



NEXT GEN EMPLOYABILITY PROGRAM

Creating a future-ready workforce

Student Name :Thenmozhi M
Student ID :autb21csl012

College Name

Arasu Engineering College

CAPSTONE PROJECT SHOWCASE

Project Title

Voting Application using Django Framework-THENMOZHI
MARIYAPPAN(4314,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:

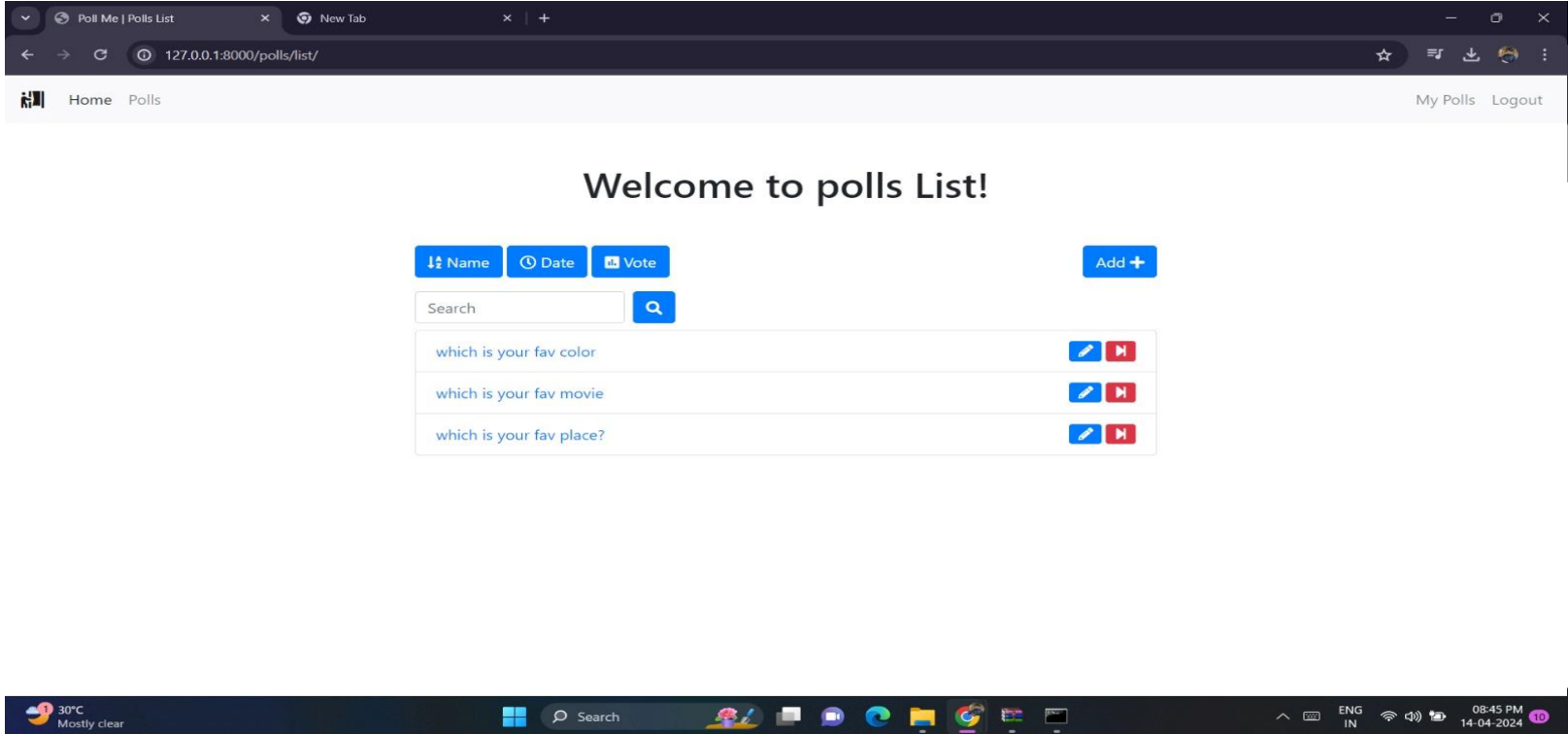
- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication 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 screenshot shows a web browser window with two tabs: 'Poll Me | Polls List' and 'New Tab'. The address bar shows the URL '127.0.0.1:8000/polls/list/'. The page has a navigation bar with 'Home' and 'Polls' links, and a user profile icon with 'My Polls' and 'Logout' links. The main content area displays a welcome message 'Welcome to polls List!' and a table of polls. The table has columns for Name, Date, and Vote. There are three polls listed: 'which is your fav color', 'which is your fav movie', and 'which is your fav place?'. Each poll has an edit icon (pencil) and a delete icon (trash). A search bar is located above the table, and an 'Add +' button is in the top right corner of the table area. The Windows taskbar at the bottom shows the date and time as 08:45 PM on 14-04-2024, along with various system icons and the language set to ENG IN.

