



# NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name :P.Dharunesh  
Student ID :autb21cs1004

College Name

Arasu Engineering College

# CAPSTONE PROJECT SHOWCASE

## Project Title

Voting Application using Django Framework-P.Dharunesh(4303,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:

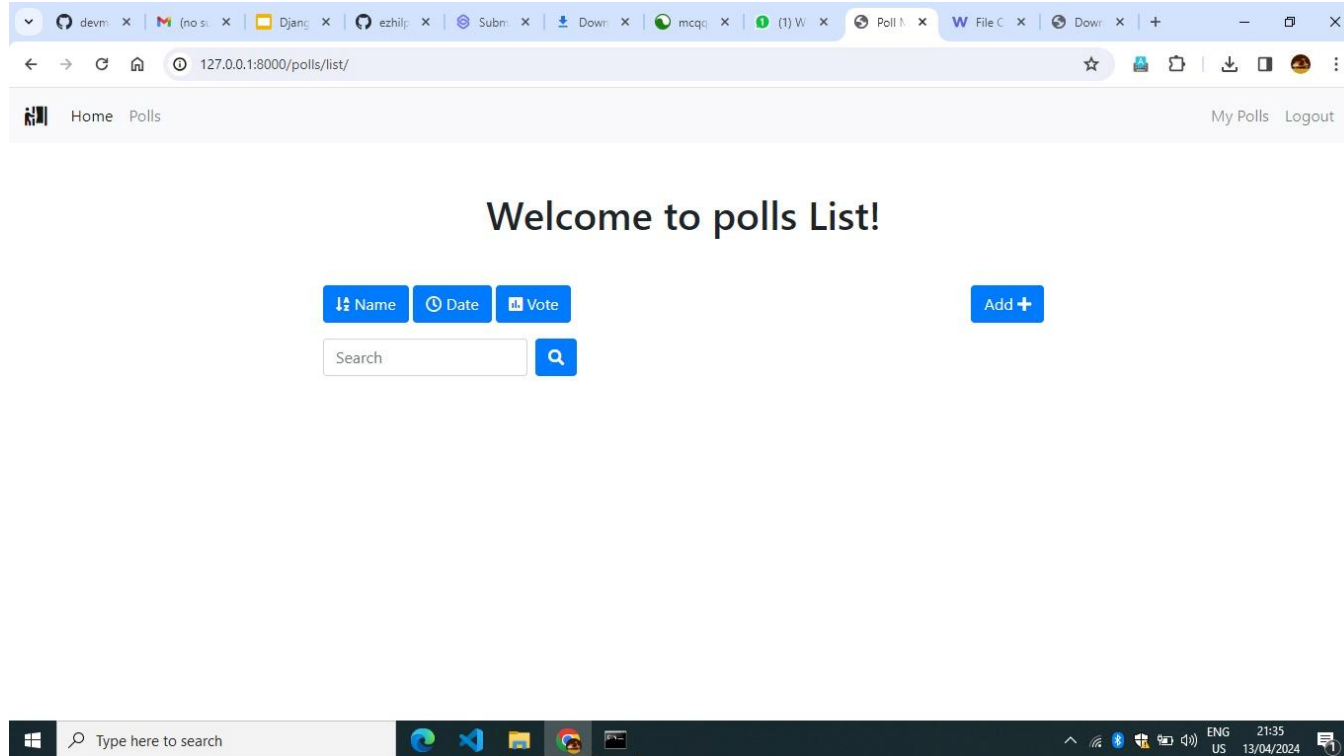
- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication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

