



NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name :Thamreen.M
Student ID :au820621104085

College Name

Arasu Engineering College

CAPSTONE PROJECT SHOWCASE

Project Title

Voting Application using Django Framework-Thamreen.M(4085,AEC)

Abstract | Problem Statement | Project Overview | Proposed Solution |
Technology Used | Modelling & Results | Conclusion



Abstract

The proposed voting application is a web-based platform that allows users to create and participate in online votes. The application is built using the Django framework, a popular and well-supported Python-based web framework that provides a robust foundation for building scalable and secure web applications . The application is also designed to be flexible and scalable, with a modular architecture that allows for easy customization and extension. This makes it suitable for a wide range of use cases, from small-scale internal votes to large-scale public elections . Overall, the proposed voting application is a secure, user-friendly, and flexible platform for conducting online votes. Its use of the Django framework ensures a robust and scalable foundation, while its focus on security and user experience makes it an ideal choice for a wide range of voting scenarios.

Problem Statement

Online voting has become increasingly popular in recent years, with a growing number of organizations and governments turning to digital platforms to conduct elections and polls. However, online voting also presents a number of challenges, particularly in terms of security and integrity . Overall, the proposed voting application will address the challenges of security and integrity in online voting, while also providing a user-friendly platform for conducting online votes. Its use of the Django framework will ensure a robust and scalable foundation, while its focus on security and user experience will make it an ideal choice for a wide range of voting scenarios. In addition to its focus on security, the application will also prioritize user experience, with a clean and intuitive interface that makes it easy for users to create and participate in votes. The application will support multiple types of votes, including single-choice and multiple-choice votes, and will allow users to set deadlines and restrictions for each vote.

Project Overview

The project overview for a voting application using the Django framework involves creating a secure and user-friendly online voting system. The application allows users to register, vote, and view real-time results. Here is a steps involved in building the voting application:

- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication to allow users to register, log in, and participate in voting.
- 4.Building the Voting Interface:** Develop the interface where users can view options, select their choices, and submit votes.
- 5.Implementing Real-time Results:** Display the voting results dynamically to provide instant feedback to users.
- 6.Developing an Admin Panel:** Build an admin panel to manage the voting process, candidates, and user accounts effectively.

Proposed Solution

The proposed solution for a voting application using the Django framework is to create a secure and user-friendly online voting platform. The application will allow users to register, vote, and view real-time results. To build the application, the Django framework will be used as the foundation due to its robustness and scalability. The application will have a user-friendly interface, a secure database, real-time results, and an admin panel for efficient management of elections, candidates, and user accounts.

In summary, the proposed solution for a voting application using the Django framework is a secure, user-friendly, and flexible platform for conducting online votes. Its use of the Django framework ensures a robust and scalable foundation, while its focus on security and user experience makes it an ideal choice for a wide range of voting scenarios.