Poll Me | Polls List New Tab

127.0.0.1:8000/polls/list/

Home Polls My Polls Logout

Welcome to polls List!

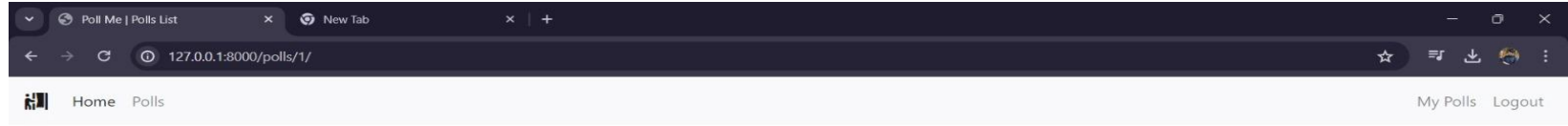
↑↓ Name ⌚ Date 🗳 Vote Add +

Search 🔍

which is your fav color	✎	🗑
which is your fav movie	✎	🗑
which is your fav place?	✎	🗑

30°C Mostly clear Search ENG IN 08:45 PM 14-04-2024

Poll Page



Polls details page

which is your fav color

☒ black

☐ white

Vote

Cancel

Voting Page



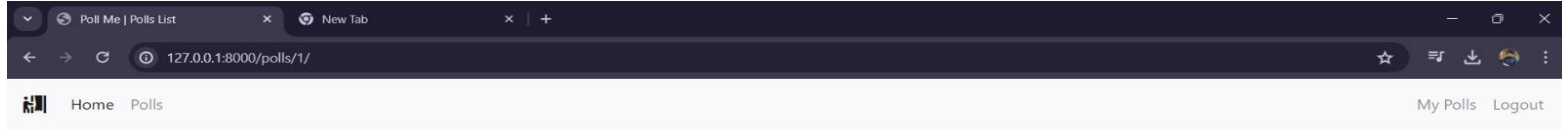
Result for: which is your fav place?

Total: 1 votes

tirupati-100%	
mayiladuthurai	0
tirupati	1

[Back To Polls](#)

Voting Details Page



Polls details page

which is your fav color

- ☒ black
☐ white

[Vote](#)

[Cancel](#)

