



NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name :Venkatesh K
Student ID :au820621104315

College Name

Arasu Engineering College

CAPSTONE PROJECT SHOWCASE

Project Title

Voting Application using Django Framework-Venkatesh K(4315,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:

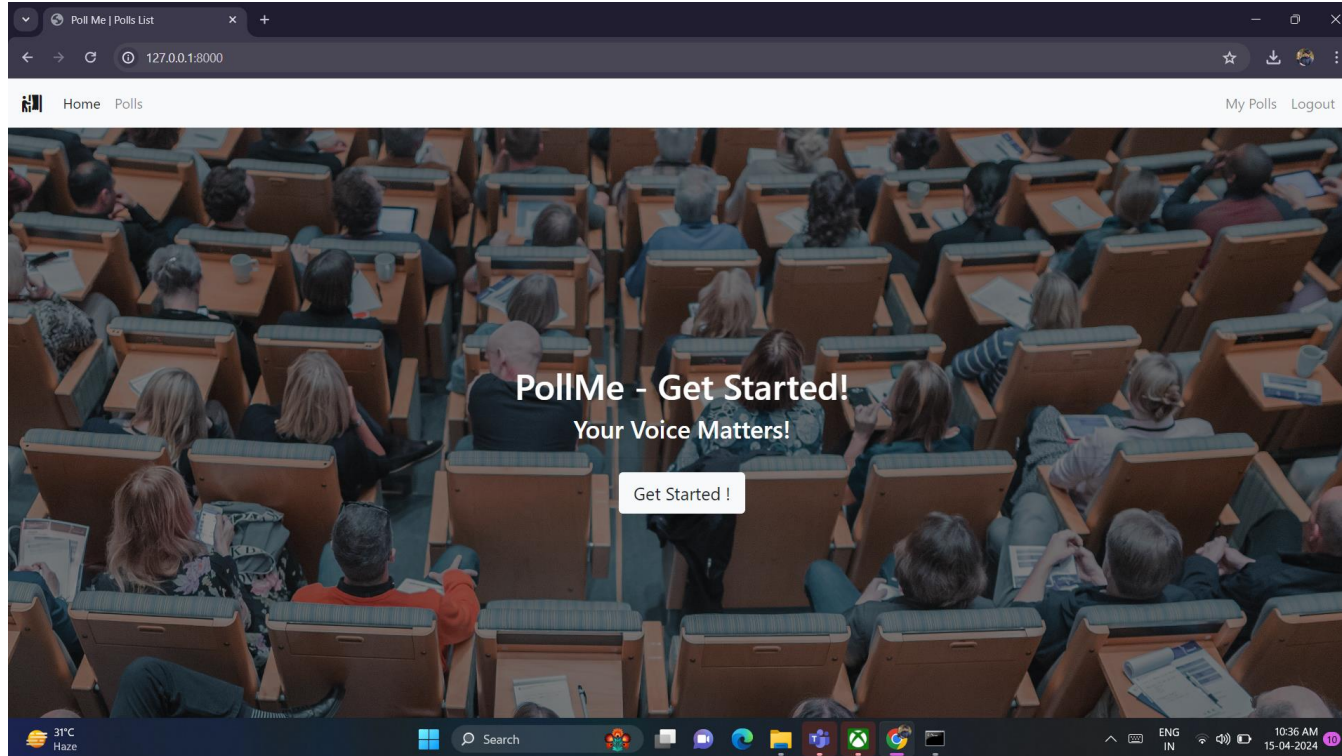
- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication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

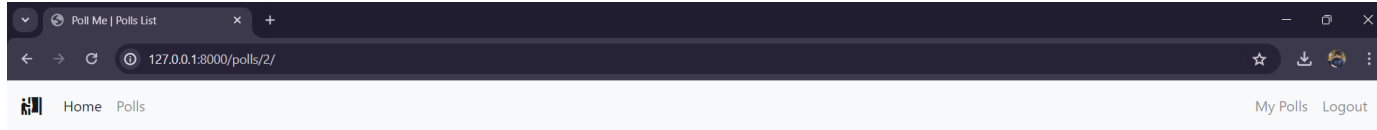
The proposed solution for a voting application using the Django framework is to create a secure and user-friendly online voting platform. The application will allow users to register, vote, and view real-time results. To build the application, the Django framework will be used as the foundation due to 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



