



NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name :Nishaanth M
Student ID :autb21csl019

College Name

Arasu Engineering College

CAPSTONE PROJECT SHOWCASE

Project Title

Voting Application using Django Framework-Nishaanth(4311,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:

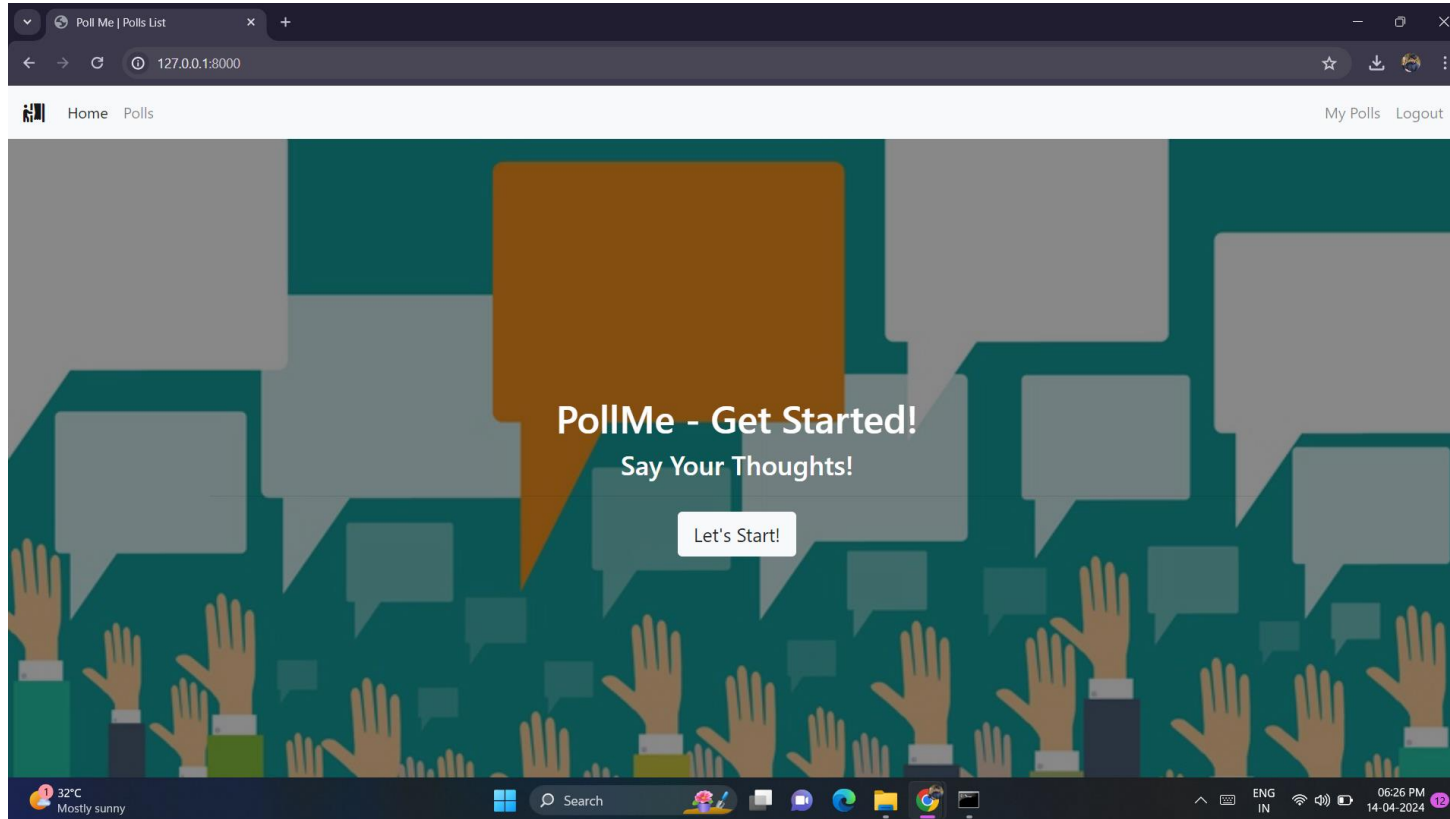
- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication 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



Poll Page



Welcome to polls List!