Admin Login Page



Login

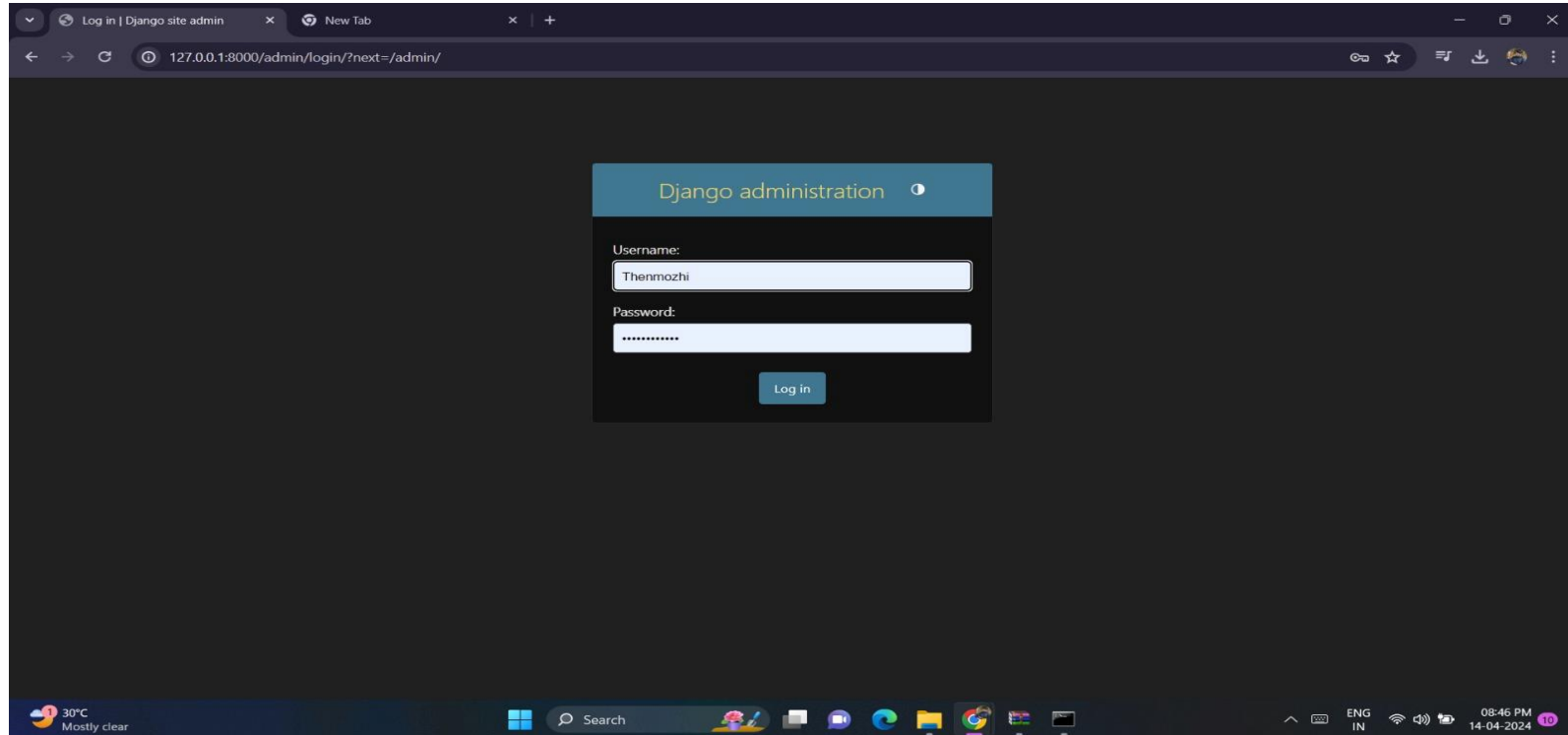
Username

Password

[Login](#)

Don't have an account? [Sign Up](#)

Admin Home Page



The screenshot shows a web browser window with two tabs: 'Log in | Django site admin' and 'New Tab'. The address bar displays '127.0.0.1:8000/admin/login/?next=/admin/'. The main content area is dark gray and features a central white box titled 'Django administration' with a help icon. Inside this box, there are two input fields: 'Username:' with the value 'Thenmozhi' and 'Password:' with masked characters. Below the password field is a 'Log in' button. The Windows taskbar at the bottom shows the system clock as 08:46 PM on 14-04-2024, along with weather information (30°C, Mostly clear) and various application icons.

