award of degree of Bachelor of Computer Science during the year 2022-2025.

Guide **Head of the Department**

Examiner

1.

2.

DEPARTMENT OF COMPUTER SCIENCE

SRINIVASAN COLLEGE OF ARTS &SCIENCE

Perambalur- 621 212.

Mr. **A.MOHAMED AZHARUDHEEN**., M.C.A.,M.Phil,

HEAD & Assistant Professor, Department of Computer Science

Srinivasan College of Arts & Science

Perambalur.

BONAFIDE CERTIFICATE

Certified that this project titled **"Online Voting System"** done by the following member,

NAME	REG.NO
M.RAJASEKAR	CB22S619321
G.SABARISH	CB22S619322
P.SANJAI	CB22S619324

Who carried out this knowledge, under my supervision? Certified the work reported there in does not from part of any other project or reported there in does not from part of any other project work on he basis of which a degree or award was conferred on an earlier occasion on his or other candidate.

Guide

(S.KAVIBHARATHI)

ACKNOWLEDGEMENTS

I would like to take pleasure in recording our deep sense of gratitude to my family and friends. We would like to express our sincere thanks to our Esteemed Chancellor **Shri.A.Srinivasan Dhanalakshmi Srinivasan University,** our Secretary sir **Thiru.P.Neelaraj**, of our college for arranging the lab facilities to complete our project work.

We express our sincere thanks to our Principal **Dr.N.Vetrivelan**., MCA., M.Phil.,Ph.D., and Prof. V.Chandra Chowdry, DEAN(Academic) for providing the necessary facilities in this college for the successful completion of this project.

We wish to express our sincere thanks to **Prof.A.Mohamed Azharudheen,** MCA., M.Phill., (Ph.D) HEAD, department of computer Science for his invaluable support and helping us in various ways throughout this project.

We would like to express our profound gratitude to our guide **Mr.S.Kavibharathi.,**M.C.A., has been taken an interest in this project and guided us with great deal of patience, their advice support and involvement and inspiration.

"The teacher is like a candle which rights others in consuming itself ". Hence I take immense pleasure to thank our respected "Department Staff" for being a great support to me.

CONTENT

ABSTRACT

The College internet Voting System is an internet platform that streamlines and automates online polls and elections. Every user is interacting with the system on their own. The user's browser is used for all remote user interaction. Voters should first fill out the registration form and submit their information. This information is compared to the information in the college database; On Election Day alone, users check in to the system with their username and password and cast their votes. There are two user levels in this system: administrator and voter levels. The centralized application server used in this project can centrally handle hundreds of terminals to provide comprehensive centralized access control depending on time. The created code will communicate with client and admin processes. The system will receive the accessed data via a centralized server, and the MySQL Server will verify the data before saving and retrieving it. It eliminates the need for paper-based voting, making the election process ecofriendlier and time-efficient. With a user-friendly interface and robust security mechanisms, the Online College Voting System provides a reliable and streamlined solution for conducting college elections effectively.

1. INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

The main conduit between the student body and the administration is provided by university student leaders. In accordance with the university act, these leaders are so democratically elected to represent the interests of the student body. Every student expects that elections will always be conducted fairly and that the results will be tabulated correctly. Due to various difficulties on Election Day, there have been issues with voter turnout in the past elections. At first, the students were supposed to line up and cast their votes in the polls according to the different schools. Since there is no prior voter registration, the existing system does not verify or account for the individuals who wish to cast a ballot. Today, there is not enough information available, which makes manual creation difficult. Because of this, a lot of manual voting systems have developed a method that ensures a high degree of commitment to identifying the client's true appearance. Our focus in this project, College Online Election, is on the web-based voting framework process that is used to identify the real voters. The system comprises two primary modules: Admin and Student. The Admin is responsible for managing the system, including logging in, adding candidate details, and overseeing the election process. Once the candidates are registered, students can participate in the voting process by first registering on the platform and logging into their accounts. After authentication, students can view the list of candidates along with their details before casting their votes securely. The system ensures that each student can vote only once, preventing any fraudulent activities. It incorporates encryption techniques to maintain the integrity and confidentiality of the votes. Once the voting period ends, the Admin has the authority to finalize and publish the election results on the website, making them accessible to all students.. To create a user-friendly web based application for Colleges and Universities to conduct safe, secure, timesaver and smooth elections for various postings. Students from all over the world can vote for their representatives using the online voting system. It can be referred to as "Internet Voting" because it is conducted online. It appears to accurately capture the voter's preferences.

1.2 PROBLEM STATEMENT

The Online College Voting System is a web-based application designed to digitize and streamline the election process within a college environment, ensuring fairness, security, and transparency. The system consists of two primary modules: Admin and Student. The Admin is responsible for managing the system, including logging in, adding candidate details, overseeing the election process, and publishing the final results. Students can register and log into the system to participate in the voting process. After authentication, they can view candidate details and cast their votes securely. The system ensures that each student votes only once, preventing duplication and unauthorized access. Once the voting period ends, the Admin verifies and publishes the final results on the platform. The system eliminates traditional paper-based voting, reducing manual workload, errors, and the risk of fraudulent activities. It offers a userfriendly interface, real-time vote counting, and enhanced security features such as encrypted vote storage. By automating the election process, the Online College Voting System ensures efficiency, accuracy, and accessibility, making student elections more reliable and transparent.

2. SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

The E-voting system is that the system enforced to create the legal system simple for both voting and conjointly for the administrator to look at and check the result for every space one by one. Here the choice is completed anyplace with in spite of the world within which they reside. The voters or the administrator will login as long as they have already got a user id and countersign. New user may be solely created by the administrator. The elector will login together with his user id as long as he has not casted the vote already. If he has already casted the vote the elector standing is modified to voted and he cannot login there upon user id unless the administrator refreshes the voting system. In existing system the results will be modified easily by third parties. There is no way to protect data in server. Server can be hacked by third parties and fake results will be announced easily. So the existing approach does not provide trusted environment for online voting process.

2.1.1 DISADVANTAGES OF EXISTING SYSTEM

- The voting data transmitted and stored on the database are all open and transparent.
- Possible to data modification.
- Possible for Fault result announcement based on modified data.

2.2 PROPOSED SYSTEM

To implement an e-voting scheme based on web based technology that meets the fundamental e-voting properties whilst, at the same time, provides a degree of decentralization and places as much control of the process in the hands of the voters as was deemed possible. The system has been designed to support a voting application in the real world environment taking into account specific requirements such as privacy, eligibility, convenience, receipt-freeness and verifiability. The proposed system aims to achieve secure digital voting without compromising its usability. The proposed system will provide online voters registration forms for students which they will fill and upon registration of their details, they will be allowed to log in and interact with the system. The student details will be saved in the student details database.

2.2.1 ADVANTAGES OF PROPOSED SYSTEM

- Only eligible voters are allowed to vote.
- Every voter shall cast only one vote.
- It must be impossible to change anybody's vote.
- Elimination of error handling through real-time tracking of voting result with no double spending.
- The web application guarantees traceability and non-degradability of information.
- It decreases the success rate of attackers.
- It provides security, transparency and efficiency.

3. SYSTEM SPECIFICATION

3.1 HARDWARE REQUIREMENTS

Processor	: Dual core processor 2.6.0 GHZ
RAM	: 4GB
Hard disk	: 320 GB
Compact Disk	: 650 Mb
Keyboard	: Standard keyboard

3.2 SOFTWARE REQUIREMENTS

Operating system	: Windows OS
Front End	: PHP 5.2.6
Back End	: MYSQL 5.0.51b
Web Server	: XAMP Server 2.0
IDE	: Macromedia Dreamweaver 8 (Adobe)

4. SOFTWARE DESCRIPTION

4.1.1 FRONT END

Front End: PHP

PHP is an open-source server-side scripting language that many devs use for web development. It is also a general-purpose language that you can use to make lots of projects, including Graphical User Interfaces (GUIs). In this article, I will help you explore the world of PHP so you can learn how it works and its basic features. By the end, you will be able to write your first Hello World program in PHP. The term PHP is an acronym for PHP: Hypertext Preprocessor. PHP is a server-side scripting language designed specifically for web development. It is open-source which means it is free to download and use. It is very simple to learn and use. The files have the extension ".php".

Rasmus Lerdorf inspired the first version of PHP and participated in the later versions. It is an interpreted language and it does not require a compiler.

- PHP code is executed in the server.
- It can be integrated with many databases such as Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Sybase, and Informix.
- It supports main protocols like HTTP Basic, HTTP Digest, IMAP, FTP, and others.
- Websites like www.facebook.com and www.yahoo.com are also built on PHP.
- One of the main reasons behind this is that PHP can be easily embedded in HTML files and HTML codes can also be written in a PHP file.
- The thing that differentiates PHP from the client-side language like HTML is, that PHP codes are executed on the server whereas HTML codes are directly rendered on the browser. PHP codes are first executed on the server and then the result is returned to the browser.
- The only information that the client or browser knows is the result returned after executing the PHP script on the server and not the actual PHP codes present in the PHP file. Also, PHP files can support other client-side scripting languages like CSS and JavaScript.

Characteristics of PHP

- Simple and fast
- Efficient
- Secured
- Flexible
- Cross-platform, it works with major operating systems like Windows, Linux, and macOS.
- Open Source
- Powerful Library Support
- Database Connectivity

Advantages of PHP

PHP has some advantages that have made it so popular, and it's been the go-to language for web servers for more than 15 years now. Here are some of PHP's benefits:

- Cross-Platform: PHP is platform-independent. You don't have to have a particular OS to use it because it runs on every platform, whether it's Mac, Windows, or Linux.
- Open Source: PHP is open source. The original code is made available to everyone who wants to build upon it. This is one of the reasons why one of its frameworks, Laravel, is so popular.
- Easy to learn: PHP is not hard to learn for absolute beginners. You can pick it up pretty if you already have programming knowledge.
- PHP syncs with all Databases: You can easily connect PHP to all Databases, relational and non-relational. So it can connect in no time to MySQL, Postgress, MongoDB, or any other database.
- Supportive Community: PHP has a very supportive online community. The official documentation provides guides on how to use the features and you can easily get your problem fixed while stuck.

Syntax of PHP

The structure which defines PHP computer language is called PHP syntax. The PHP script is executed on the server and the HTML result is sent to the browser. It can normally have HTML and PHP tags. PHP or Hypertext Preprocessor is a widely used open-source general-purpose scripting language and can be embedded with HTML. PHP files are saved with the ".php" extension. PHP scripts can be written anywhere in the document within PHP tags along with normal HTML.

- You can embed PHP code anywhere in a document. It starts with an opening tag of <?php (the PHP code goes in here) and ends with a closing tag ?>.
- All PHP statements end with a semicolon ;. A PHP file is always named with the file extension of .php – for example, index.php or home.php.