Polls details page

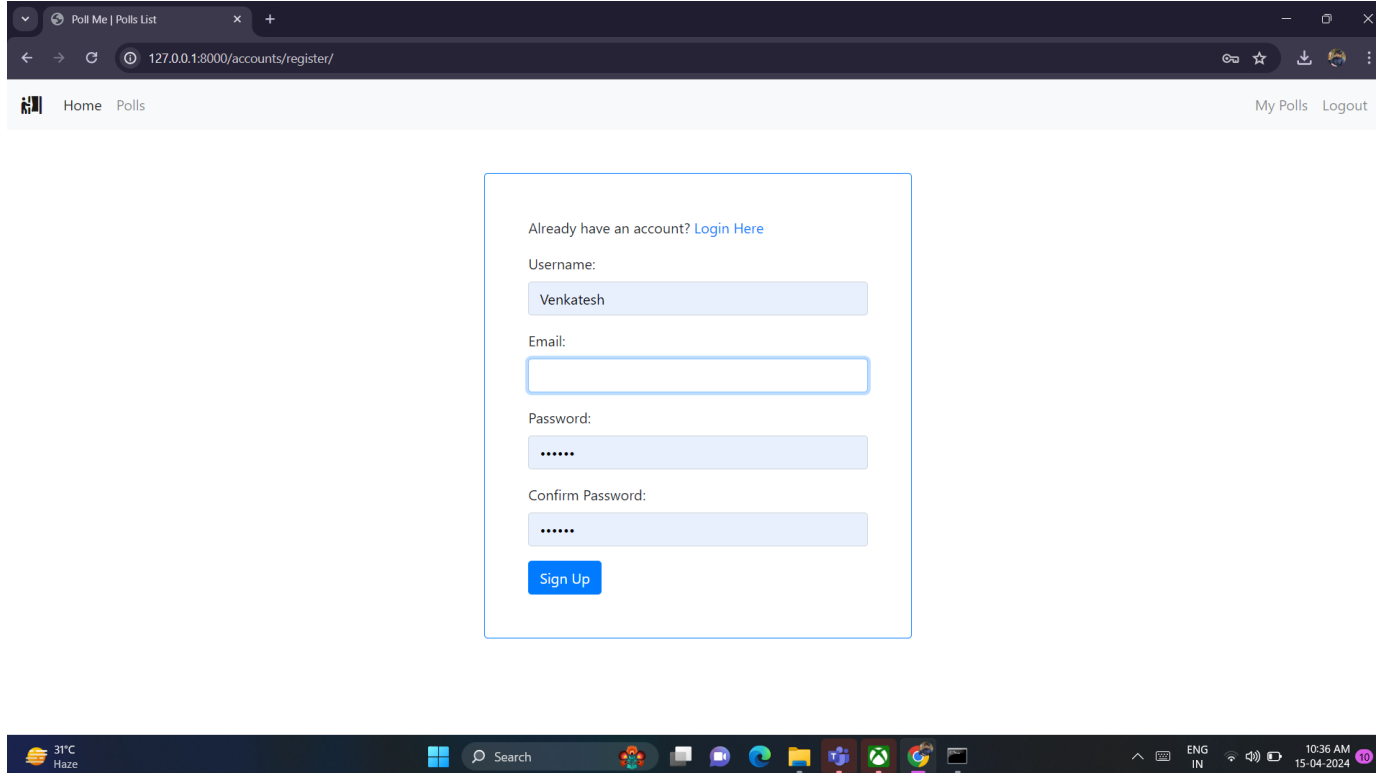
How to create an unordered list (a list with the list items in bullets) in HTML?

- ☐
- ☐
- ☐

[Vote](#)

[Cancel](#)

Voting Page



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/accounts/register/". The browser's address bar also shows "Poll Me | Polls List" and navigation icons. The page has a header with "Home" and "Polls" links, and "My Polls" and "Logout" links on the right. The main content area contains a registration form with the following fields and buttons:

Already have an account? [Login Here](#)

Username:

Email:

Password:

Confirm Password:

The Windows taskbar at the bottom shows the system clock as 10:36 AM on 15-04-2024, along with various system icons and application shortcuts.