Sort by

Sort by

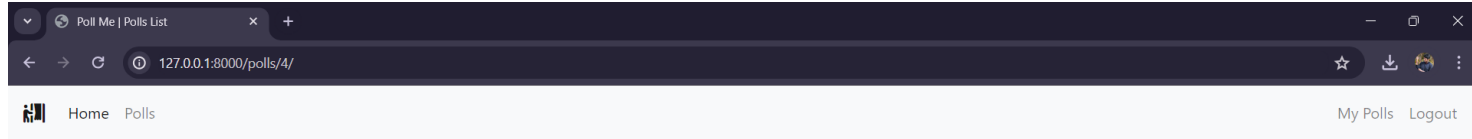
Sort by

Add +

Search

which food you like? ✓	Edit
what is your fav anime? ✓	Edit
which social media you use ... ✓	Edit
which framework is best? ✓	Edit

Voting Page



Polls details page

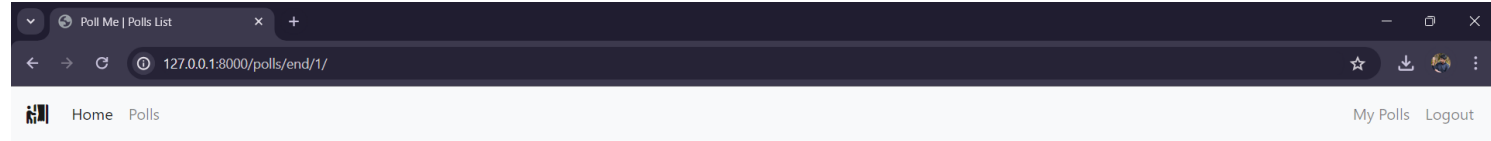
which framework is best?