Data Types in PHP

Variables in PHP store values of different data types. Now let's discuss some data types that work with PHP:

- String
- Integer
- Float
- Boolean

String data type

A string is a data type which is represented with some text inside double quotes " ". A string

Integer data type

Integers are whole numbers that have no decimal point. Integers can either be negative numbers (-34567) or positive numbers (34567).

Float data type

Floats are not whole numbers, but rather they are numbers with decimal points. Floats can also be negative decimal numbers (-34.567) or positive decimal numbers (34.567).

Boolean data type

Boolean is a data type that represents two possible outcome, true or false. Booleans are used mostly when we are working with conditional statements like if, else, elseif, and swtich.

Function of PHP

PHP functions are similar to other programming languages. A function is a piece of code which takes one more input in the form of parameter and does some processing and returns a value. You already have seen many functions like fopen() and fread() etc. They are built-in functions but PHP gives you option to create your own functions as well.

There are two parts which should be clear to you −

- Creating a PHP Function
- Calling a PHP Function

In fact you hardly need to create your own PHP function because there are already more than 1000 of built-in library functions created for different area and you just need to call them according to your requirement.

Please refer to PHP Function Reference for a complete set of useful functions.

Creating a Function

While creating a user defined function we need to keep few things in mind:

- Any name ending with an open and closed parenthesis is a function.
- A function name always begins with the keyword function.
- To call a function we just need to write its name followed by the parenthesis
- A function name cannot start with a number. It can start with an alphabet or underscore.
- A function name is not case-sensitive.

Reusability: If we have a common code that we would like to use at various parts of a program, we can simply contain it within a function and call it whenever required. This reduces the time and effort of repetition of a single code. This can be done both within a program and also by importing the PHP file, containing the function, in some other program

Easier error detection: Since, our code is divided into functions, we can easily detect in which function, the error could lie and fix them fast and easily.

Easily maintained: As we have used functions in our program, so if anything or any line of code needs to be changed, we can easily change it inside the function and the change will be reflected everywhere, where the function is called. Hence, easy to maintain.

Creating PHP Function

It's very easy to create your own PHP function. Suppose you want to create a PHP function which will simply write a simple message on your browser when you will call it. Following example creates a function called write Message () and then calls it just after creating it.

PHP Functions with Parameters

PHP gives you option to pass your parameters inside a function. You can pass as many as parameters your like. These parameters work like variables inside your function. Follow

Passing Arguments by Reference

It is possible to pass arguments to functions by reference. This means that a reference to the variable is manipulated by the function rather than a copy of the variable's value. Any changes made to an argument in these cases will change the value of the original variable. You can pass an argument by reference by adding an ampersand to the variable name in either the function call or the function definition.\

PHP Functions returning value

A function can return a value using the return statement in conjunction with a value or object. return stops the execution of the function and sends the value back to the calling code. You can return more than one value from a function using return array(1,2,3,4).

4.2 BACK END

BACKEND: MYSQL SERVER

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

MySQL databases are relational.

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and "pointers" between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

The SQL part of "MySQL" stands for "Structured Query Language". SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, "SQL-92" refers to the standard released in 1992, "SQL:1999" refers to the standard released in 1999, and "SQL:2003" refers to the current version of the standard. We use the phrase "the SQL standard" to mean the current version of the SQL Standard at any time.

MySQL software is Open Source.

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), http://www.fsf.org/licenses/, to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information (http://www.mysql.com/company/legal/licensing/).

The MySQL Database Server is very fast, reliable, scalable, and easy to use.

If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). We also provide MySQL Server as an embedded multithreaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

A large amount of contributed MySQL software is available.

MySQL Server has a practical set of features developed in close cooperation with our users. It is very likely that your favorite application or language supports the MySQL.

Connectivity

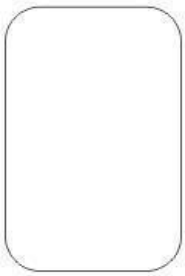
- Clients can connect to MySQL Server using several protocols:
- Clients can connect using TCP/IP sockets on any platform.
- On Windows systems, clients can connect using named pipes if the server is started with the named_pipe system variable enabled. Windows servers also support shared-memory connections if started with the shared_memory system variable enabled. Clients can connect through shared memory by using the --protocol=memory option.
- On Unix systems, clients can connect using Unix domain socket files.
- MySQL client programs can be written in many languages. A client library written in C is available for clients written in C or C++, or for any language that provides C bindings.
- APIs for C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, and Tcl are available, enabling MySQL clients to be written in many languages. See Chapter 29, Connectors and APIs.
- The Connector/ODBC (MyODBC) interface provides MySQL support for client programs that use ODBC (Open Database Connectivity) connections. For example, supported, as are many others. See MySQL Connector/ODBC Developer Guide.
- The Connector/J interface provides MySQL support for Java client programs that use JDBC connections. Clients can be run on Windows or Unix. Connector/J source is available. See MySQL Connector/J 5.1 Developer Guide.
- MySQL Connector/NET enables developers to easily create .NET applications that require secure, high-performance data connectivity with MySQL. It implements the required ADO.NET interfaces and integrates into ADO.NET aware tools. Developers can build applications using their choice of .NET languages. MySQL Connector/NET is a fully managed ADO.NET driver written in 100% pure C#. See MySQL Connector/NET Developer Guide.

5. SYSTEM DESIGN

5.1 DATAFLOW DIAGRAM

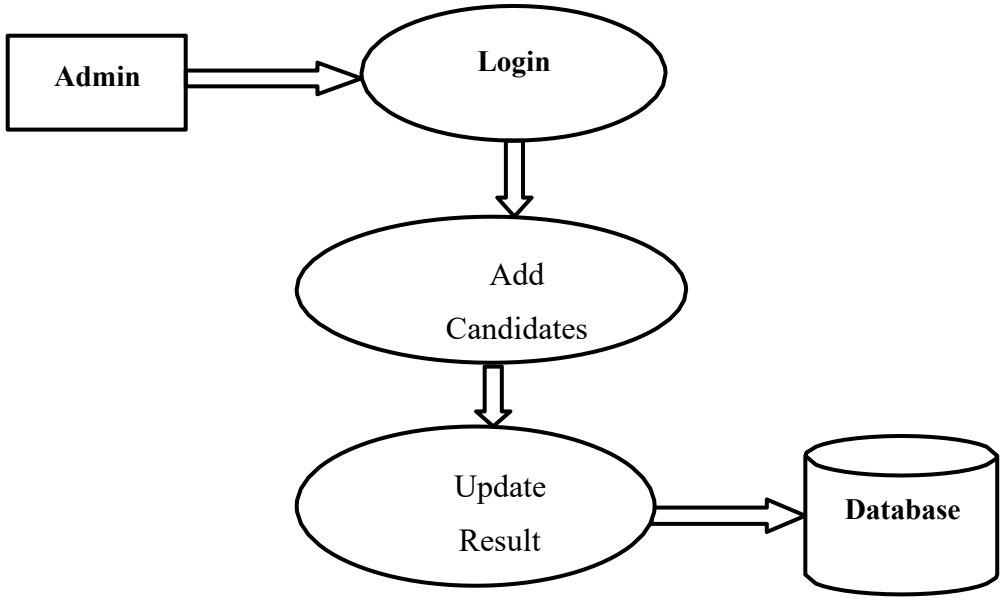
A two-dimensional diagram explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output. Individuals seeking to draft a data flow diagram must identify external inputs and outputs, determine how the inputs and outputs relate to each other, and explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects.

Data flow Symbols:

Symbol	Description
	An **entity**. A source of data or a destination for data.
	A **process** or task that is performed by the system.
	A **data store**, a place where data is held between processes.
	A **data flow**.

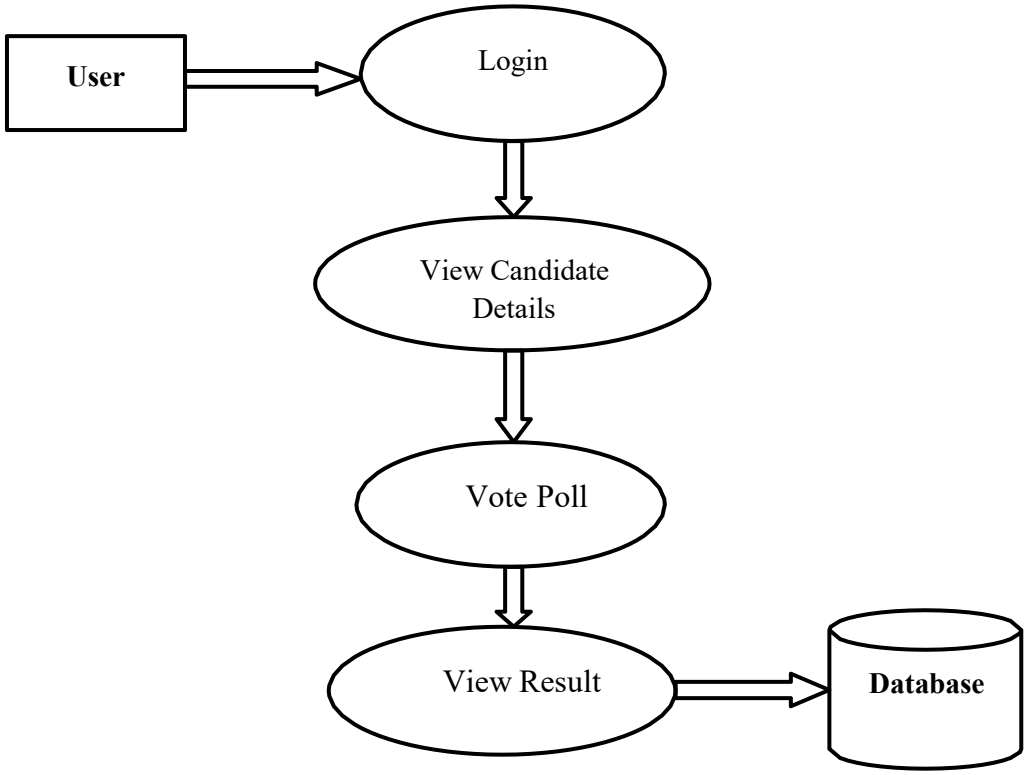
LEVEL 0 (ADMIN)

The Level 0 DFD shows how the system is divided into 'sub-systems' (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job, and shows the flow of data between the various parts of the system.



LEVEL-1 (USER)

The next stage is to create the Level 1 Data Flow Diagram. This highlights the main functions carried out by the system. As a rule, to describe the system was using between two and seven functions - two being a simple system and seven being a complicated system. This enables us to keep the model manageable on screen or paper.