Voting Details Page



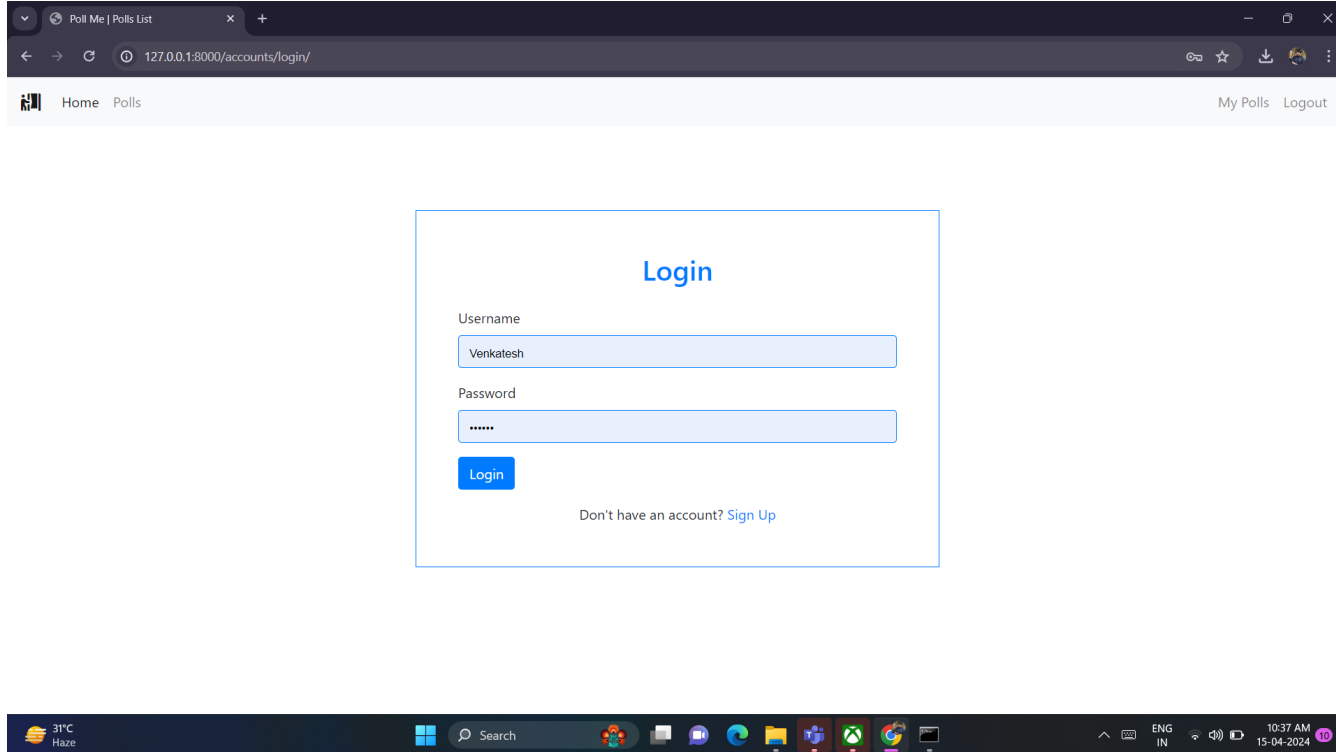
Result for: How to create an unordered list (a list with the list items in bullets) in HTML?