Log in | Django site admin New Tab

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

Django administration ⓘ

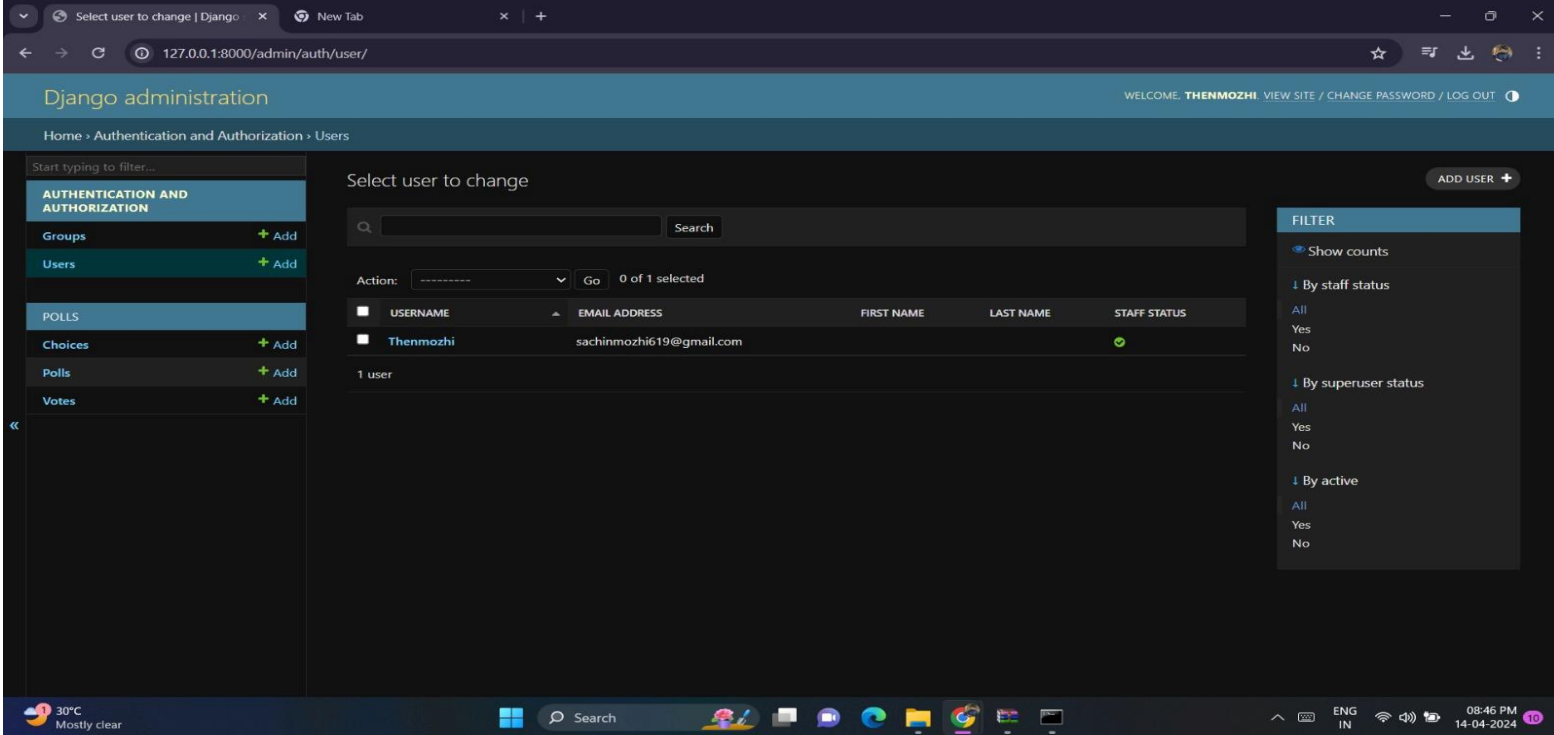
Username:

Password:

Log in

30°C Mostly clear Search ENG IN 08:46 PM 14-04-2024

Authentication and Authorization Page



The screenshot displays the Django administration interface for the Authentication and Authorization section. The browser address bar shows the URL `127.0.0.1:8000/admin/auth/user/`. The page title is "Django administration" and the user is logged in as "THENMOZHI".

The left sidebar contains the following navigation items:

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

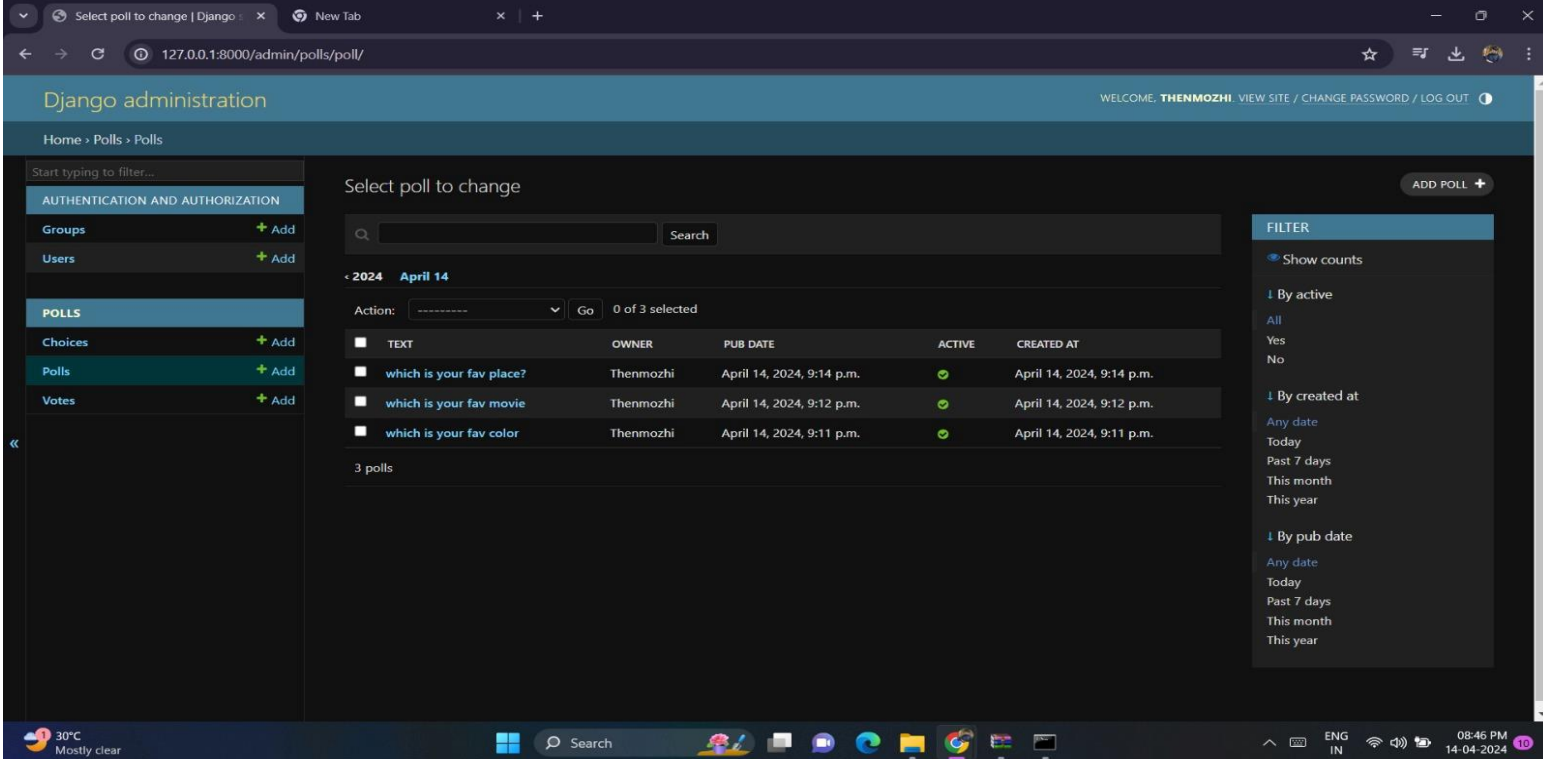
The main content area is titled "Select user to change". It features a search bar and a table of users. The table has columns for USERNAME, EMAIL ADDRESS, FIRST NAME, LAST NAME, and STAFF STATUS. One user, "Thenmozhi", is listed with the email address "sachinmozhi619@gmail.com" and a staff status of "Yes".

The right sidebar contains a "FILTER" section with the following options:

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

The bottom of the screen shows the Windows taskbar with the date and time "08:46 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 interface is in dark mode. On the left, a sidebar contains navigation links: 'AUTHENTICATION AND AUTHORIZATION' (Groups, Users), 'POLLS' (Choices, Polls, Votes), and a search bar. The main content area is titled 'Select poll to change' and features a search bar, a date filter for '2024 April 14', and a table of polls. The table has columns for 'TEXT', 'OWNER', 'PUB DATE', 'ACTIVE', and 'CREATED AT'. Three polls are listed, all owned by 'Thenmozhi' and created on April 14, 2024. A right sidebar contains a 'FILTER' section with options for 'Show counts', 'By active', 'By created at', and 'By pub date'. The bottom of the screen shows a Windows taskbar with the date and time as 08:46 PM on 14-04-2024.

Django administration

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

Home > Polls > Polls

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups + Add
- Users + Add

POLLS

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

Select poll to change

Search

2024 April 14

Action: [dropdown] Go 0 of 3 selected

TEXT	OWNER	PUB DATE	ACTIVE	CREATED AT
<input type="checkbox"/> which is your fav place?	Thenmozhi	April 14, 2024, 9:14 p.m.	<input checked="" type="checkbox"/>	April 14, 2024, 9:14 p.m.
<input type="checkbox"/> which is your fav movie	Thenmozhi	April 14, 2024, 9:12 p.m.	<input checked="" type="checkbox"/>	April 14, 2024, 9:12 p.m.
<input type="checkbox"/> which is your fav color	Thenmozhi	April 14, 2024, 9:11 p.m.	<input checked="" type="checkbox"/>	April 14, 2024, 9:11 p.m.