5.2 DATABASE DESIGN

A table is a data structure that organizes information into rows and columns. It can be used to both store and display data in a structured format. For example, databases store data in tables so that information can be quickly accessed from specific rows. Websites often use tables to display multiple rows of data on page. Spreadsheets combine both purposes of a table by storing and displaying data in a structured format.

Databases often contain multiple tables, with each one designed for a specific purpose. For example, a company database may contain separate tables for employees, clients, and suppliers. Each table may include its own set of fields, based on what data the table needs to store. In database tables, each field is considered a column, while each entry (or record), is considered a row. A specific value can be accessed from the table by requesting data from an individual column and row.

Table Name: admin

Column	Type	Null	Default
Name	Varchar (50)	No	NULL
Psw	Varchar (50)	No	NULL

Table Name: candidates

Column	Type	Null	Default
Id	Int (11)	No	NULL
Name	Varchar (255)	No	NULL
Position	Varchar (255)	No	NULL
Party	Varchar (255)	No	NULL
Experience	Int (11)	No	NULL
Bio	Text	No	NULL
Manifesto	Text	No	NULL
Photo	Varchar (255)	No	NULL
Election_id	Int (11)	No	NULL

Table Name: elections

Column	Type	Null	Default
Id	Int (11)	No	NULL
Name	Varchar (255)	No	NULL
description	text	Yes	NULL
Start_date	date	No	NULL
end_date	date	No	NULL
results_announced	Tinyint (1)	Yes	0

Table Name: Stureg

Column	Type	Null	Default
Id	Int (11)	No	NULL
Name	Varchar (255)	Yes	NULL
Dob	Date	Yes	NULL
gender	Varchar (10)	Yes	NULL
email	Varchar (255)	Yes	NULL
year	Varchar (20)	Yes	NULL
dep	Varchar (50)	Yes	NULL
Uname	Varchar (50)	Yes	NULL
psw	Varchar (255)	Yes	NULL
status	Enum (' pending', 'approved', 'rejected')	Yes	pending

Table Name: votes

Column	Type	Null	Default
Id	Int (11)	No	NULL
student_id	Int (11)	Yes	NULL
candidate_id	Int (11)	Yes	NULL
election_id	Int (11)	No	NULL
Disabled	tinyint (1)	Yes	0

6. SYSTEM TESTING

6.1 TYPES OF TESTING

Testing is a series of different tests that whose primary purpose is to fully exercise the computer based system. Although each test has a different purpose, all work should verify that all system element have been properly integrated and performed allocated function. Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system. The philosophy behind testing is to find the errors. A good test is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers the undiscovered error. Test cases are devised with this purpose in mind.

A test case is a set of data that the system will process as an input.

SYSTEM TESTING

After a system has been verified, it needs to be thoroughly tested to ensure that every component of the system is performing in accordance with the specific requirements and that it is operating as it should including when the wrong functions are requested or the wrong data is introduced. Testing measures consist of developing a set of test criteria either for the entire system or for specific hardware, software and communications components. For an important and sensitive system such as an electronic voting system, a structured system testing program may be established to ensure that all aspects of the system are thoroughly tested.

UNIT TESTING

The first test in the development process is the unit test. The source code is normally divided into modules, which in turn are divided into smaller units called units. These units have specific behaviour. The test done on these units of code is called unit test. Unit test depends upon the language on which the project is developed. Unit tests ensure that each unique path of the project performs accurately to the documented specifications and contains clearly defined inputs and expected results. Functional and reliability testing in an Engineering environment. Producing tests for the behaviour of components (nodes and vertices) of a product to ensure their correct behaviour prior to system integration.

VALIDATION TESTING

Valid and invalid data should be created and the program should be made to process this data to catch errors. When the user of each module wants to enter into the page by the login page using the use rid and password .If the user gives the wrong password or use rid then the information is provided to the user like “you must enter user id and password”. Here the inputs given by the user are validated. That is password validation, format of date are correct, textbox

validation. Changes that need to be done after result of this testing

7. SYSTEM IMPLEMENTATION

7.1 NAME OF THE MODULES

MODULES

ADMIN

- Login
- View Request
- Accept/ Reject
- Add Candidate
- Update Voting Result

VOTER

- Register
- Login
- View Candidate
- Vote
- View Result

7.2 MODULE DESCRIPTION

Admin

- Login

  In this module, the admin can login in the system using his/her username and password.

- Approve/ reject request

  In this module, the admin can approve/ reject the candidate request.

- Add Candidates

  In this module, the admin can add the candidate information like candidate name, id, phone number, mail id, location etc.

- Update Result

Student

- Register

  In this module, there is registration form available where new student can create their account by providing required information to the system. The registration form details are like name, email, gender, mobile number, address, and etc. These details are stored in the database. And then can getting to the username and password in the system.

- Login

  In this module, the student can login the system using his/her user name and password.

- View Candidates

  In this module, the voter can view the candidate details like candidate name, id, phone number, mail id, location etc.

- Vote poll

  In this module, user can view the candidate details who are stand for the voting and vote the particular candidate.

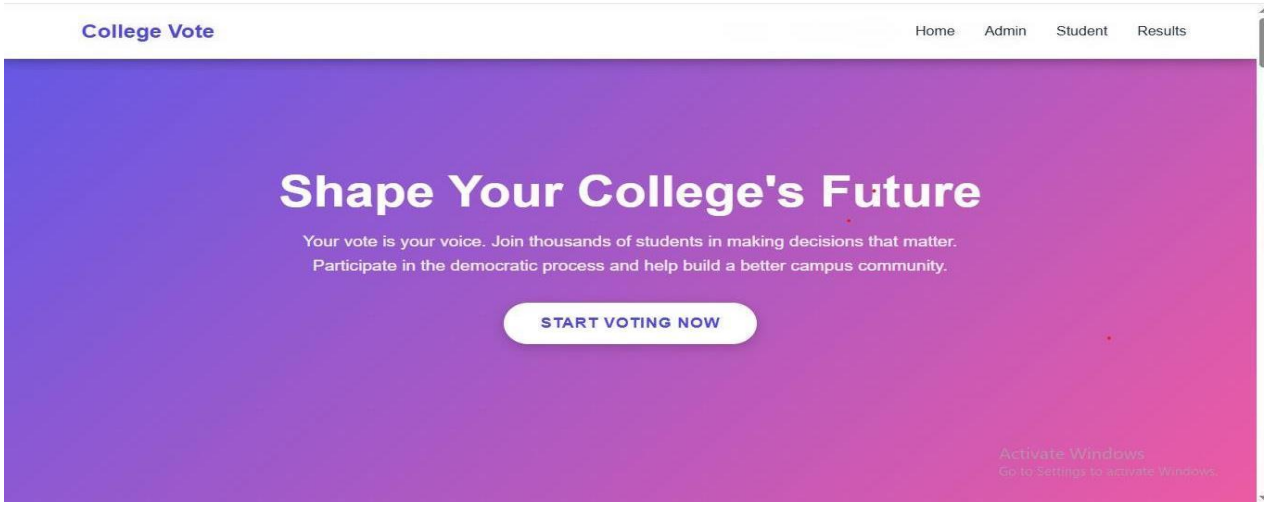
- View result

  In this module, user can view the vote result updated by the admin.