- ☐ django
- ☐ flask

Vote

Cancel

Voting Details Page



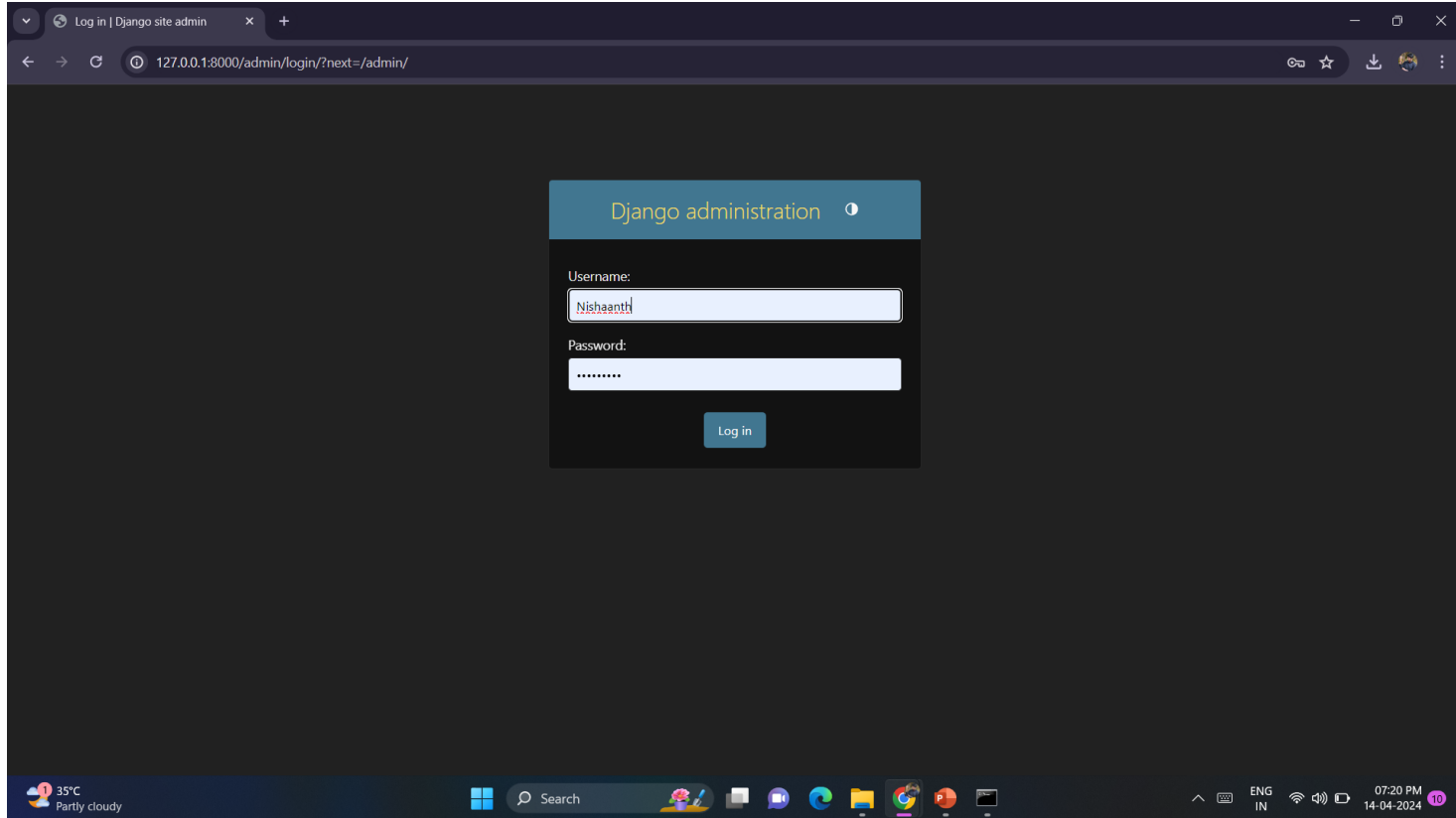
"which food you like?" Has Ended Polling!

Total: 1 votes

shawarma-100%	
shawarma	1
biryani	0

[Back To Polls](#)

Admin Login Page



The screenshot shows a web browser window with the title "Log in | Django site admin". The address bar displays the URL "127.0.0.1:8000/admin/login/?next=/admin/". The main content area features a dark background with a central light blue box titled "Django administration". Inside this box, there are two input fields: "Username:" with the text "Nishaanth" and "Password:" with masked characters. A "Log in" button is positioned below the password field. The browser's taskbar at the bottom shows the system clock as 07:20 PM on 14-04-2024, along with various system icons and a search bar.

Log in | Django site admin

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

Django administration

Username:

Nishaanth

Password:

Log in

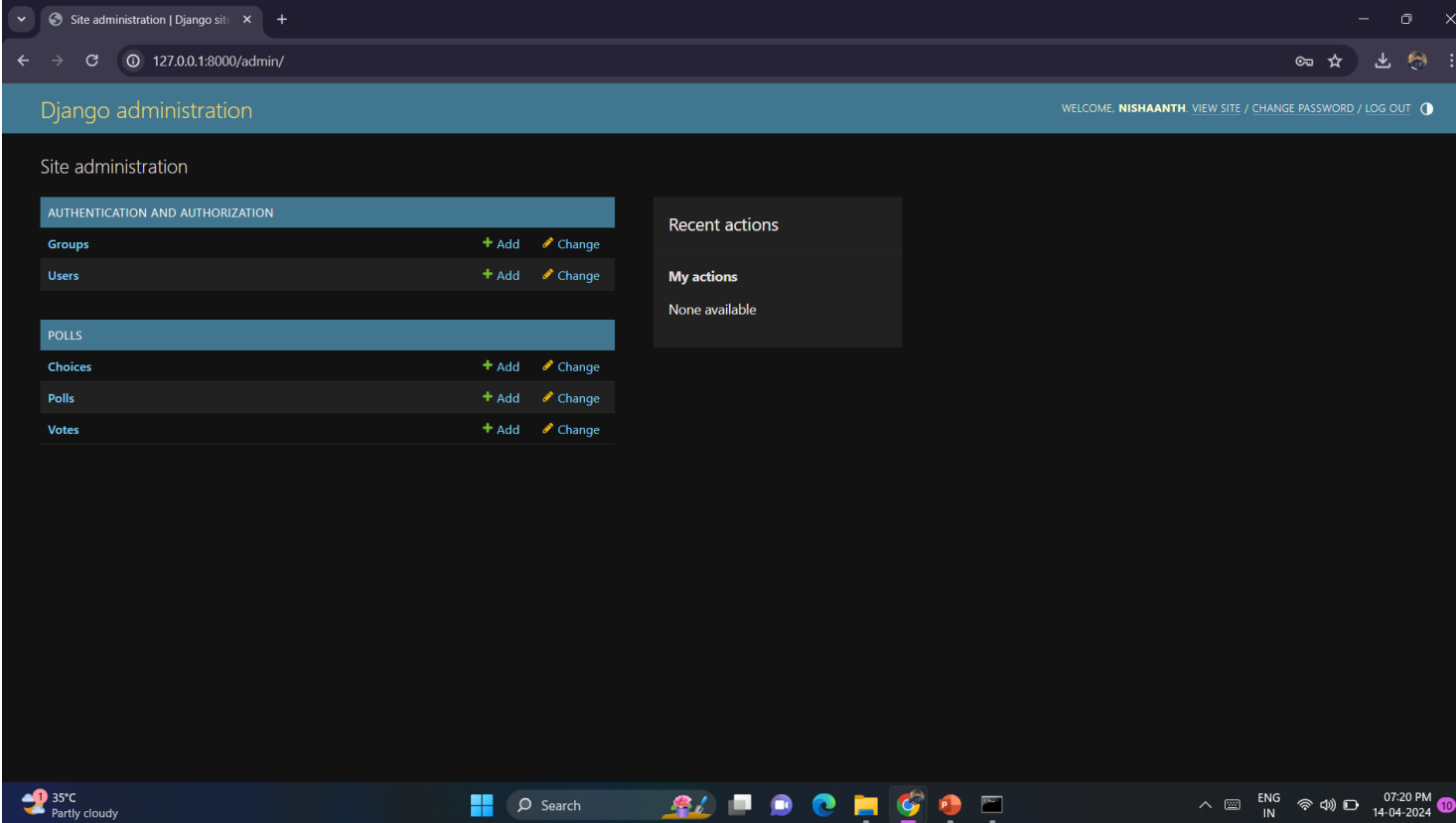
35°C Partly cloudy

Search

ENG IN

07:20 PM 14-04-2024

Admin Home Page



The screenshot displays the Django administration interface within a web browser. The browser's address bar shows the URL `127.0.0.1:8000/admin/`. The page header includes the title "Django administration" and a welcome message for "NISHAANTH" with links to "VIEW SITE", "CHANGE PASSWORD", and "LOG OUT".