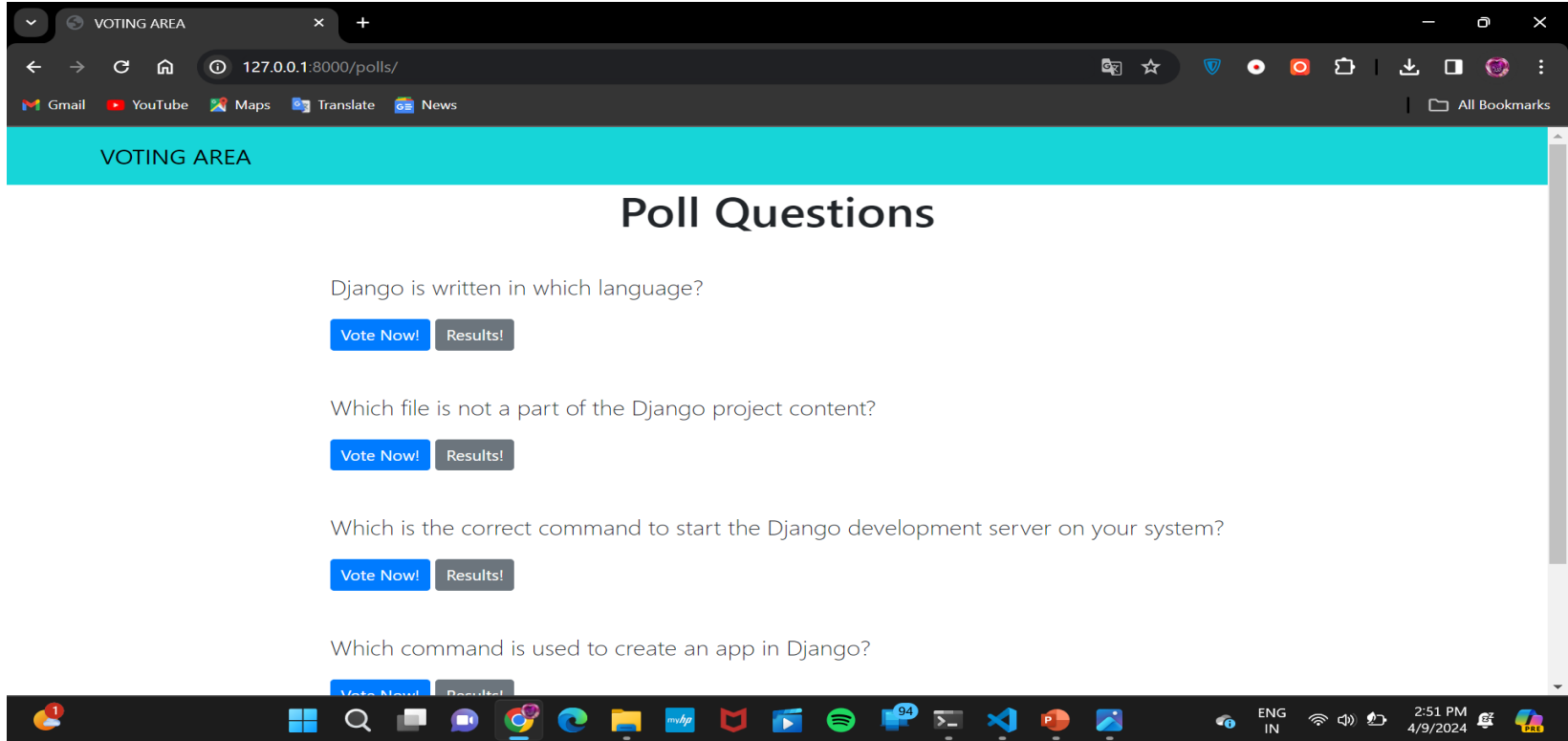
The proposed solution for a voting application using the Django framework is to create a secure and user-friendly online voting platform. The application will allow users to register, vote, and view real-time results. To build the application, the Django framework will be used as the foundation due to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



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/polls/". The page has a teal header with the text "VOTING AREA". Below the header, the main heading is "Poll Questions". There are four poll questions listed, each with a "Vote Now!" button and a "Results!" button.

VOTING AREA

### Poll Questions

Django is written in which language?

[Vote Now!](#) [Results!](#)

Which file is not a part of the Django project content?

[Vote Now!](#) [Results!](#)

Which is the correct command to start the Django development server on your system?