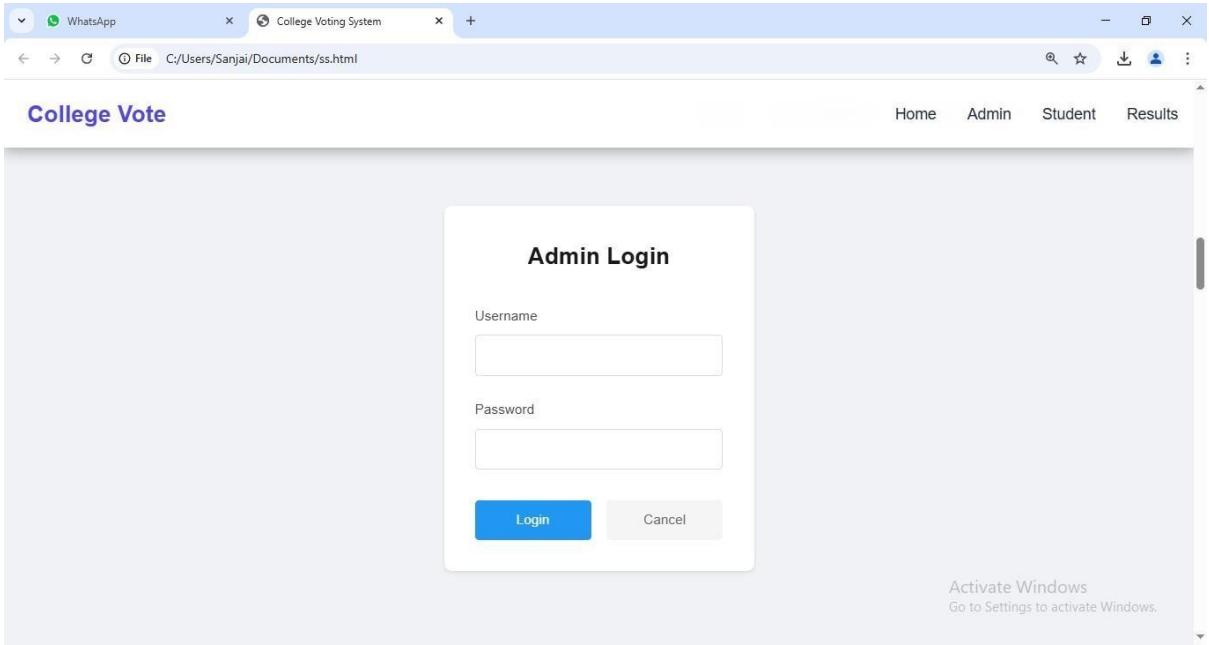
8. APPENDICES

8.1 SCREENSHOTS

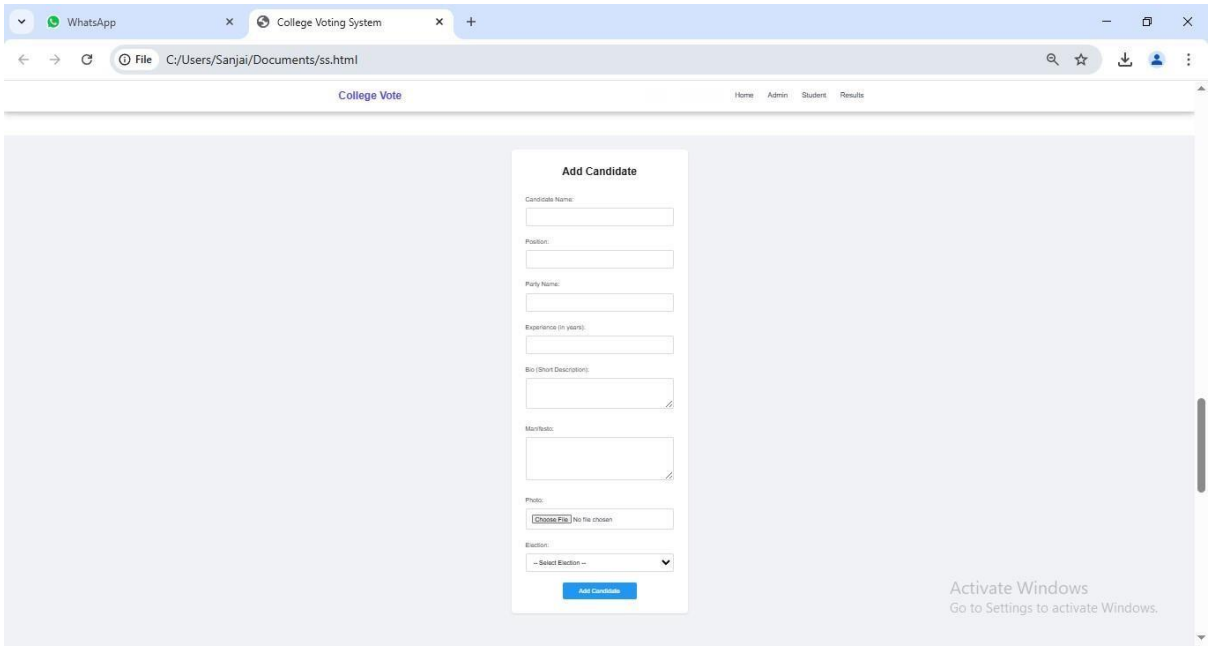
Profile

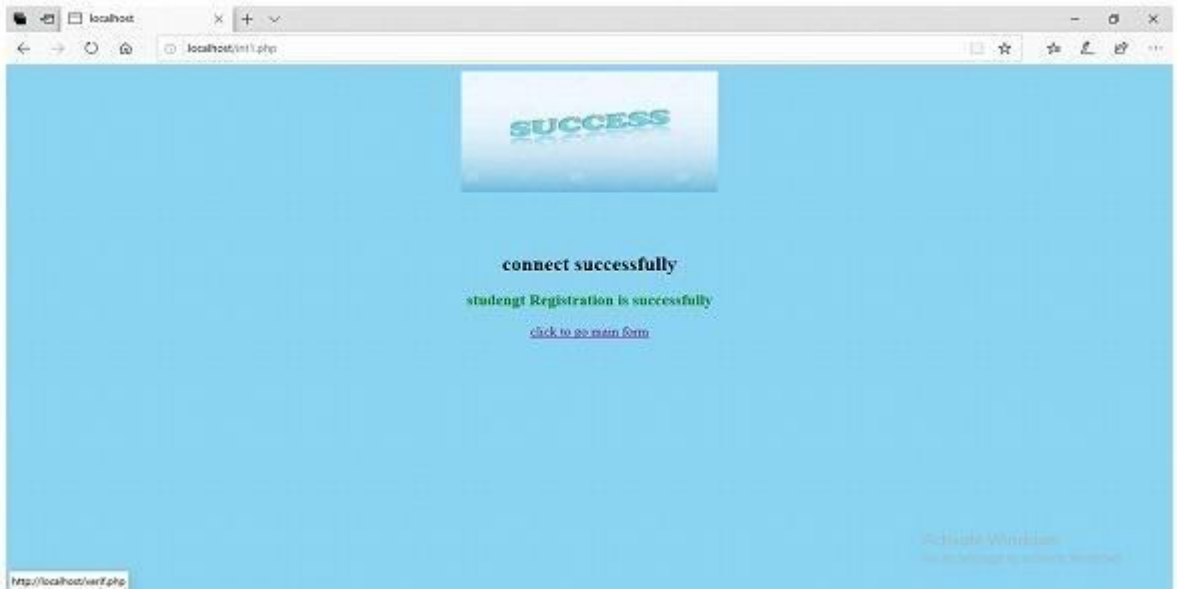


Admin Login

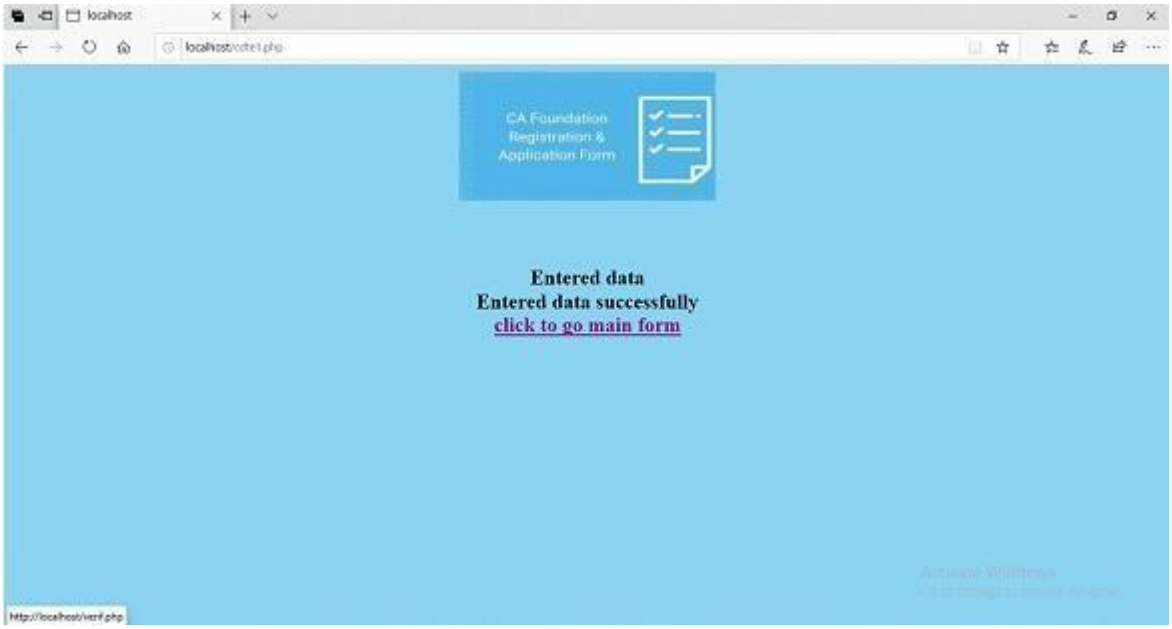


Student Login

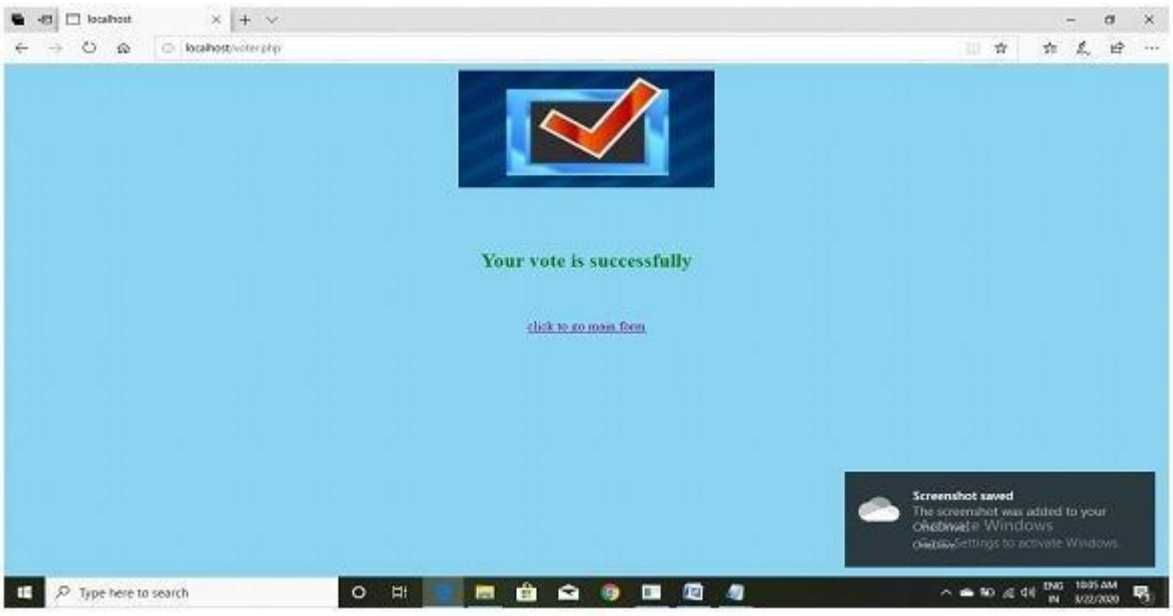




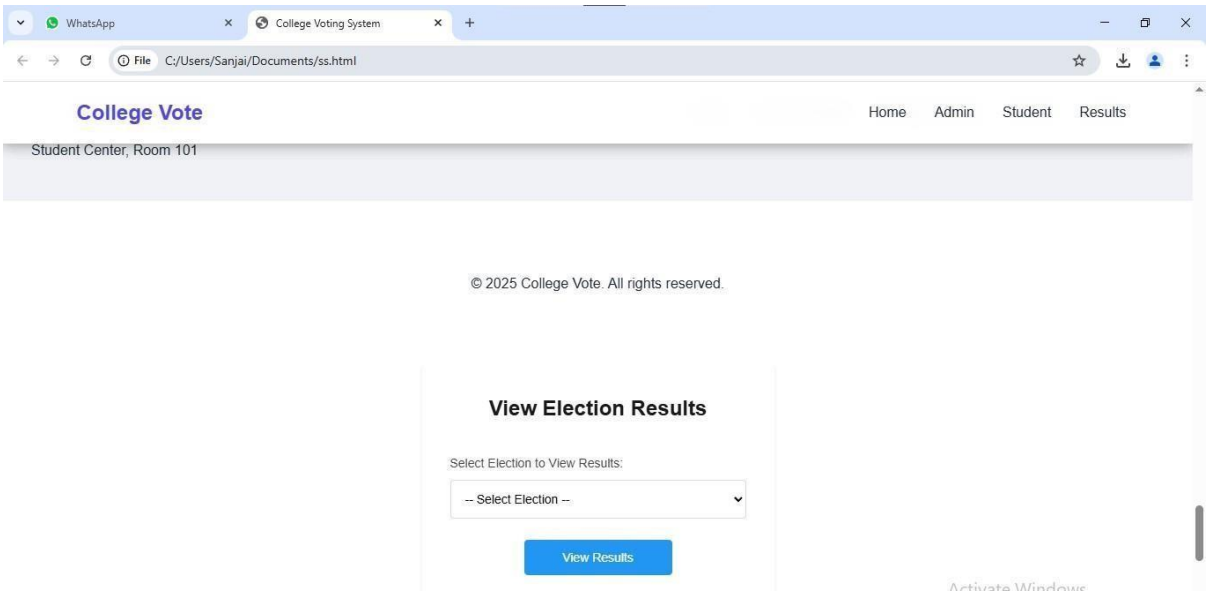
Candidate registration



Voting Process



Result


WhatsApp
College Voting System
File C:/Users/Sanjai/Documents/ss.html
College Vote
Home
Admin
Student
Results
Student Center, Room 101
© 2025 College Vote. All rights reserved.
View Election Results
Select Election to View Results:
-- Select Election --
View Results