Home Page

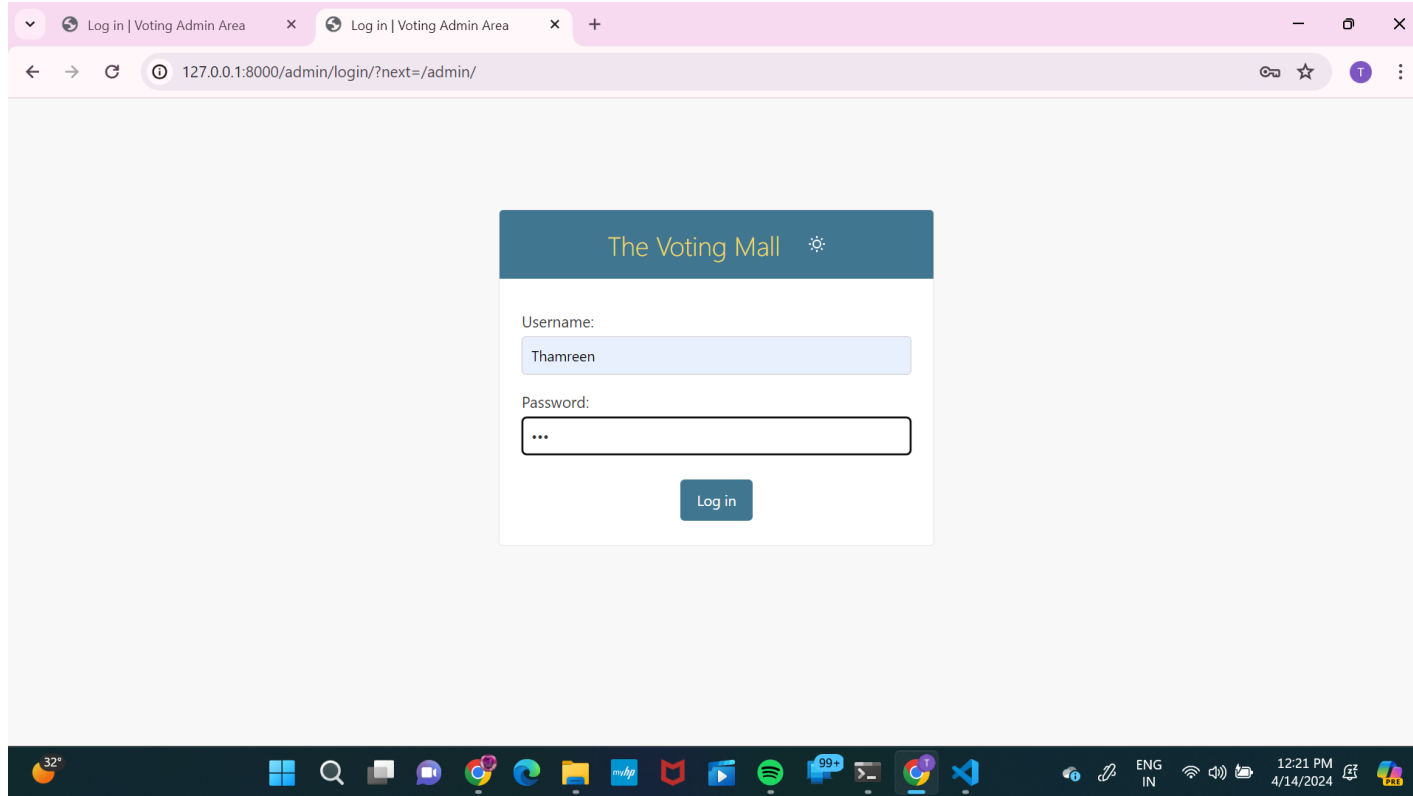
The Poll Mall

Welcome to Poll Mall!

This is my first Django Project after long time!

[View Available Polls!](#)

Admin Login Page



The screenshot shows a web browser window with two tabs, both titled "Log in | Voting Admin Area". The address bar displays the URL "127.0.0.1:8000/admin/login/?next=/admin/". The main content area features a login form titled "The Voting Mall" with a settings icon. The form includes a "Username:" label, a text input field containing "Thamreen", a "Password:" label, a password input field with masked characters "...", and a "Log in" button. The Windows taskbar at the bottom shows the system clock at 12:21 PM on 4/14/2024, along with various application icons and system status indicators like temperature (32°) and language (ENG IN).

Log in | Voting Admin Area

Log in | Voting Admin Area

127.0.0.1:8000/admin/login/?next=/admin/

The Voting Mall

Username:

Thamreen

Password:

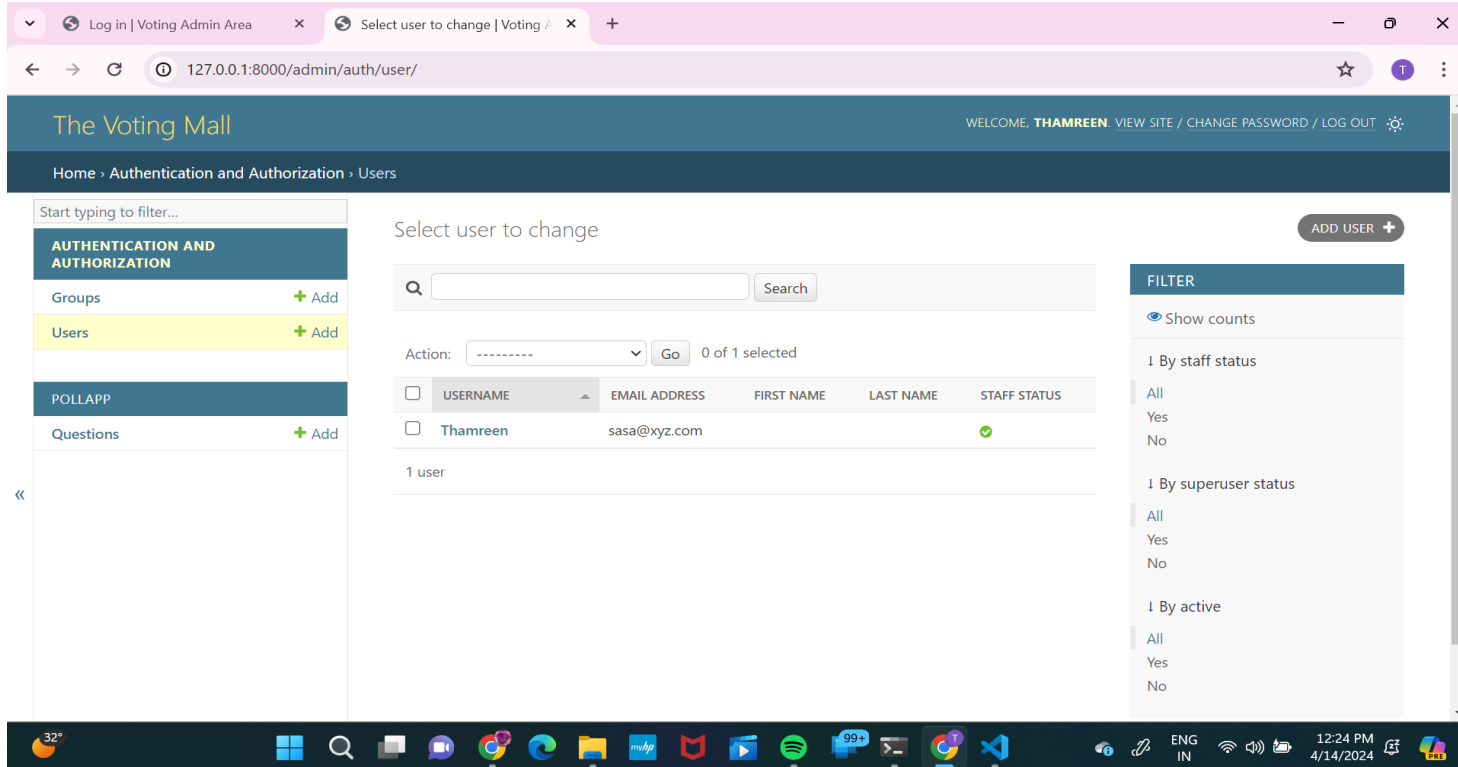
...

Log in

32°

12:21 PM
4/14/2024

Admin Details Page



The screenshot displays the 'Admin Details Page' within 'The Voting Mall' application. The browser's address bar shows the URL `127.0.0.1:8000/admin/auth/user/`. The page header includes a welcome message for 'THAMREEN' and links for 'VIEW SITE', 'CHANGE PASSWORD', and 'LOG OUT'.

The left sidebar contains a navigation menu with the following sections:

- AUTHENTICATION AND AUTHORIZATION**
 - Groups + Add
 - Users + Add**
- POLLAPP**
 - Questions + Add