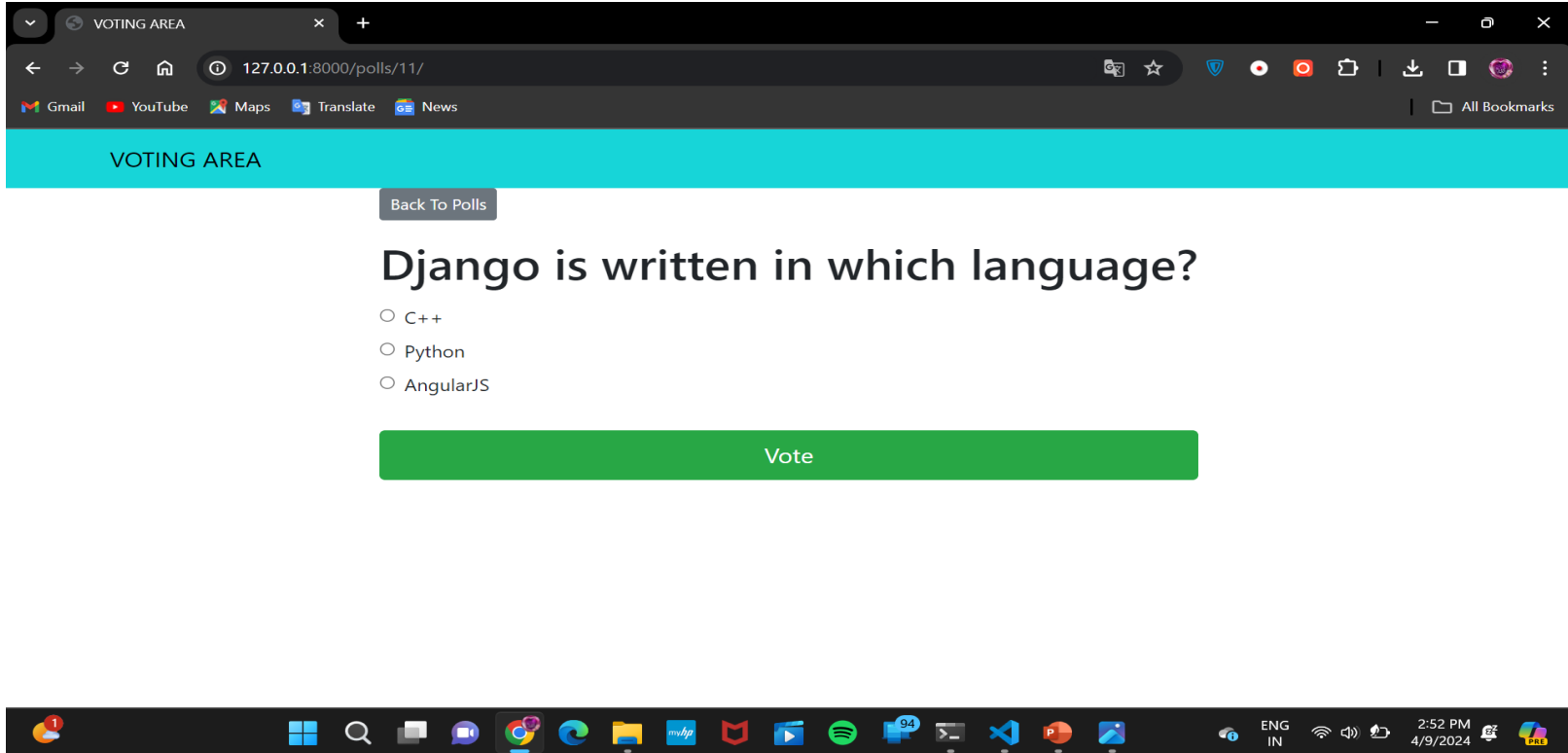
[Vote Now!](#) [Results!](#)

Which command is used to create an app in Django?

[Vote Now!](#) [Results!](#)



## Voting Page



The screenshot shows a web browser window with a single tab titled "VOTING AREA". The address bar displays the URL "127.0.0.1:8000/polls/11/". Below the browser window, a cyan banner contains the text "VOTING AREA". A grey button labeled "Back To Polls" is positioned above the poll question. The poll question is "Django is written in which language?". Below the question are three radio button options: "C++", "Python", and "AngularJS". A large green button labeled "Vote" is centered below the options. The Windows taskbar at the bottom shows various application icons, including the Start menu, Search, Task View, and several open applications like Chrome, Edge, File Explorer, and VS Code. The system tray on the right indicates the language is "ENG IN", shows network and volume icons, and displays the time as "2:52 PM" on "4/9/2024".