8.2 SOURCE CODE

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>College Voting System</title>

    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css"
rel="stylesheet">

    <style>

        @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display
=swap');



        * { margin: 0; padding: 0; box-

            sizing: borderbox;  font-family:

            'Poppins', sans-serif;

        }



        :root {

            --primary: #4F46E5;

            --primary-dark: #4338CA;

            --secondary: #EC4899;

            --dark: #1F2937;

            --light: #F9FAFB;

        }
```

```
body {
    line-height:  1.6;
    color: var(--dark);
    overflow-x: hidden;
}
.navbar { background: rgba(255, 255, 255,
    0.95); padding: 1rem;  position: fixed;
    width: 100%;  top: 0; z-index: 1000;
    box-shadow: 0 2px 15px rgba(0,0,0,0.1);
    backdrop-filter: blur(10px);
}
.nav-content   {   max-width:   1200px;
    margin: 0 auto;
    display:  flex;  justify-content:  space-between;
            align-items:
    center;
}
.logo   {   color:   var(-primary);
    font-size:
    1.5rem; font-weight:
    700; display: flex;
    align-items:  center;
    gap: 0.5rem;
}
```

```
.nav-links { display:
   flex;            gap:
   2rem;  }    .nav-
   links  a  {  color:
   var(--dark);
         textdecorati
   on:   none;
   font-weight:              500;
   transition: all 0.3s; position:
   relative;
}
.nav-links a::after {
   content:          ";
   position: absolute;
         width: 0;
   height: 2px;
   bottom: -5px;
   left:  0;   background:  var(--primary);
   transition: width 0.3s;
}
.nav-links a:hover::after {  width:
   100%;

hamburger {  display:
   none; flexdirection:
         column;
```

```
        gap: 5px;  cursor:
        pointer;
    }
    .hamburger div { width: 25px; height:
        3px; background: var(--primary);
            transition:
        all 0.3s;
    }
    .hero { background: linear-gradient(135deg, rgba(79, 70, 229, 0.9), rgba(236, 72, 153,
        0.9)),  url('/api/placeholder/1920/1080');
background-size:    cover;
        background-position: center;
        min-height: 100vh;  display:
        flex;  alignitems:  center;
        justifycontent:  center;
        textalign:  center;  color:
        white; padding: 6rem
        1rem;  position:
        relative;  overflow:
        hidden;
    }
    .hero::before  {
        content:  ";
        position: absolute;
            width:
        100%;  height:
        100%;
        background: url("data:image/svg+xml,%3Csvg width='60' height='60' viewBox='0 0  60
60' xmlns='http://www.w3.org/2000/svg'%3E%3Cg fill='none' fill- rule='evenodd'%3E%3Cg
fill='%23ffffff' fill-opacity='0.1'%3E%3Cpath d='M36 34v-4h- 2v4h-4v2h4v4h2v-4h4v-2h-
```

```
4zm0-30V0h-2v4h-4v2h4v4h2V6h4V4h-4zM6 34v-
4H4v4H0v2h4v4h2v-4h4v-2H6zM6
4V0H4v4H0v2h4v4h2V6h4V4H6z'/%3E%3C/g%3E%3C/g%3E%3C/svg%3E");
    opacity: 0.1;
}

.hero-content { max-
    width:      800px;
    position: relative;  animation:
    fadeIn 1s ease-out;
}
@keyframes fadeIn { from { opacity: 0; transform:
    translateY(20px); }  to { opacity: 1; transform:
    translateY(0); }
}

.hero h1 { font-size:
    3.5rem; marginbottom:
        1rem;
    font-weight: 700; line-height:
    1.2;
}

.hero p { font-size:
    1.2rem;      margin-
    bottom: 2rem;
```

```
    opacity: 0.9;
}

.cta-button { background: white; color:
    var(--primary); padding: 1rem 2.5rem;
    border: none; border-radius: 50px;
    fontsize: 1.1rem; font-weight: 600;
    cursor: pointer; transition: all 0.3s;
    texttransform: uppercase; letter-spacing:
    1px; box-shadow: 0 4px 15px
    rgba(0,0,0,0.2);
}

.cta-button:hover { transform: translateY(-
    3px); box-shadow:
    0 6px 20px rgba(0,0,0,0.3);
}

.features { padding: 6rem 1rem;
    max-width:
    1200px; margin:
    0 auto;
}

.features h2 { text-align:
    center; margin-bottom:
```

```
            3rem; font-size:
                    2.5rem;
            color: var(--dark);
        }
            }
            .hero p { font-size:
                1rem;
            }
            .feature-card { padding:
                2rem;
            }
        }
    </style>
</head>
<body>
    <nav class="navbar">
        <div class="nav-content">
            <div class="logo">
                <i class="fas fa-vote-yea"></i>
                College Vote
            </div>
            <div class="hamburger">
                <div></div>
                <div></div>
                <div></div>
            </div>
```

```
        <div class="nav-links">
            <a href="index.php">Home</a>
            <a href="admin.php">Admin</a>
            <a href="login.php">Student</a>
            <a href="view_results.php">Results</a>
        </div>
    </div>
</nav>

<section class="hero" id="home">
    <div class="hero-content">
        <h1>Shape Your College's Future</h1>
        <p>Your vote is your voice. Join thousands of students in making decisions that matter. Participate in the democratic process and help build a better campus community.</p> <button class="cta-button">Start Voting Now</button>
    </div>
</section>

<section class="features">
    <h2>Why Choose Us</h2> <div class="feature-
    grid">
        <div class="feature-card">
            <i class="fas fa-shield-alt"></i>
            <h3>Secure Voting</h3>
            <p>State-of-the-art encryption ensures your vote remains confidential and tamper-proof.</p>
```

```
        </div>
        <div class="feature-card">
            <i class="fas fa-mobile-alt"></i>
            <h3>Vote Anywhere</h3>
            <p>Cast your vote from any device, anywhere on campus or at home.</p>
        </div>
        <div class="feature-card">
            <i class="fas fa-chart-line"></i>
            <h3>Live Results</h3>
            <p>Watch real-time election results and statistics as they unfold.</p>
        </div>
    </div>
</section>

<footer class="footer">
    <div class="footer-content">
        <div class="footer-section">
            <h3>About Us</h3>
            <p>College Vote is committed to fostering democratic participation and ensuring fair, transparent elections within our campus community.</p>
            <div class="social-links">
                <a href="#"><i class="fab fa-facebook"></i></a>
                <a href="#"><i class="fab fa-twitter"></i></a>
                <a href="#"><i class="fab fa-instagram"></i></a>
                <a href="#"><i class="fab fa-linkedin"></i></a>
            </div>
```

```
        </div>
        <div class="footer-section">
            <h3>Quick Links</h3>
            <ul class="footer-links">
                <li><a href="#">About Us</a></li>
                <li><a href="#">How It Works</a></li>
                <li><a href="#">Election Rules</a></li>
                <li><a href="#">Privacy Policy</a></li>
            </ul>
        </div>
        <div class="footer-section">
            <h3>Contact Us</h3>
            <ul class="footer-links">
                <li><i class="fas fa-envelope"></i> support@collegevote.edu</li>
                <li><i class="fas fa-phone"></i> (555) 123-4567</li>
                <li><i class="fas fa-map-marker-alt"></i> Student Center, Room 101</li>
            </ul>
        </div>
    </div>
    <div class="footer-bottom">
        <p>&copy; 2025 College Vote. All rights reserved.</p>
    </div>
</footer>

<script>  const  hamburger  =
```

```
document.querySelector('.hamburger');  const
navLinks = document.querySelector('.nav-links');

hamburger.addEventListener('click', () => {  navLinks.classList.toggle('active');
   hamburger.classList.toggle('active');

});

// Close menu when clicking a link
document.querySelectorAll('.nav-links a').forEach(link =>
{ link.addEventListener('click', () => {
navLinks.classList.remove('active'); hamburger.classList.remove('active');
   });
});

// Close menu when clicking outside
document.addEventListener('click', (e) => { if
(!hamburger.contains(e.target) && !navLinks.contains(e.target))
{  navLinks.classList.remove('active');
hamburger.classList.remove('active');
   }
});
```

```
    // Scroll reveal animation for feature cards  const featureCards =
    document.querySelectorAll('.feature-card');                    const
    observerOptions = { threshold: 0.2, rootMargin: '0px'
    };

    const observer = new IntersectionObserver((entries) => {
      entries.forEach(entry => { if (entry.isIntersecting) {
        entry.target.style.opacity          =          1;
        entry.target.style.transform = 'translateY(0)';
        }
      });
    }, observerOptions);

    featureCards.forEach(card => { card.style.opacity
      = 0;
      card.style.transform                    =
      'translateY(20px)'; card.style.transition =
      'all          0.6s          ease-out';
      observer.observe(card);
    });
  </script>
</body>
</html>


<?php
```

```
        include("dbconnect.php");
        extract($_POST);
        session_start();

if(isset($_POST['btn']))
{
$qry=mysqli_query($conn,"select * from admin where name='$uname'
&& psw='$password'"); $num=mysqli_num_rows($qry); if($num==1)
{
?>
<script>alert('welcome to admin home page');
</script>
<?php

header("location:add_election.php");
}
else
{     echo "<script>alert('User Name Password Wrong. ....
')</script>";
}
}
?>

<!DOCTYPE html>
```

```
<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width, initial-scale=1.0">
                                                <title>College Voting System</title>

   <link        href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css"
rel="stylesheet">

   <style>

      @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display
=swap');



      * { margin: 0; padding: 0; box-

         sizing: borderbox;  font-family:

         'Poppins', sans-serif;

      }



      :root {

         --primary: #4F46E5;

         --primary-dark: #4338CA;

         --secondary: #EC4899;

         --dark: #1F2937;

         --light: #F9FAFB;

      }



      body {  line-height:

                  1.6; color:
```

```
        var(--dark);
        overflow-x: hidden;
    }

    .navbar { background: rgba(255, 255, 255,
        0.95); padding: 1rem; position:
        fixed;
        width: 100%; top:
        0; z-index:
        1000;
    }
  </style>
</head>
<body>
  <nav class="navbar">
    <div class="nav-content">
      <div class="logo">
        <i class="fas fa-vote-yea"></i>
        College Vote
      </div>
      <div class="hamburger">
        <div></div>
        <div></div>
        <div></div>
      </div>
      <div class="nav-links">
```