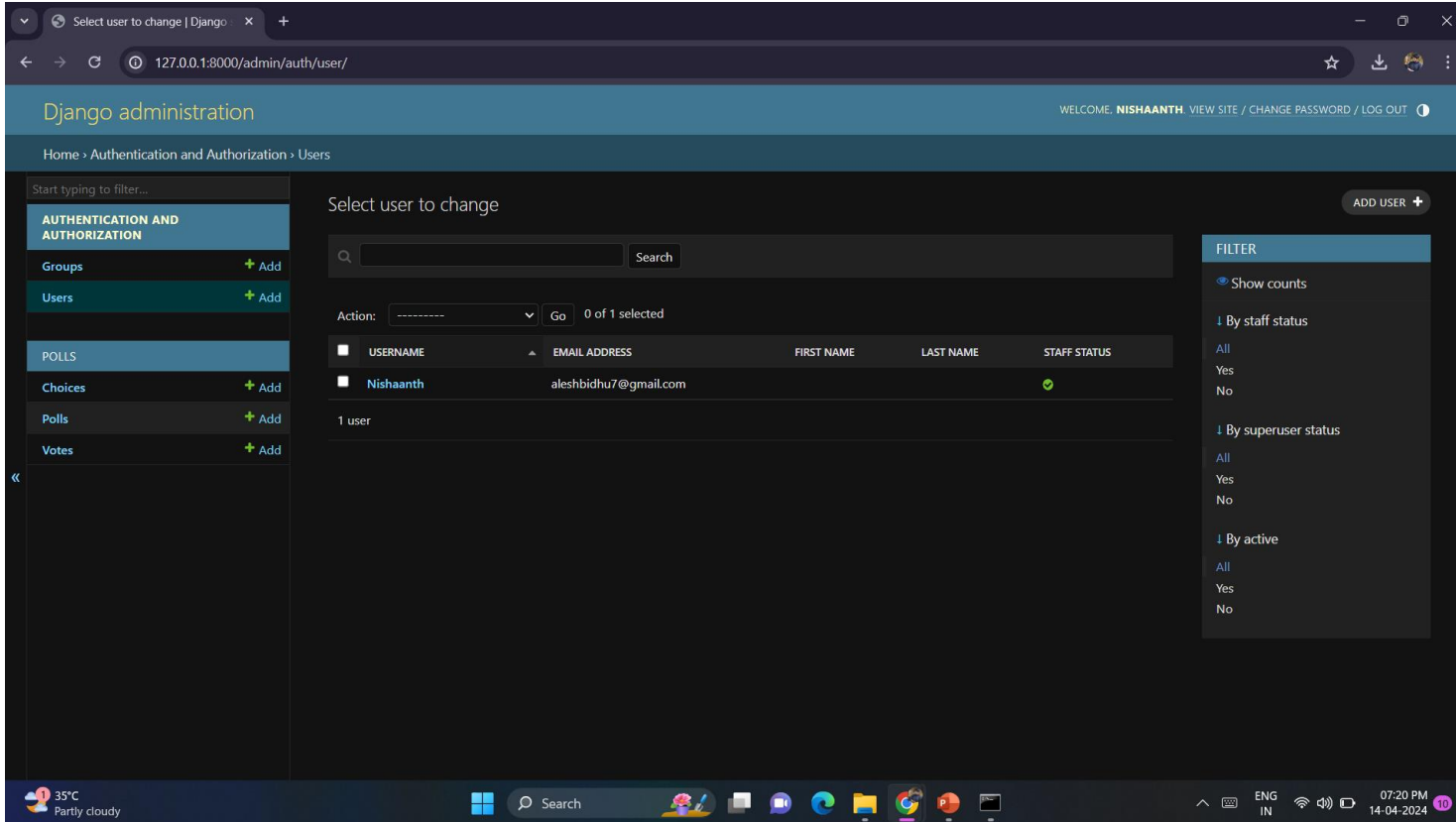
The main content area is titled "Site administration" and is divided into two sections:

- AUTHENTICATION AND AUTHORIZATION**: This section contains two rows of links. The first row, "Groups", has a green "+ Add" link and a yellow pencil "Change" link. The second row, "Users", also has a green "+ Add" link and a yellow pencil "Change" link.
- POLLS**: This section contains three rows of links. The first row, "Choices", has a green "+ Add" link and a yellow pencil "Change" link. The second row, "Polls", has a green "+ Add" link and a yellow pencil "Change" link. The third row, "Votes", has a green "+ Add" link and a yellow pencil "Change" link.