Total: 1 votes

| --100% | |
|------------|---|
| | 1 |
| | 0 |
| | 0 |

[Back To Polls](#)

Admin Login Page

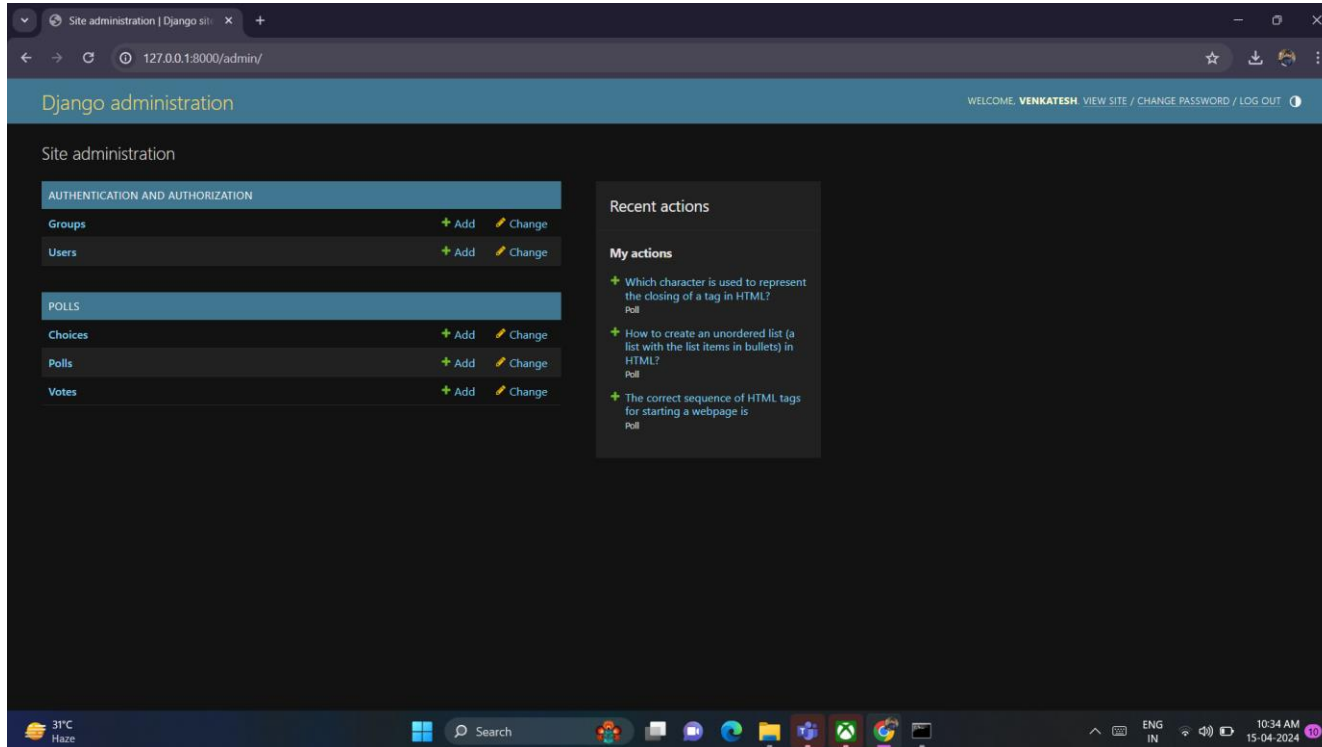


The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/accounts/login/`. The browser's navigation bar includes links for Home and Polls, and user options for My Polls and Logout. The main content area features a login form with the following elements:

- Login** (Section Header)
- Username** label above a text input field containing "Venkatesh".
- Password** label above a password input field containing "*****".
- Login** button.
- Link: "Don't have an account? [Sign Up](#)".

The Windows taskbar at the bottom shows the system clock as 10:37 AM on 15-04-2024, with a temperature of 31°C and weather condition of Haze.

Admin Home Page



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

The main content area is titled "Site administration" and is divided into two columns. The left column contains two sections: "AUTHENTICATION AND AUTHORIZATION" and "POLLS".