VOTING AREA

[Back To Polls](#)

Django is written in which language?

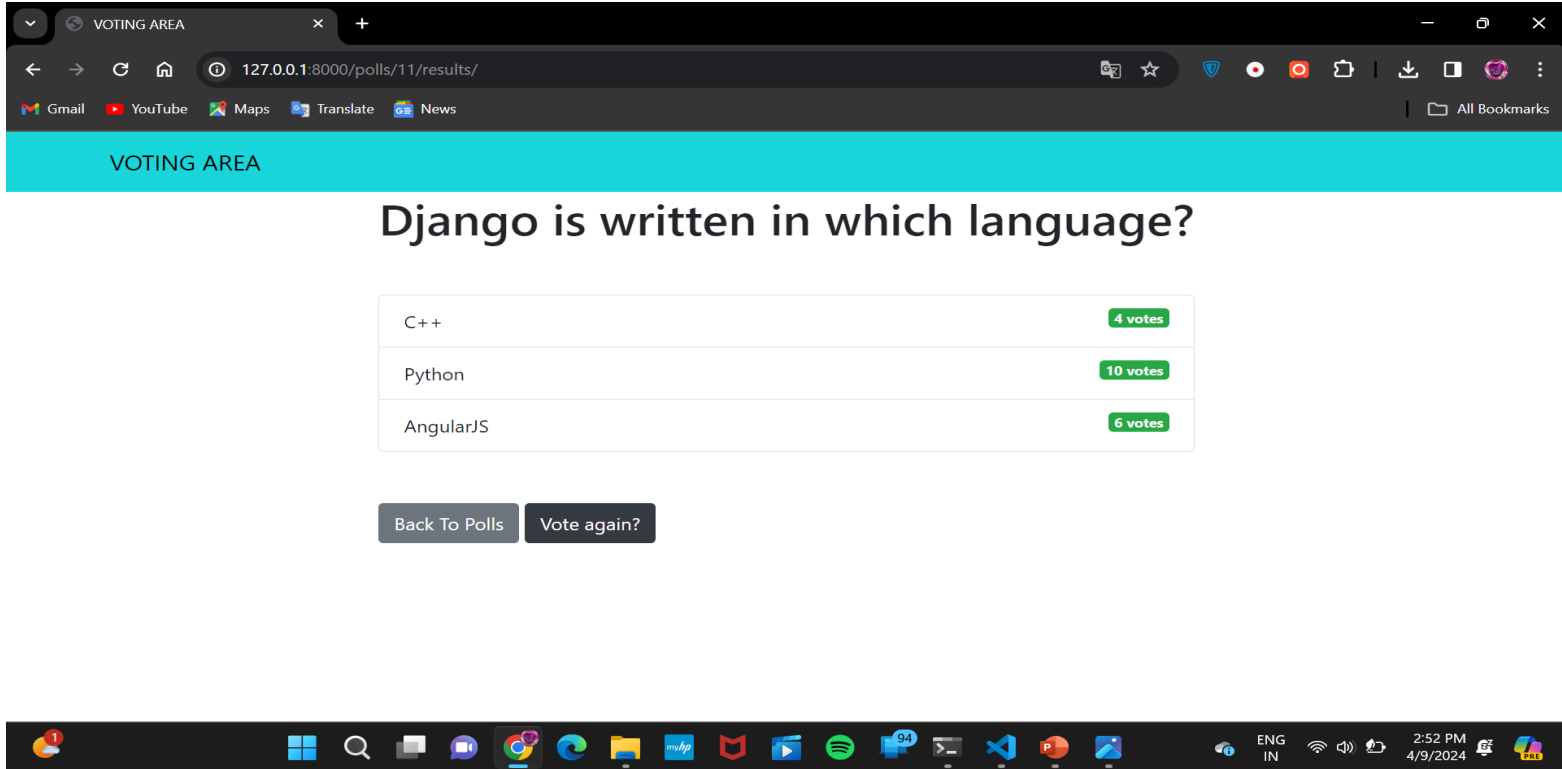
☐ C++

☐ Python

☐ AngularJS

[Vote](#)