On the right side of the page, there is a "Recent actions" section with the heading "My actions" and the text "None available".

The Windows taskbar at the bottom shows the system clock as 07:20 PM on 14-04-2024, along with weather information (35°C, Partly cloudy) and various application icons.

Authentication and Authorization Page



Django administration

WELCOME, NISHAANTH VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Authentication and Authorization > Users

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups + Add
- Users + Add

POLLS

- Choices + Add
- Polls + Add
- Votes + Add

Select user to change

Search

Action: ----- Go 0 of 1 selected

USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
Nishaanth	aleshbidhu7@gmail.com			Yes

1 user

FILTER

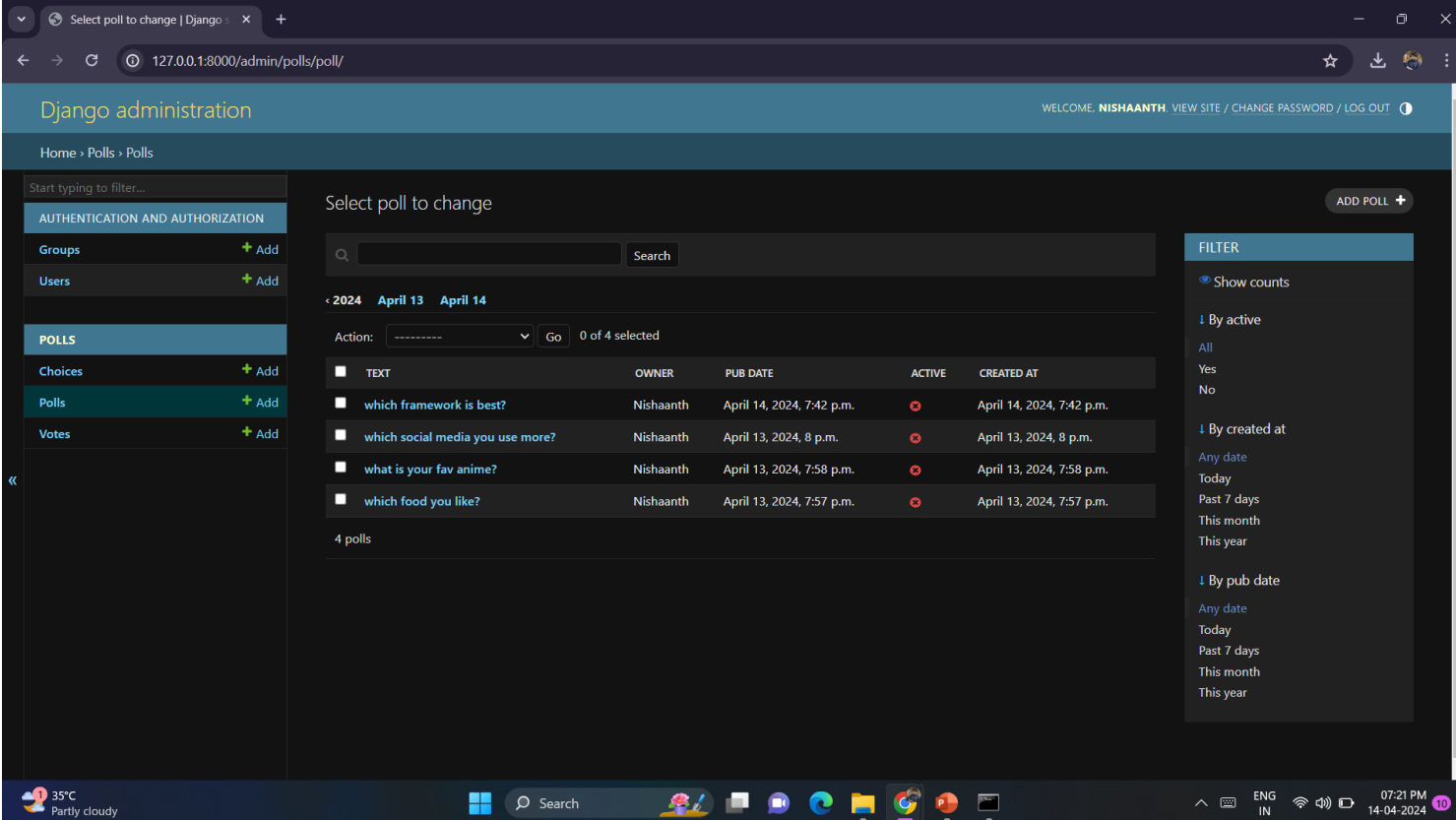
- Show counts
- By staff status
 - All
 - Yes
 - No
- By superuser status
 - All
 - Yes
 - No
- By active
 - All
 - Yes
 - No