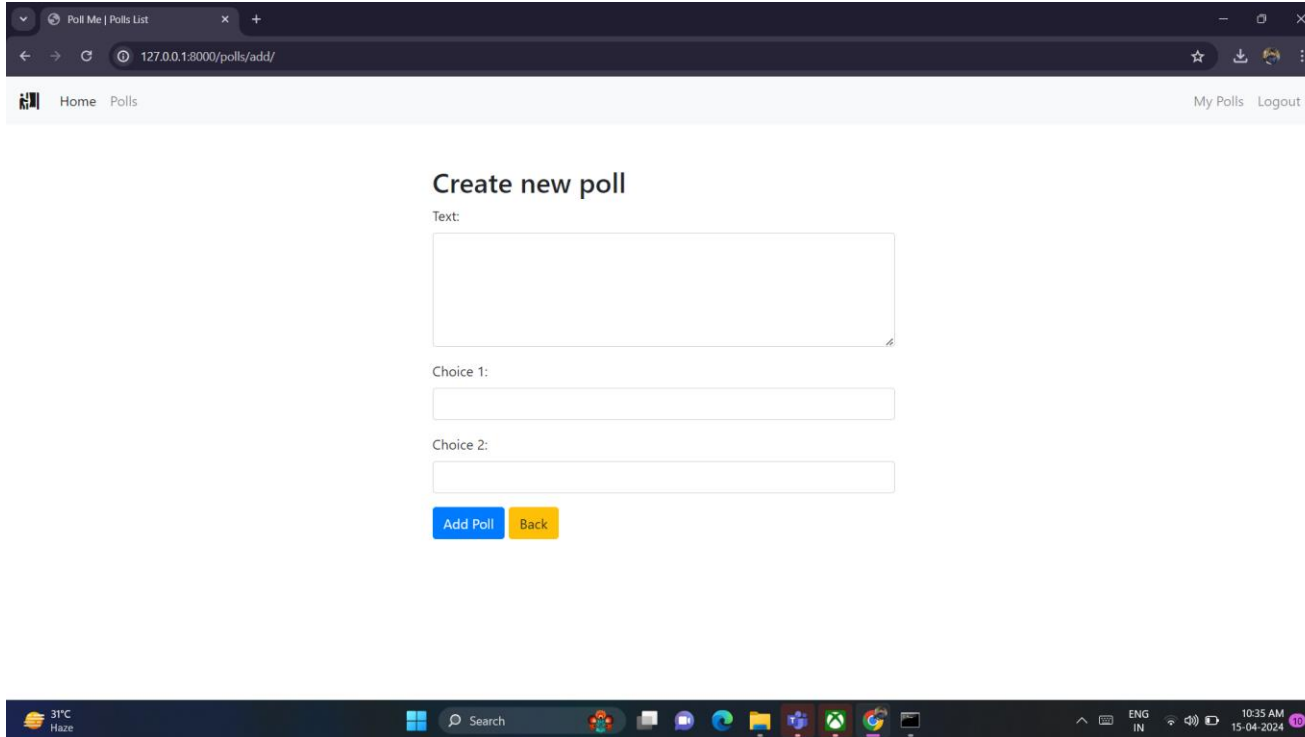
- AUTHENTICATION AND AUTHORIZATION**
 - Groups**: Includes links for "Add" and "Change".
 - Users**: Includes links for "Add" and "Change".
- POLLS**
 - Choices**: Includes links for "Add" and "Change".
 - Polls**: Includes links for "Add" and "Change".
 - Votes**: Includes links for "Add" and "Change".

The right column is titled "Recent actions" and contains a section "My actions" with three entries, each marked with a green plus icon:

- Which character is used to represent the closing of a tag in HTML?
Poll
- How to create an unordered list (a list with the list items in bullets) in HTML?
Poll
- The correct sequence of HTML tags for starting a webpage is
Poll

The Windows taskbar at the bottom shows the system clock as 10:34 AM on 15-04-2024, with a temperature of 31°C and a "Haze" weather condition.

Authentication and Authorization Page



Poll Me | Polls List

127.0.0.1:8000/polls/add/

Home Polls My Polls Logout

Create new poll

Text:

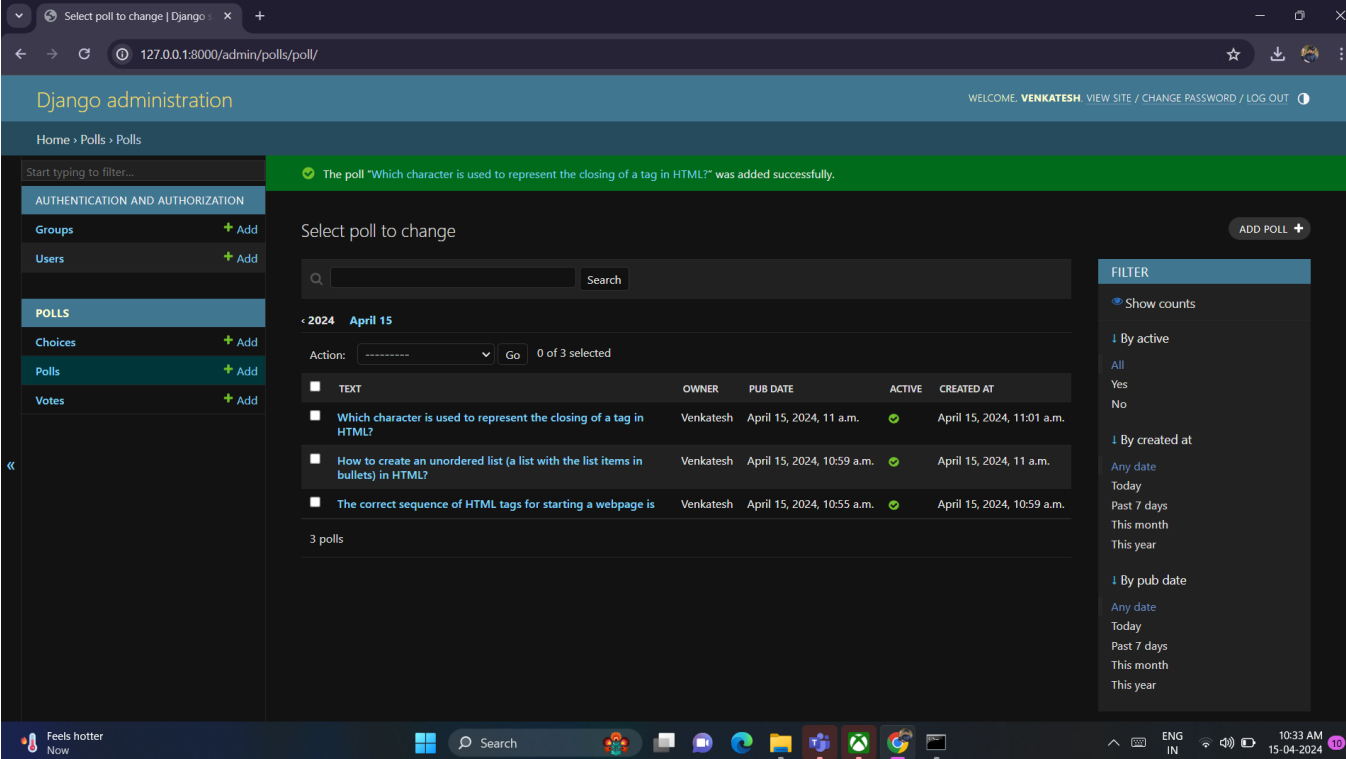
Choice 1:

Choice 2:

Add Poll Back

31°C Haze Search ENG IN 10:35 AM 15-04-2024

Questions Adding Section Page



Django administration

WELCOME: VENKATESH 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 [ADD POLL +](#)

Search

< 2024 April 15

Action: [-----](#) Go 0 of 3 selected

| <input type="checkbox"/> | TEXT | OWNER | PUB DATE | ACTIVE | CREATED AT |
|-------------------------------------|--|-----------|----------------------------|----------------|----------------------------|
| <input checked="" type="checkbox"/> | Which character is used to represent the closing of a tag in HTML? | Venkatesh | April 15, 2024, 11 a.m. | ✓ | April 15, 2024, 11:01 a.m. |
| <input type="checkbox"/> | How to create an unordered list (a list with the list items in bullets) in HTML? | Venkatesh | April 15, 2024, 10:59 a.m. | ✓ | April 15, 2024, 11 a.m. |
| <input type="checkbox"/> | The correct sequence of HTML tags for starting a webpage is | Venkatesh | April 15, 2024, 10:55 a.m. | ✓ | April 15, 2024, 10:59 a.m. |