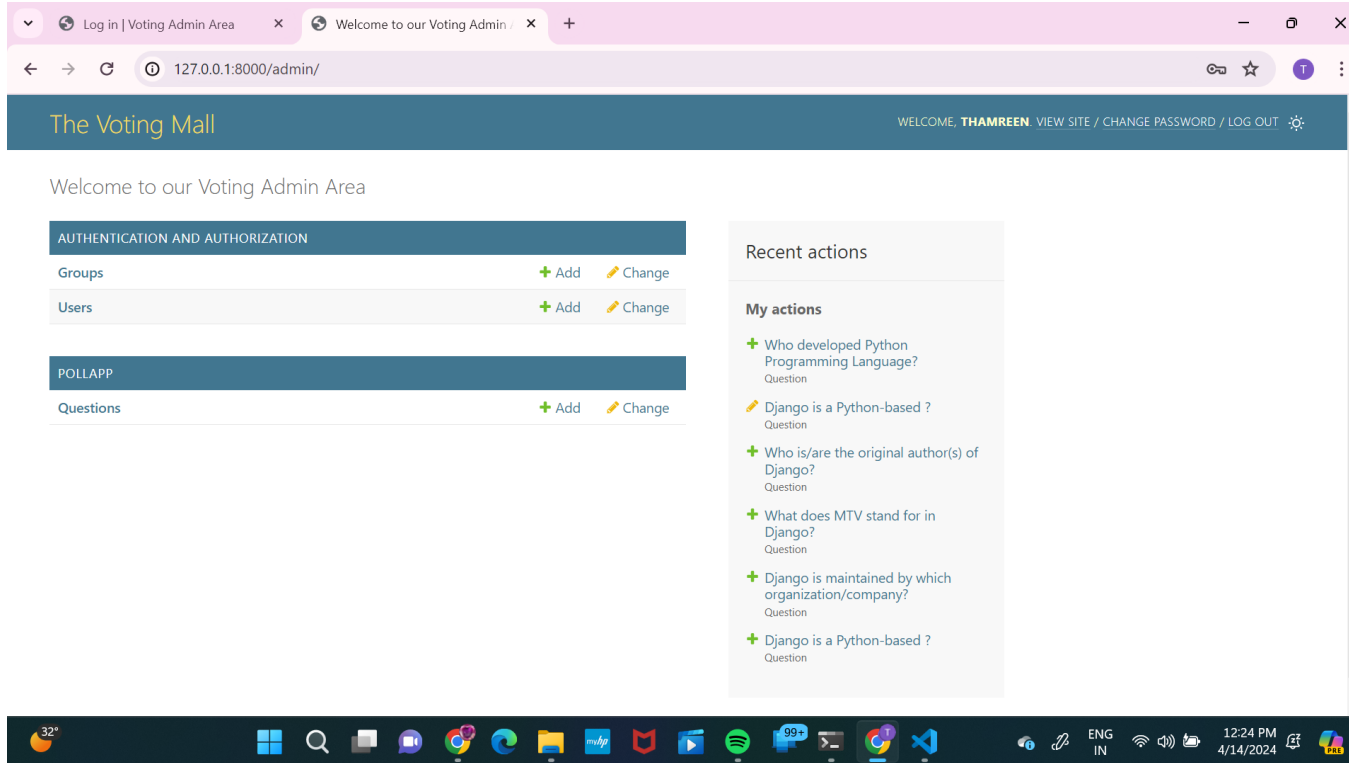
The main content area is titled 'Select user to change'. It features a search bar and a table of users. The table has columns for USERNAME, EMAIL ADDRESS, FIRST NAME, LAST NAME, and STAFF STATUS. One user, 'Thamreen', is listed with the email 'sasa@xyz.com' and a 'Yes' status.

Below the table, it indicates '1 user' is selected. On the right side of the main content area, there is a 'FILTER' panel with the following options:

- By staff status**
 - All
 - Yes
 - No
- By superuser status**
 - All
 - Yes
 - No
- By active**
 - All
 - Yes
 - No

The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 12:24 PM on 4/14/2024.

Admin Page



The screenshot displays the Django Admin interface for a project named 'The Voting Mall'. The browser's address bar shows the URL '127.0.0.1:8000/admin/'. The page header includes a welcome message for 'THAMREEN' and links for 'VIEW SITE', 'CHANGE PASSWORD', and 'LOG OUT'. The main content area is divided into two columns. The left column contains two sections: 'AUTHENTICATION AND AUTHORIZATION' with links for 'Groups' and 'Users' (each with '+ Add' and 'Change' options), and 'POLLAPP' with a link for 'Questions' (also with '+ Add' and 'Change' options). The right column features a 'Recent actions' section titled 'My actions', which lists several questions added recently, such as 'Who developed Python Programming Language?' and 'Django is a Python-based ?'. The Windows taskbar at the bottom shows the system clock as 12:24 PM on 4/14/2024.