35°C Partly cloudy

Search

ENG IN 07:20 PM 14-04-2024

Questions Adding Section Page



The screenshot shows the Django administration interface for the 'Polls' app. The browser address bar indicates the URL is 127.0.0.1:8000/admin/polls/poll/. The page title is 'Django administration'. The user is logged in as 'NISHAANTH' and can view the site, change password, or log out.

The left sidebar contains the following navigation links:

- Start typing to filter...
- AUTHENTICATION AND AUTHORIZATION
 - Groups + Add
 - Users + Add
- POLLS
 - Choices + Add
 - Polls + Add
 - Votes + Add

The main content area is titled 'Select poll to change'. It features a search bar and a table of polls. The table has columns for TEXT, OWNER, PUB DATE, ACTIVE, and CREATED AT. There are 4 polls listed.

TEXT	OWNER	PUB DATE	ACTIVE	CREATED AT
<input type="checkbox"/> which framework is best?	Nishaanth	April 14, 2024, 7:42 p.m.		April 14, 2024, 7:42 p.m.
<input type="checkbox"/> which social media you use more?	Nishaanth	April 13, 2024, 8 p.m.		April 13, 2024, 8 p.m.
<input type="checkbox"/> what is your fav anime?	Nishaanth	April 13, 2024, 7:58 p.m.		April 13, 2024, 7:58 p.m.
<input type="checkbox"/> which food you like?	Nishaanth	April 13, 2024, 7:57 p.m.		April 13, 2024, 7:57 p.m.