3 polls

FILTER

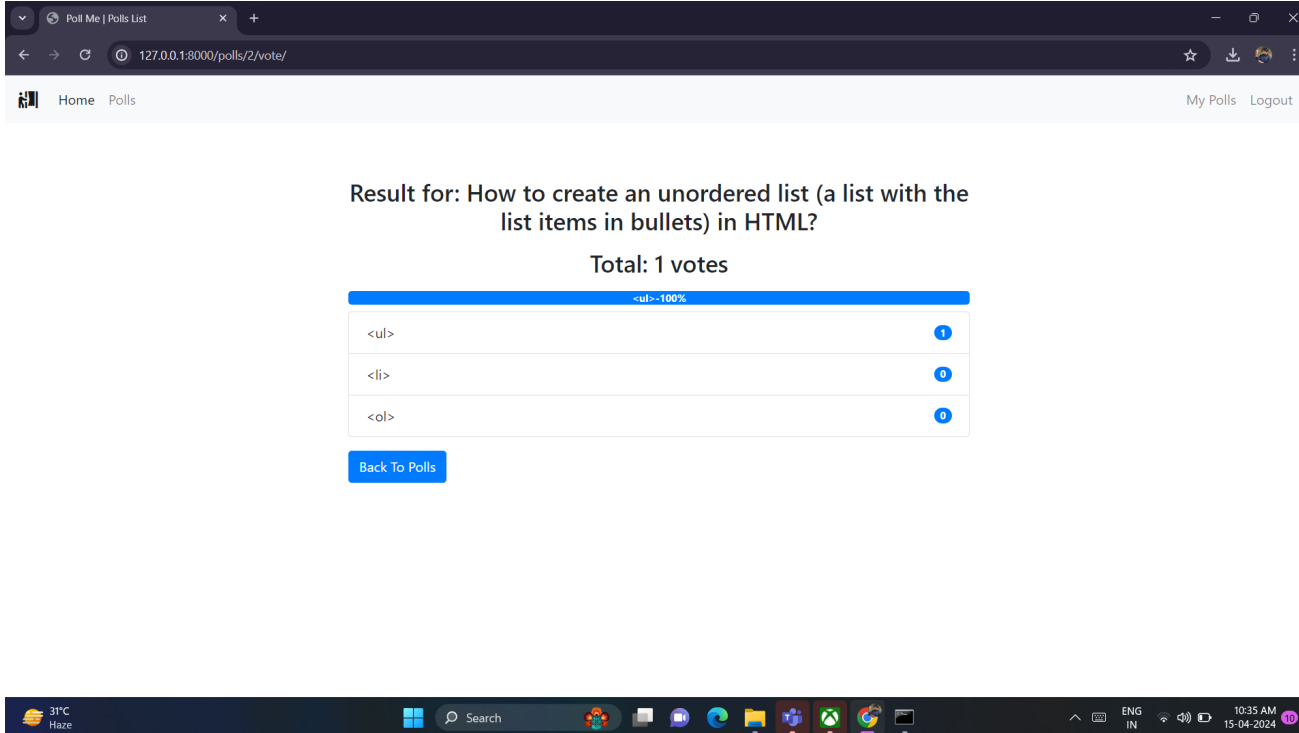
- Show counts
- By active
 - All
 - Yes
 - No
- By created at
 - Any date
 - Today
 - Past 7 days
 - This month
 - This year
- By pub date
 - Any date
 - Today
 - Past 7 days
 - This month
 - This year

Feels hotter Now

Search

ENG IN 10:33 AM 15-04-2024

Voting Details Page



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/polls/2/vote/". The page has a navigation bar with "Home" and "Polls" links, and "My Polls" and "Logout" buttons. The main content area displays the poll question: "Result for: How to create an unordered list (a list with the list items in bullets) in HTML?". Below the question, it says "Total: 1 votes". A table shows the voting results for three options: "", "", and "". The first option has 1 vote, while the others have 0. A "Back To Polls" button is at the bottom.

| -100% | |
|-----------|---|
| | 1 |
| | 0 |
| | 0 |

[Back To Polls](#)

31°C Haze 10:35 AM 15-04-2024

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!