3 polls

ADD POLL +

FILTER

- 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

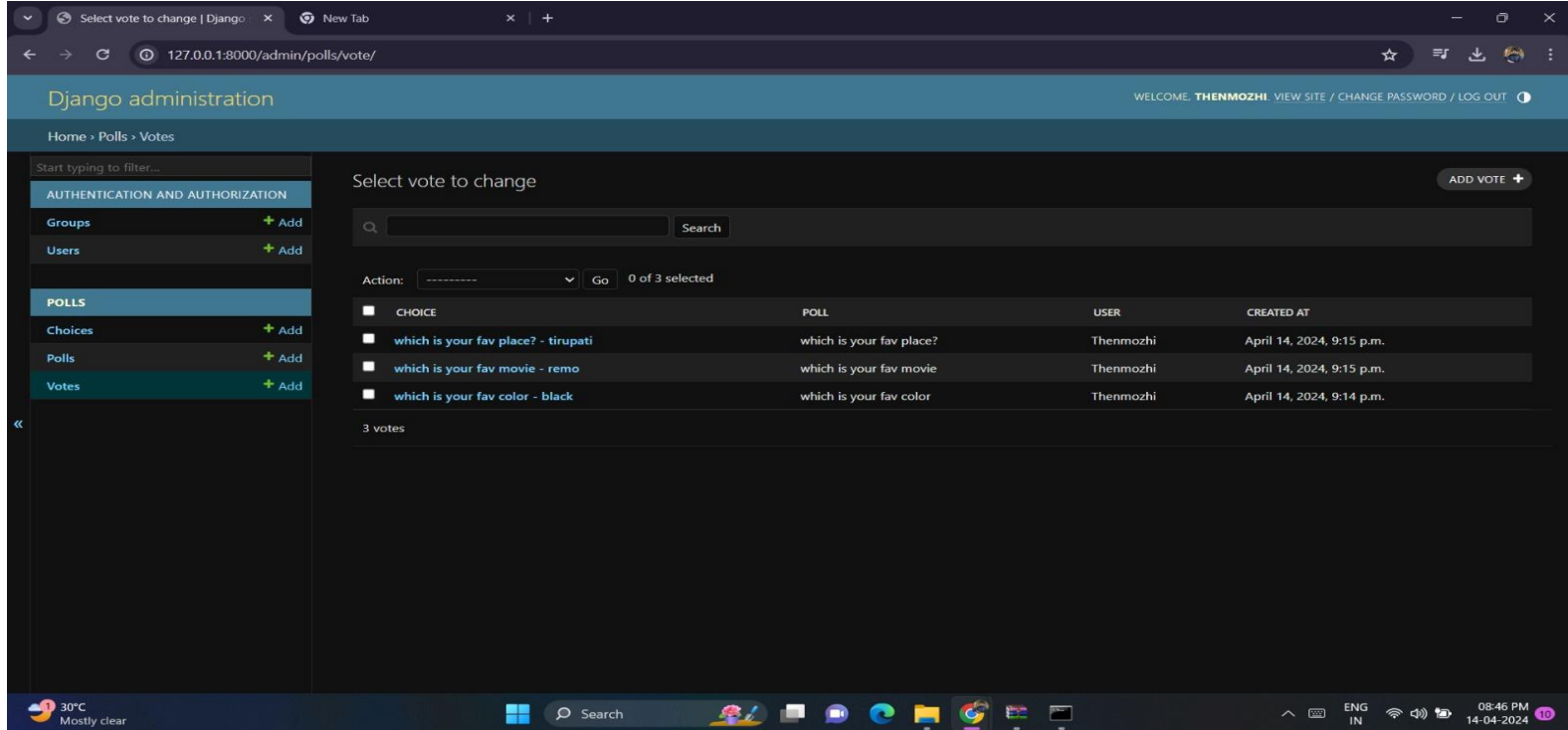
30°C Mostly clear

Search

ENG IN

08:46 PM 14-04-2024

Voting Details Page



The screenshot shows the Django administration interface for the 'polls' app. The left sidebar contains the following menu items:

- 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 vote to change' and includes a search bar and an 'ADD VOTE +' button. Below the search bar, there is an 'Action:' dropdown menu and a 'Go' button, indicating that 0 of 3 items are selected.

CHOICE	POLL	USER	CREATED AT
<input type="checkbox"/> which is your fav place? - tirupati	which is your fav place?	Thenmozhi	April 14, 2024, 9:15 p.m.
<input type="checkbox"/> which is your fav movie - remo	which is your fav movie	Thenmozhi	April 14, 2024, 9:15 p.m.
<input type="checkbox"/> which is your fav color - black	which is your fav color	Thenmozhi	April 14, 2024, 9:14 p.m.

Below the table, it indicates '3 votes'.

The Windows taskbar at the bottom shows the date and time as 08:46 PM on 14-04-2024, along with system icons for network, volume, and battery.

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!