Log in | Voting Admin Area x Welcome to our Voting Admin / x +

127.0.0.1:8000/admin/

The Voting Mall

WELCOME, **THAMREEN** [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#) ⚙

Welcome to our Voting Admin Area

AUTHENTICATION AND AUTHORIZATION

Groups	+ Add Change
Users	+ Add Change

POLLAPP

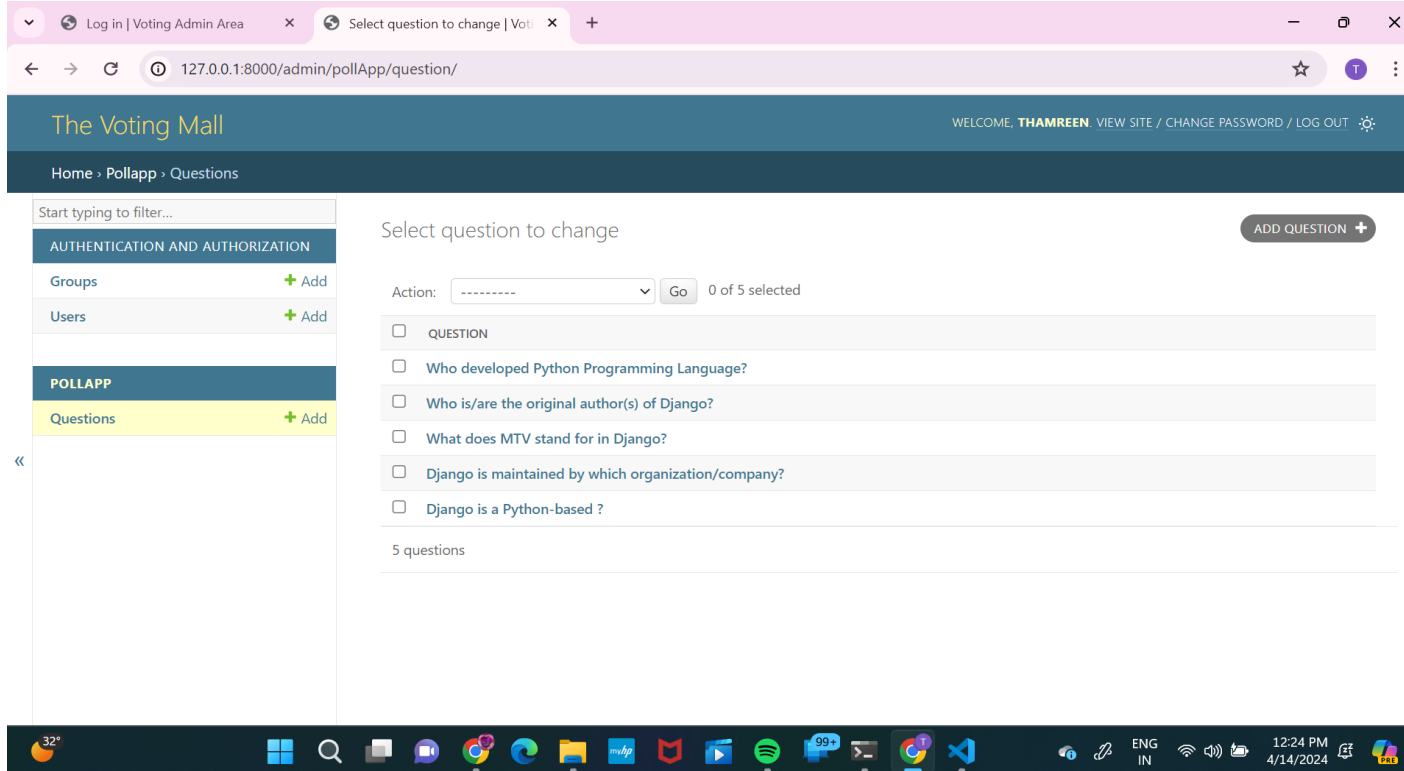
Questions	+ Add Change
-----------	--

Recent actions

My actions

- + Who developed Python Programming Language? Question
- + Django is a Python-based ? Question
- + Who is/are the original author(s) of Django? Question
- + What does MTV stand for in Django? Question
- + Django is maintained by which organization/company? Question
- + Django is a Python-based ? Question