```
          <a href="index.php">Home</a>
            <a href="admin.php">Admin</a>
            <a href="login.php">Student</a>
            <a href="view_results.php">Results</a>
          </div>
        </div>
    </nav>
<br /><br /><br /><br />

 <div style="display: flex; justify-content: center; align-items: center; min-height: 100vh; margin: 0; background-color: #f0f2f5; font-family: Arial, sans-serif;">
 <form method="post" style="background-color: white; padding: 2rem; border-radius: 8px; box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1); width: 320px;">
    <div style="text-align: center; margin-bottom: 2rem;">
      <h1 style="color: #1a1a1a; font-size: 24px; margin: 0;">Admin Login</h1>
    </div>

    <div style="margin-bottom: 1.5rem;">
      <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Username</label>
      <input type="text" id="uname" name="uname"
        style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;"
        onFocus="this.style.borderColor='#2196F3'"
        onBlur="this.style.borderColor='#ddd'">
    </div>
```

```
    <div style="margin-bottom: 2rem;">
      <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Password</label>
      <input type="password" id="password" name="password"
        style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;"
        onFocus="this.style.borderColor='#2196F3'"
        onBlur="this.style.borderColor='#ddd'">
    </div>

    <div style="display: flex; gap: 1rem;">
      <button type="submit" name="btn" id="btn"
        style="flex: 1; padding: 0.75rem; background-color: #2196F3; color: white; border: none; border-radius: 4px; cursor: pointer; font-size: 14px;">
        Login
      </button>

      <button type="reset"
        style="flex: 1; padding: 0.75rem; background-color: #f5f5f5; color: #666; border: none; border-radius: 4px; cursor: pointer; font-size: 14px;">
        Cancel
      </button>
    </div>
  </form>
</div>

<br /><br /><br /><br />
```

```
<footer class="footer">
    <div class="footer-content">
        <div class="footer-section">
            <h3>About Us</h3>
                <p>College Vote is committed to fostering democratic participation and ensuring fair, transparent elections within our campus community.</p>
            <div class="social-links">
                <a href="#"><i class="fab fa-facebook"></i></a>
                <a href="#"><i class="fab fa-twitter"></i></a>
                <a href="#"><i class="fab fa-instagram"></i></a>
                <a href="#"><i class="fab fa-linkedin"></i></a>
            </div>
        </div>
        <div class="footer-section">
            <h3>Quick Links</h3>
            <ul class="footer-links">
                <li><a href="#">About Us</a></li>
                <li><a href="#">How It Works</a></li>
                <li><a href="#">Election Rules</a></li>
                <li><a href="#">Privacy Policy</a></li>
            </ul>
        </div>
        <div class="footer-section">
            <h3>Contact Us</h3>
            <ul class="footer-links">
                <li><i class="fas fa-envelope"></i> support@collegevote.edu</li>
```

```
            <li><i class="fas fa-phone"></i> (555) 123-4567</li>
            <li><i class="fas fa-map-marker-alt"></i> Student Center, Room 101</li>
          </ul>
        </div>
      </div>
      <div class="footer-bottom">
        <p>&copy; 2025 College Vote. All rights reserved.</p>
      </div>
    </footer>

    <script> const hamburger = document.querySelector('.hamburger'); const navLinks = document.querySelector('.nav-links');

      hamburger.addEventListener('click', () => { navLinks.classList.toggle('active'); hamburger.classList.toggle('active');

      });

      // Close menu when clicking a link document.querySelectorAll('.nav-links a').forEach(link => { link.addEventListener('click', () => { navLinks.classList.remove('active'); hamburger.classList.remove('active');

        });
      });
```

```
    // Close menu when clicking outside
    document.addEventListener('click', (e) => { if
    (!hamburger.contains(e.target) && !navLinks.contains(e.target))
    { navLinks.classList.remove('active');
    hamburger.classList.remove('active');
      }
    });


  </script>
</body>
</html>


<?php
$conn = mysqli_connect("localhost", "root", "", "college_voting");
if (!$conn) { die("Connection failed: " . mysqli_connect_error());
}
?> <?php
include("dbconnect.php");
session_start();                    if
(isset($_POST['add_candidate'])) {
  $name = $_POST['name'];
  $position = $_POST['position'];
  $party = $_POST['party'];
```

```
    $experience = $_POST['experience'];
    $bio = $_POST['bio'];
    $manifesto = $_POST['manifesto'];
    $election_id = $_POST['election_id'];

    $photo = $_FILES['photo']['name'];
    $target = "upload/" . basename($photo);

    $query = "INSERT INTO candidates (name, position, party, experience, bio, manifesto, photo, election_id)
              VALUES ('$name', '$position', '$party', '$experience', '$bio', '$manifesto', '$photo', '$election_id')";

    if (mysqli_query($conn, $query)) { if
        (move_uploaded_file($_FILES['photo']['tmp_name'], $target))
        { echo "<script>alert('Candidate added successfully');</script>"; }
    } else {
        echo "<script>alert('Error adding candidate');</script>";
    }
}