## Voting Details Page

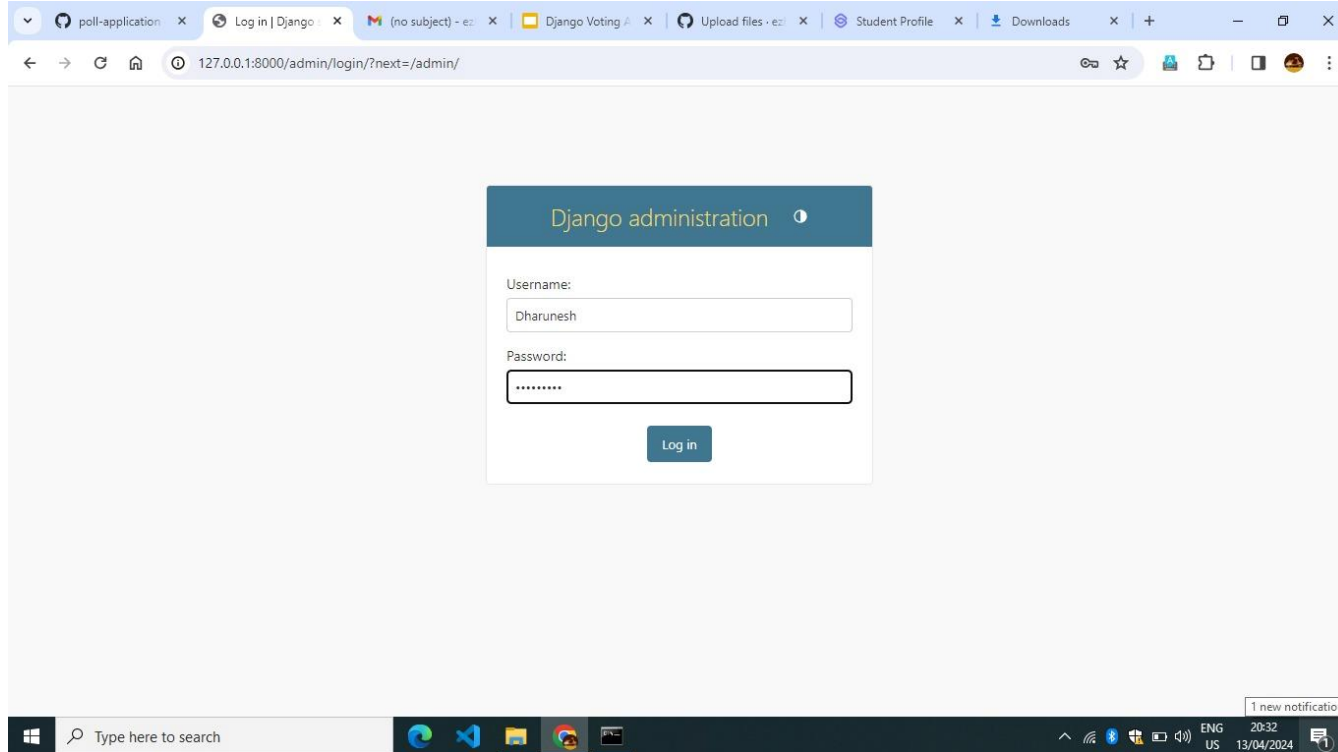


The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000/polls/11/results/`. The page has a teal header with the text "VOTING AREA". Below the header, the question "Django is written in which language?" is displayed. A table lists three options with their respective vote counts:

C++	4 votes
Python	10 votes
AngularJS	6 votes

At the bottom of the page, there are two buttons: "Back To Polls" and "Vote again?". The Windows taskbar at the bottom shows the time as 2:52 PM on 4/9/2024.