Question Adding Page



The screenshot displays the 'Question Adding Page' within 'The Voting Mall' application. The browser's address bar shows the URL `127.0.0.1:8000/admin/pollApp/question/`. The application's header includes the title 'The Voting Mall' and a welcome message for 'THAMREEN'. A sidebar on the left contains a search bar and a menu with categories: 'AUTHENTICATION AND AUTHORIZATION' (Groups, Users) and 'POLLAPP' (Questions). The main content area is titled 'Select question to change' and features an 'ADD QUESTION +' button. Below this, there is an 'Action:' dropdown menu and a 'Go' button. A list of five questions is displayed, each with a checkbox: 'QUESTION', 'Who developed Python Programming Language?', 'Who is/are the original author(s) of Django?', 'What does MTV stand for in Django?', and 'Django is maintained by which organization/company?'. The bottom of the page shows a Windows taskbar with various application icons and system information.

Log in | Voting Admin Area | Select question to change | Voting Admin Area

127.0.0.1:8000/admin/pollApp/question/

The Voting Mall

WELCOME, THAMREEN. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Pollapp > Questions

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

POLLAPP

Questions + Add

Select question to change

ADD QUESTION +

Action: [dropdown] Go 0 of 5 selected

☐ QUESTION

☐ Who developed Python Programming Language?

☐ Who is/are the original author(s) of Django?

☐ What does MTV stand for in Django?

☐ Django is maintained by which organization/company?

☐ Django is a Python-based ?

5 questions

Voting Details Page

Log in | Voting Admin Area

Who developed Python Program

127.0.0.1:8000/admin/pollApp/question/5/change/

The Voting Mall

WELCOME, **THAMREEN** [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home > Pollapp > Questions > Who developed Python Programming Language?

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

POLLAPP

Questions [+ Add](#)

Change question

Who developed Python Programming Language?

HISTORY

Question text:

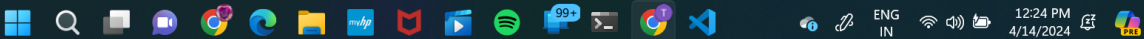
Who developed Python Programming Language?

Date Information (Show)

CHOICES

CHOICE TEXT	VOTES	DELETE?
<div>Wick van Rossum</div> <div>Wick van Rossum</div>	<div>9</div>	<input type="checkbox"/>
<div>Rasmus Lerdorf</div> <div>Rasmus Lerdorf</div>	<div>6</div>	<input type="checkbox"/>
<div>Guido van Rossum</div> <div>Guido van Rossum</div>	<div>31</div>	<input type="checkbox"/>

32°



12:24 PM
4/14/2024

Technology Used

Front-end



Back-end



Future Enhancements:

Future enhancements in a voting application using the Django framework, several key features and improvements can be considered based on the information from the provided sources,

1.Asynchronous Programming: Implementing asynchronous programming can enhance the performance of the application by allowing tasks to run concurrently, improving responsiveness and scalability.

2.Microservices Architecture: Adopting a microservices architecture can make the application more modular, easier to maintain, and scalable by breaking it into smaller, independent services that communicate with each other

3.Serverless Computing: Utilizing serverless computing can optimize resource utilization and reduce costs by enabling automatic scaling and only paying for actual usage, enhancing the application's efficiency and cost-effectiveness.

4.Client-Side Encryption: Enhancing security by implementing client-side encryption can protect sensitive data and ensure the confidentiality of votes, contributing to a more secure e-voting platform.

5.Blockchain Technology: Integrating blockchain technology can provide transparent and verifiable voting processes, ensuring the integrity of elections and promoting trust in the system

Conclusion

To create a voting application using Django, one should have a solid understanding of Python programming, Django framework, HTML, CSS, and Bootstrap. The development process involves creating a new Django project, creating a Django app, defining models, creating views, defining templates, and creating URLs. The application can be further enhanced with features such as real-time results, a user-friendly interface, and a secure database design. It can also include an admin panel for managing elections, candidates, and user accounts. Overall, a voting application using the Django framework is a powerful and flexible solution for creating online voting systems that can cater to various use cases and requirements.

Thank You!