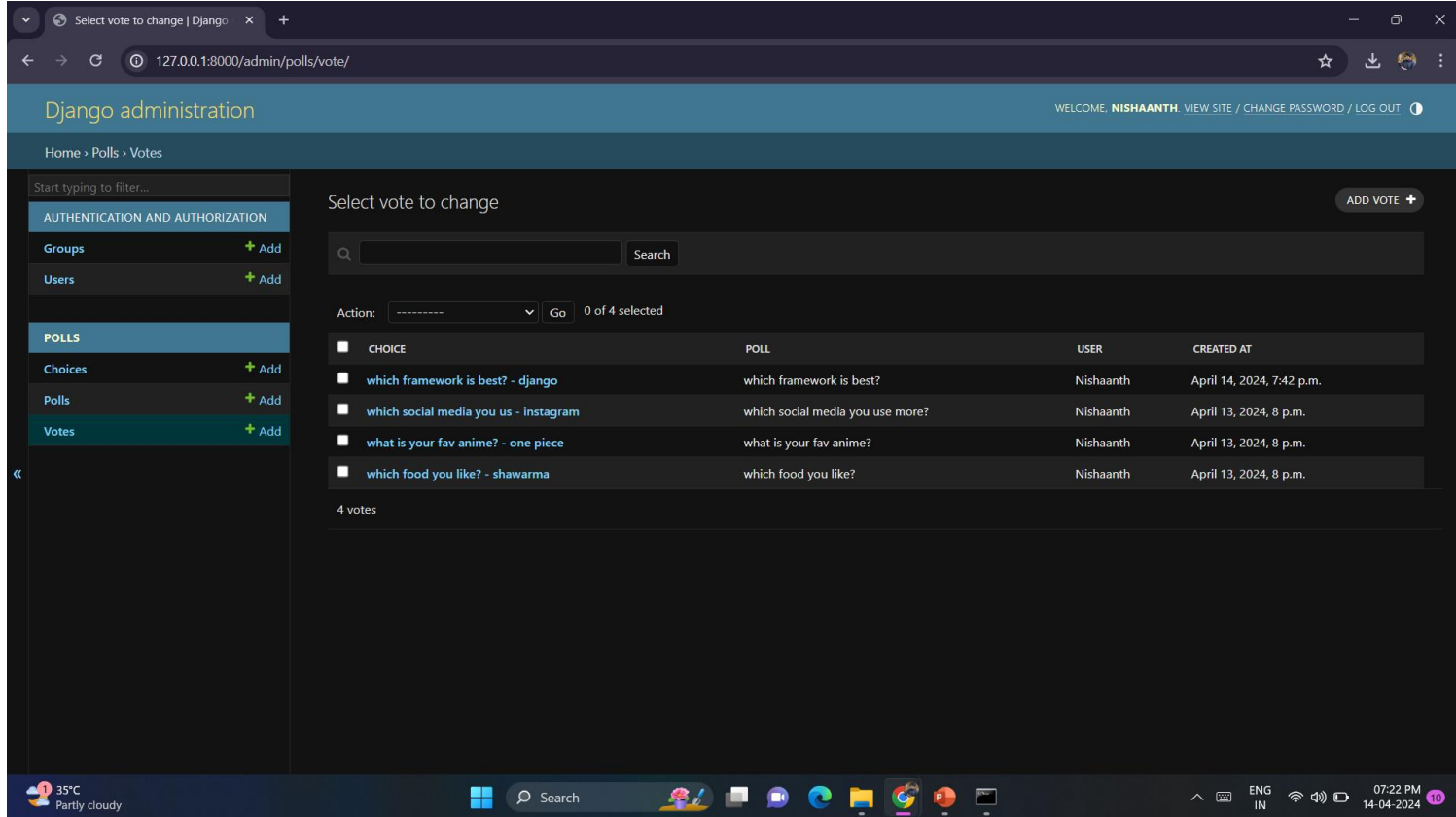
Below the table, it says '4 polls'.

On the right side, there is a 'FILTER' dropdown menu with the following options:

- Show counts
- By active
 - All
 - Yes
 - No
- By created at
 - Any date
 - Today
 - Past 7 days
 - This month
 - This year
- By pub date
 - Any date
 - Today
 - Past 7 days
 - This month
 - This year

The bottom of the screen shows a Windows taskbar with the date 14-04-2024 and time 07:21 PM.

Voting Details Page



127.0.0.1:8000/admin/polls/vote/

Django administration

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

Home > Polls > Votes

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups + Add
- Users + Add

POLLS

- Choices + Add
- Polls + Add
- Votes + Add

Select vote to change

ADD VOTE +

Search

Action: 0 of 4 selected

CHOICE	POLL	USER	CREATED AT
<input type="checkbox"/> which framework is best? - django	which framework is best?	Nishaanth	April 14, 2024, 7:42 p.m.
<input type="checkbox"/> which social media you us - instagram	which social media you use more?	Nishaanth	April 13, 2024, 8 p.m.
<input type="checkbox"/> what is your fav anime? - one piece	what is your fav anime?	Nishaanth	April 13, 2024, 8 p.m.
<input type="checkbox"/> which food you like? - shawarma	which food you like?	Nishaanth	April 13, 2024, 8 p.m.

4 votes

35°C Partly cloudy

Search

ENG IN

07:22 PM 14-04-2024

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!