// Fetch elections for the dropdown
$elections = mysqli_query($conn, "SELECT id, name FROM
elections"); ?>
<!DOCTYPE html>
<html lang="en">
<head>
```

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>College Voting System</title>
<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css" rel="stylesheet">
<style>
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display=swap');

* {
.social-links { display:
flex; gap: 1rem;
margin-top:
1rem;
}

.social-links a { color:
white; text-
decoration: none; font-size:
1.5rem;
opacity: 0.8;
transition: all 0.3s;
}
.social-links a:hover { opacity:
1; transform: translateY(-
3px);
```

```
}

.footer-bottom { text-align: center;
  margin-top: 3rem; padding-top: 2rem;
  border-top: 1px solid
  rgba(255,255,255,0.1);
}

@media (max-width: 768px) {
  .hamburger {
    display: flex;
  }

  .nav-links { display:
    none; position:
    absolute; top:
    100%;
    left: 0; right:
    0;
    background: white;
    flex-direction:
    column; padding:
    1rem; gap:
    1rem;
    text-align:
```

```
            center;
            box-
            shadow: 0
            4px
            15px rgba(0,0,0,0.1);
        }
        .nav-links.active { display: flex;
        }
        .hero h1 { font-size:
            2.5rem;
        }
        .hero p { font-size:
            1rem;
        }
        .feature-card { padding:
            2rem;
        }
    }
</style>
```

```
</head>

<body>

  <nav class="navbar">

    <div class="nav-content">

      <div class="logo">

        <i class="fas fa-vote-yea"></i>

        College Vote

      </div>

      <div class="hamburger">

        <div></div>

        <div></div>

        <div></div>

      </div>

      <div class="nav-links">

        <a href="add_election.php">Home</a>

        <a href="add_candidate.php">Add Candidate</a>

        <a href="manage_students.php">Manage Student</a>

        <a href="admin_announce_result.php">Results</a>

                                <a href="index.php">Logout</a>

      </div>

    </div>

  </nav>



<br /><br /><br /><br />

<div style="display: flex; flex-direction: column; align-items: center; min-height: 100vh; margin: 0; background-color: #f0f2f5; font-family: Arial, sans-serif; padding: 2rem;">
```

```
<!-- Add Candidate Form -->

<form method="POST" enctype="multipart/form-data" style="background-color: white; padding: 2rem; border-radius: 8px; box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1); width: 400px; margin-bottom: 2rem;">

  <div style="text-align: center; margin-bottom: 2rem;">

   <h1 style="color: #1a1a1a; font-size: 24px; margin: 0;">Add Candidate</h1> </div>


  <div style="margin-bottom: 1.5rem;">

   <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Candidate Name:</label>

   <input type="text" name="name" required

    style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">

  </div>



  <div style="margin-bottom: 1.5rem;">

   <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Position:</label>

   <input type="text" name="position" required

    style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">

  </div>



  <div style="margin-bottom: 1.5rem;">

   <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Party Name:</label>

   <input type="text" name="party" required

    style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">
```

```
  </div>

  <div style="margin-bottom: 1.5rem;">
    <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Experience (in years):</label>
    <input type="number" name="experience" min="0" required
      style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">
  </div>

  <div style="margin-bottom: 1.5rem;">
    <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Bio (Short Description):</label>
    <textarea name="bio" rows="3" required
      style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;"></textarea>
  </div>

  <div style="margin-bottom: 1.5rem;">
    <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Manifesto:</label>
    <textarea name="manifesto" rows="5" required
      style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;"></textarea>
  </div>

  <div style="margin-bottom: 1.5rem;">
```

```
    <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Photo:</label>

    <input type="file" name="photo" required

      style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">

   </div>


   <div style="margin-bottom: 1.5rem;">

    <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Election:</label>

    <select name="election_id" required

      style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">

      <option value="">-- Select Election --</option>

      <?php while ($row = mysqli_fetch_assoc($elections)) { ?>

        <option value="<?php echo $row['id']; ?>"><?php echo $row['name']; ?></option>

      <?php } ?>

    </select>

   </div>


   <div style="display: flex; justify-content: center;">

    <button type="submit" name="add_candidate"

      style="width: 50%; padding: 0.75rem; background-color: #2196F3; color: white; border: none; border-radius: 4px; cursor: pointer; font-size: 14px;">

      Add Candidate

    </button>

   </div>
```

```
  </form>
  <div>
<br /><br /><br /><br />
    <footer class="footer">
        <div class="footer-content">
            <div class="footer-section">
                <h3>About Us</h3>
                <p>College Vote is committed to fostering democratic participation and ensuring fair, transparent elections within our campus community.</p>
                <div class="social-links">
                    <a href="#"><i class="fab fa-facebook"></i></a>
                    <a href="#"><i class="fab fa-twitter"></i></a>
                    <a href="#"><i class="fab fa-instagram"></i></a>
                    <a href="#"><i class="fab fa-linkedin"></i></a>
                </div>
            </div>
            <div class="footer-section">
                <h3>Quick Links</h3>
                <ul class="footer-links">
                    <li><a href="#">About Us</a></li>
                    <li><a href="#">How It Works</a></li>
                    <li><a href="#">Election Rules</a></li>
                    <li><a href="#">Privacy Policy</a></li>
                </ul>
            </div>
            <div class="footer-section">
```

```
            <h3>Contact Us</h3>
            <ul class="footer-links">
                <li><i class="fas fa-envelope"></i> support@collegevote.edu</li>
                <li><i class="fas fa-phone"></i> (555) 123-4567</li>
                <li><i class="fas fa-map-marker-alt"></i> Student Center, Room 101</li>
            </ul>
        </div>
    </div>
    <div class="footer-bottom">
        <p>&copy; 2025 College Vote. All rights reserved.</p>
    </div>
</footer>

<script>
    const hamburger = document.querySelector('.hamburger');
    const navLinks = document.querySelector('.nav-links');

    hamburger.addEventListener('click', () => {
        navLinks.classList.toggle('active');
        hamburger.classList.toggle('active');

    });

    // Close menu when clicking a link
    document.querySelectorAll('.nav-links a').forEach(link =>
```

```
{ link.addEventListener('click', () => { navLinks.classList.remove('active');
hamburger.classList.remove('active');
  });
});

// Close menu when clicking outside
document.addEventListener('click', (e) => { if
(!hamburger.contains(e.target) && !navLinks.contains(e.target))
    { navLinks.classList.remove('active');
    hamburger.classList.remove('active');
  }
});


  </script>
</body> </html> <?php
include("dbconnect.php")
;
session_start();
$student_id = $_SESSION['id'];
// Fetch candidates for active elections (where results have not been announced) $result
= mysqli_query($conn, "
  SELECT candidates.id, candidates.name, candidates.position, candidates.party,
candidates.experience, candidates.bio, candidates.manifesto, candidates.photo, elections.id
AS election_id, elections.name AS election_name
  FROM candidates
```

```
    INNER JOIN elections
    ON candidates.election_id = elections.id
    WHERE CURDATE() BETWEEN elections.start_date AND elections.end_date AND
    elections.results_announced = 0
");

// Handle voting
if (isset($_POST['vote'])) {
    $candidate_id = $_POST['candidate_id'];
    $election_id = $_POST['election_id']; // Now it is defined

    // Check if the student has already voted in this election
    $checkVote = mysqli_query($conn, "SELECT * FROM votes WHERE
student_id='$student_id' AND election_id='$election_id'"); if
(mysqli_num_rows($checkVote) > 0) { echo "<script>alert('You have already
voted in this election');</script>";
    } else {
        $qry = mysqli_query($conn, "INSERT INTO votes (student_id, candidate_id,
election_id) VALUES ('$student_id', '$candidate_id', '$election_id')"); if ($qry)
{ echo "<script>alert('Vote cast successfully');</script>";
        } else { echo "<script>alert('Voting
            failed');</script>";
        }
    }
}
?>
```

```
            <h3>Contact Us</h3>
            <ul class="footer-links">
                <li><i class="fas fa-envelope"></i> support@collegevote.edu</li>
                <li><i class="fas fa-phone"></i> (555) 123-4567</li>
                <li><i class="fas fa-map-marker-alt"></i> Student Center, Room 101</li> </ul>
        </div>
    </div>
    <div class="footer-bottom">
        <p>&copy; 2025 College Vote. All rights reserved.</p>
    </div>
</footer>
<script> const hamburger =
    document.querySelector('.hamburger'); const navLinks =
    document.querySelector('.nav-links');
    hamburger.addEventListener('click', () =>
    { navLinks.classList.toggle('active');
        hamburger.classList.toggle('active');

    });

    // Close menu when clicking a link document.querySelectorAll('.nav-links
    a').forEach(link =>
        { link.addEventListener('click', () => {
        navLinks.classList.remove('active'); hamburger.classList.remove('active');
```

```
    });
});

// Close menu when clicking outside
document.addEventListener('click', (e) => { if
(!hamburger.contains(e.target) && !navLinks.contains(e.target))
{ navLinks.classList.remove('active');
hamburger.classList.remove('active');
    }
});

//JavaScript to dynamically show candidate details and set election ID
document.querySelector('select[name="candidate_id"]').addEventListener('change', function
() {
    var selectedOption = this.options[this.selectedIndex];
    document.getElementById('position').innerText = selectedOption.getAttribute('data-
position'); document.getElementById('party').innerText =
    selectedOption.getAttribute('data-party');
    document.getElementById('experience').innerText = selectedOption.getAttribute('data-
experience'); document.getElementById('bio').innerText =
    selectedOption.getAttribute('data-bio');
    document.getElementById('manifesto').innerText = selectedOption.getAttribute('data-
manifesto');
    document.getElementById('candidate-photo').src = "upload/" +
selectedOption.getAttribute('data-photo');

    // Set election_id in the hidden input
```

```
    document.getElementById('election_id').value = selectedOption.getAttribute('data-
election-id');
});

    </script>
</body>
</html>
<?php

include("dbconnect.php");
session_start();

// Check if user is admin
//if ($_SESSION['role'] !== 'admin') {
//     header("Location: login.php");
//     exit();
//}

// Fetch elections where results have been announced
$elections = mysqli_query($conn, "SELECT id, name FROM elections WHERE
results_announced = 1");

if (isset($_GET['election_id'])) {
    $election_id = $_GET['election_id'];

    // Fetch candidates and their vote counts for the selected election
```

```
    $result = mysqli_query($conn, "
        SELECT candidates.id, candidates.name, candidates.position, candidates.party, candidates.experience, candidates.bio, candidates.manifesto, candidates.photo,
        COALESCE(COUNT(votes.id), 0) AS vote_count
FROM candidates
LEFT JOIN votes ON candidates.id = votes.candidate_id AND votes.election_id = '$election_id'
WHERE candidates.election_id = '$election_id'
GROUP BY candidates.id
ORDER BY vote_count DESC

    ");

    // Fetch the total number of votes in the election to calculate the winner and loser
    $total_votes_result = mysqli_query($conn, "SELECT COUNT(id) AS total_votes FROM votes WHERE election_id = '$election_id'");
    $total_votes = mysqli_fetch_assoc($total_votes_result)['total_votes'];
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>College Voting System</title>
```

```
<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css" rel="stylesheet">
<style>
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display=swap');

* { margin: 0; padding: 0; box-sizing: borderbox;  font-family: 'Poppins', sans-serif;
}

:root {
--primary: #4F46E5;
--primary-dark: #4338CA;
--secondary: #EC4899;
--dark: #1F2937;
--light: #F9FAFB;
}
body {  line-height: 1.6; color: var(--dark);
overflow-x: hidden;
}
```

```
.navbar { background: rgba(255, 255,
  255, 0.95); padding: 1rem; position:
  fixed;
  width: 100%; top: 0; z-index: 1000;
  box-shadow: 0 2px 15px rgba(0,0,0,0.1);
  backdrop-filter: blur(10px);
}

.nav-content { max-width: 1200px;
  margin: 0 auto;
  display: flex; justify-content: space-between;
  align-items:
  center;
}

.logo { color: var(-primary);
  font-size:
  1.5rem; font-weight:
  700; display: flex;
  align-items: center;
  gap: 0.5rem;
}

.nav-links
 { display: flex;
gap: 2rem; }
```

```
.nav-links a { color: var(-
   -dark);
          textdecoration:
          none; font-
   weight:        500;
   transition:  all  0.3s;  position:
   relative;
}

.nav-links a::after {
   content:            ";
   position:
   absolute;    width:
   0;  height: 2px;
   bottom: -5px;
   left:  0;    background:  var(--primary);
   transition: width 0.3s;
}

.nav-links a:hover::after {  width:
   100%;
}

.hamburger   {   display:
   none;    flex-direction:
   column; gap: 5px;
```

```
    cursor: pointer;
}

.hamburger div { width: 25px; height:
    3px; background:
    var(--primary);
    transition: all 0.3s;
}

.hero {
    background: linear-gradient(135deg, rgba(79, 70, 229, 0.9), rgba(236, 72, 153, 0.9)),
url('/api/placeholder/1920/1080'); background-size:
        cover;
    background-position: center;
    min-height: 100vh;  display:
    flex;  alignitems:  center;
    justifycontent:  center;
    textalign:  center;  color:
    white; padding: 6rem
    1rem;  position: relative;
    overflow: hidden;
}
.hero::before  {  content:  ";
    position: absolute;  width:
    100%; height:
    100%;  background:  url("data:image/svg+xml,%3Csvg  width='60'  height='60'
    viewBox='0 0
```

```
60 60' xmlns='http://www.w3.org/2000/svg'%3E%3Cg fill='none' fill-
rule='evenodd'%3E%3Cg fill='%23ffffff' fill-opacity='0.1'%3E%3Cpath d='M36 34v-4h- 2v4h-
4v2h4v4h2v-4h4v-2h-4zm0-30V0h-2v4h-4v2h4v4h2V6h4V4h-4zM6 34v-
4H4v4H0v2h4v4h2v-4h4v-2H6zM6
4V0H4v4H0v2h4v4h2V6h4V4H6z'/%3E%3C/g%3E%3C/g%3E%3C/svg%3E");
    opacity: 0.1;
}

.hero-content { max-width:
  800px; position: relative;
  animation:
  fadeIn 1s ease-out;
}

@keyframes fadeIn { from { opacity: 0; transform:
  translateY(20px); } to { opacity: 1; transform:
  translateY(0); }
}

.hero h1 { font-size: 3.5rem;
      margin-
  bottom: 1rem;
  fontweight: 700;
      lineheight: 1.2;
}

.hero p { font-size:
  1.2rem;
```

```
        marginbottom:
        2rem; opacity:
    0.9;
}

.cta-button                {
    background:    white;
    color: var(--primary);
    padding: 1rem 2.5rem;
    border: none;
    borderradius:
        50px;  fontsize:
        1.1rem;
        fontweight: 600;
    cursor: pointer;
    transition: all 0.3s;
        text-transform:
    uppercase;
        letterspacing:
        1px;
        boxshadow: 0
    4px 15px
    rgba(0,0,0,0.2);
}
```

```
.cta-button:hover { transform: translateY(3px);
    box-shadow: 0 6px 20px rgba(0,0,0,0.3);
}

.features {
    padding: 6rem 1rem;
    max-width:  1200px;
    margin: 0 auto;
}

.features h2 { text-align:
    center;  margin-bottom:
    3rem; font-size:
            2.5rem;
    color: var(--dark);

}
.feature-grid  {  display:  grid;  grid-template-columns:
    repeat(auto-fit, minmax(300px, 1fr)); gap: 2rem;
}
.feature-card { text-align: center;
    padding:  2.5rem;  background:  white;  border-radius:
    20px; box-shadow: 0 10px
    30px rgba(0,0,0,0.1);
    transition: all 0.3s;
}
```

```
.feature-card:hover { transform: translateY(10px);
    box-shadow: 0 15px 40px rgba(0,0,0,0.15);
}

.feature-card i { font-size:
    2.5rem; color: var(--
    primary);          margin-
    bottom: 1.5rem;
}

.feature-card h3
    {  margin: 1rem 0;
    color:           var(--
    dark);    font-size:
    1.5rem;
}

.footer { background: var(-dark);
    color: white;
    padding: 4rem 1rem 2rem;
}
.footer-content { max-width: 1200px; margin: 0 auto; display: grid;
    grid-template-columns: repeat(auto-fit, minmax(250px,
    1fr)); gap: 3rem;
}
```

```
.footer-section  h3        { margin-bottom:
    1.5rem;         font-size: 1.2rem;
    color: var(-secondary);
}

.footer-links { list-style:
    none;
}
.footer-links li {  margin-bottom:
    0.8rem;
}

.footer-links  a  {  color:
    white;  text-decoration:
    none; opacity: 0.8;
    transition: opacity 0.3s;
}

.footer-links a:hover { opacity:
    1;
}

.social-links  {  display:
    flex;        gap:    1rem;
    margin-top:
```

```
    1rem;
}

.social-links a { color:
    white;              text-
    decoration:  none;  font-size:
    1.5rem;
    opacity: 0.8;

    transition: all 0.3s;
}

.social-links a:hover { opacity:
    1; transform: translateY(-
    3px);
}

.footer-bottom  {  text-align:  center;
    margin-top: 3rem; padding-top: 2rem;
    border-top:          1px          solid
    rgba(255,255,255,0.1);
}

@media (max-width: 768px) {
    .hamburger {
        display: flex;
    }
```

```
.nav-links { display:
    none;       position:
    absolute;   top:
    100%;
    left: 0; right:
    0;
    background:   white;      flex-direction:
    column; padding: 1rem; gap: 1rem; text-
    align: center; boxshadow: 0 4px 15px
    rgba(0,0,0,0.1);
}

.nav-links.active { display: flex;
}

.hero h1 { font-size:
    2.5rem;
}

.hero p { font-size:
    1rem;
}

.feature-card { padding:
    2rem;
```

```
        }
      }
    </style>
</head>
<body>
    <nav class="navbar">
        <div class="nav-content">
            <div class="logo">
                <i class="fas fa-vote-yea"></i>
                College Vote
            </div>
            <div class="hamburger">
                <div></div>
                <div></div>
                <div></div>
            </div>
            <div class="nav-links">
                <a href="index.php">Home</a>
                <a href="admin.php">Admin</a>
                <a href="login.php">Student</a>
                <a href="view_results">Results</a>
            </div>
        </div>
    </nav>
```

```
<br /><br /><br />

<form method="GET" style="background-color: white; padding: 2rem; border-radius: 8px; box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1); width: 400px; font-family: system-ui, -apple-system, sans-serif;margin:0 auto;">

  <div style="text-align: center; margin-bottom: 2rem;">

    <h1 style="color: #1a1a1a; font-size: 24px; margin: 0;">View Election Results</h1>

  </div>



  <div style="margin-bottom: 1.5rem;">

                <label style="display: block; margin-bottom: 0.5rem; color: #4a4a4a; font-size: 14px;">Select Election to View Results:</label>

    <select name="election_id" required style="width: 100%; padding: 0.75rem; border: 1px solid #ddd; border-radius: 4px; box-sizing: border-box; font-size: 14px;">

      <option value="">-- Select Election --</option>

      <?php while ($row = mysqli_fetch_assoc($elections)) { ?>

        <option value="<?php echo $row['id']; ?>"><?php echo $row['name']; ?></option>

      <?php } ?>

    </select>

  </div>



  <div style="display: flex; justify-content: center;">

    <button type="submit" style="width: 50%; padding: 0.75rem; background-color: #2196F3; color: white; border: none; border-radius: 4px; cursor: pointer; font-size: 14px;">

      View Results

    </button>

  </div>

</form>
```

```
<?php if (isset($result)) { ?>

  <div style="margin-top: 2rem; font-family: system-ui, -apple-system, sans-serif;">

    <h3 style="color: #1a1a1a; font-size: 20px; margin-bottom: 1rem;">Results for Election: <?php echo $_GET['election_id']; ?></h3>

    <div style="overflow-x: auto; background-color: white; border-radius: 8px; box-shadow: 0 1px 3px rgba(0, 0, 0, 0.1);">

      <table style="width: 100%; border-collapse: collapse; font-size: 14px;">

        <tr style="background-color: #f8f9fa;">

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Candidate Name</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Position</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Party</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Experience (Years)</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Bio</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Manifesto</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Photo</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Vote Count</th>

          <th style="padding: 12px 16px; text-align: left; font-weight: 600; color: #1a1a1a; border-bottom: 1px solid #e2e8f0;">Percentage</th>

        </tr>

        <?php

        $winner_vote_count = 0;

        $loser_vote_count = PHP_INT_MAX;

        $candidates = mysqli_fetch_all($result, MYSQLI_ASSOC);
```

```
    foreach ($candidates as $candidate) { if
    ($candidate['vote_count'] > $winner_vote_count) {
        $winner_vote_count = $candidate['vote_count'];
    } if ($candidate['vote_count'] < $loser_vote_count)
    {
        $loser_vote_count = $candidate['vote_count'];
    }
} foreach ($candidates as $candidate)
{
    $percentage = ($candidate['vote_count'] / $total_votes) * 100;
    $row_style = 'padding: 12px 16px; border-bottom: 1px solid #e2e8f0; color:
    #4a5568;';

    if ($candidate['vote_count'] == $winner_vote_count) {
        $row_style .= 'background-color: #d1fae5;';
    } elseif ($candidate['vote_count'] == $loser_vote_count) {
        $row_style .= 'background-color: #fee2e2;';
    }

    echo "<tr style='$row_style'>";  echo "<td style='padding: 12px 16px;'>" .
    htmlspecialchars($candidate['name']) .
"</td>";
    echo "<td style='padding: 12px 16px;'>" . htmlspecialchars($candidate['position']) .
"</td>";
    echo "<td style='padding: 12px 16px;'>" . htmlspecialchars($candidate['party']) .
"</td>";
```

```
        echo "<td style='padding: 12px 16px;'>" . htmlspecialchars($candidate['experience']) .
"</td>"; echo "<td style='padding: 12px 16px;'>" . htmlspecialchars($candidate['bio'])

        .

"</td>";

        echo "<td style='padding: 12px 16px;'>" . htmlspecialchars($candidate['manifesto']) .
"</td>";

        echo "<td style='padding: 12px 16px;'><img src='upload/" .
htmlspecialchars($candidate['photo']) . "' alt='Candidate Photo' style='width: 60px; height:
60px; object-fit: cover; border-radius: 4px;'></td>"; echo "<td style='padding: 12px

        16px;'>" . $candidate['vote_count'] . "</td>"; echo "<td style='padding: 12px

        16px;'>"  .  round($percentage,  2)  .  "%</td>";

        echo "</tr>";

      }

      ?>

    </table>

  </div>

 </div>

<?php } ?>

<br /><br /><br /><br />

  <footer class="footer">

    <div class="footer-content">

      <div class="footer-section">

        <h3>About Us</h3>

          <p>College Vote is committed to fostering democratic participation and ensuring
fair, transparent elections within our campus community.</p>

        <div class="social-links">

          <a href="#"><i class="fab fa-facebook"></i></a>
```

```
            <a href="#"><i class="fab fa-twitter"></i></a>
            <a href="#"><i class="fab fa-instagram"></i></a>
            <a href="#"><i class="fab fa-linkedin"></i></a>
        </div>
    </div>
    <div class="footer-section">
        <h3>Quick Links</h3>
        <ul class="footer-links">
            <li><a href="#">About Us</a></li>
            <li><a href="#">How It Works</a></li>
            <li><a href="#">Election Rules</a></li>
            <li><a href="#">Privacy Policy</a></li>
        </ul>
    </div>
    <div class="footer-section">
        <h3>Contact Us</h3>
        <ul class="footer-links">
            <li><i class="fas fa-envelope"></i> support@collegevote.edu</li>
            <li><i class="fas fa-phone"></i> (555) 123-4567</li>
            <li><i class="fas fa-map-marker-alt"></i> Student Center, Room 101</li>
        </ul>
    </div>
</div>
<div class="footer-bottom">
    <p>&copy; 2025 College Vote. All rights reserved.</p>
```

```
    </div>
  </footer>
  <script>  const  hamburger  =
    document.querySelector('.hamburger'); const navLinks = document.querySelector('.nav-
    links');

    hamburger.addEventListener('click', () =>
      { navLinks.classList.toggle('active');
      hamburger.classList.toggle('active');

    });
    // Close menu when clicking a link  document.querySelectorAll('.nav-links
    a').forEach(link =>
      {  link.addEventListener('click',  ()  =>  {
      navLinks.classList.remove('active'); hamburger.classList.remove('active');
      });
    });
    //  Close  menu  when  clicking  outside
    document.addEventListener('click',  (e)  =>  {  if
    (!hamburger.contains(e.target) && !navLinks.contains(e.target))
    { navLinks.classList.remove('active');
    hamburger.classList.remove('active');
      }
    });
  </script>
</body>
</html>
```

9. CONCLUSION

In conclusion, the college voting system offers a streamlined and efficient way for students to participate in the democratic process of electing their representatives. By implementing a secure and user-friendly platform, the system ensures transparency, accuracy, and ease of access for students to cast their votes. The use of technology reduces the chances of fraud or errors, and the system provides real-time results, ensuring that the process is smooth and timely. Overall, the college voting system fosters greater student engagement, empowers them to voice their opinions, and contributes to the development of a more organized and responsible student governance.

FUTURE ENHANCEMENT

The future scope of the college voting system is vast, with potential advancements in security, accessibility, and efficiency. Integration of blockchain technology can enhance transparency and prevent tampering, ensuring a more secure voting process. Biometric authentication, such as fingerprint or facial recognition, can further strengthen voter verification. Additionally, incorporating AI-driven data analysis can provide insights into voting trends and student engagement. Expanding the system to support multiple institutions through a centralized platform can make inter-college elections more accessible. With continuous technological improvements, the college voting system can evolve into a more robust, scalable, and foolproof mechanism, promoting fair and efficient elections in educational institutions.

10. REFERENCES

BOOK REFERENCES

1. Heinold, Brian. "A practical introduction to Python programming." (2021).
2. Kneusel, Ronald T. Practical deep learning: A Python-based introduction. No Starch Press, 2021.
3. Dhruv, Akshit J., Reema Patel, and Nishant Doshi. "Python: the most advanced programming language for computer science applications." Science and Technology Publications, Lda (2021): 292-299.
4. Sundnes, Joakim. Introduction to scientific programming with Python. Springer Nature, 2020.
5. Hill, Christian. Learning scientific programming with Python. Cambridge University Press, 2020.

WEBSITE REFERENCES

1. https://medium.com/javarevisited/10-free-python-tutorials-and-courses-from-google-microsoft-and-coursera-for-beginners-96b9ad20b4e6
2. https://www.bestcolleges.com/bootcamps/guides/learn-python-free/
3. https://www.programiz.com/python-programming
4. https://realpython.com/
5. https://www.codecademy.com/learn/learn-python