## Admin Login Page



The screenshot shows a web browser window with the Django administration login page. The browser's address bar displays the URL `127.0.0.1:8000/admin/login/?next=/admin/`. The page features a central login form with a blue header bar that reads "Django administration". Below the header, the form includes a "Username:" label, a text input field containing "Dharunesh", a "Password:" label, a password input field with masked characters, and a "Log in" button. The browser's taskbar at the bottom shows various application icons and system information, including the date and time (20:32, 13/04/2024).

poll-application x Log in | Django x (no subject) - e: x Django Voting x Upload files - e: x Student Profile x Downloads x + -

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

Django administration

Username:

Dharunesh

Password:

\*\*\*\*\*

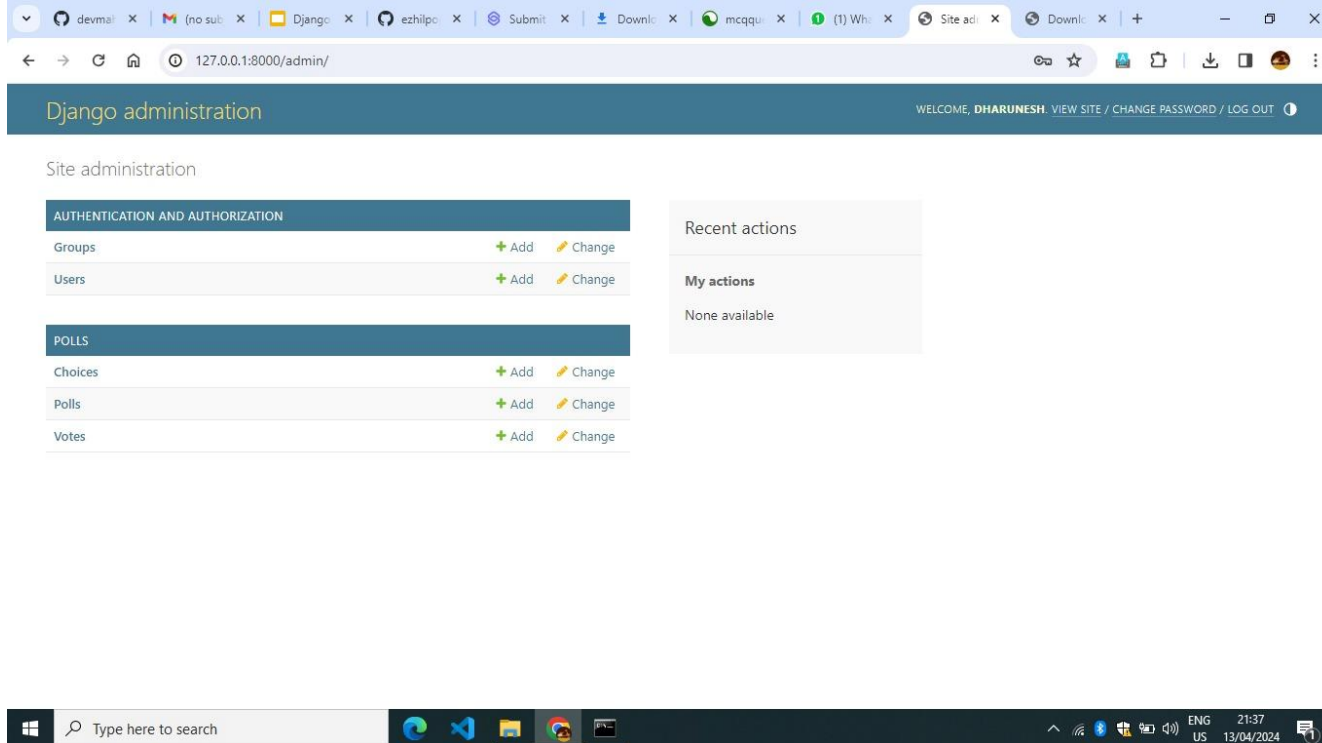
Log in

1 new notification

Type here to search

ENG US 20:32 13/04/2024

## 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 "DHARUNESH" with links to "VIEW SITE", "CHANGE PASSWORD", and "LOG OUT".

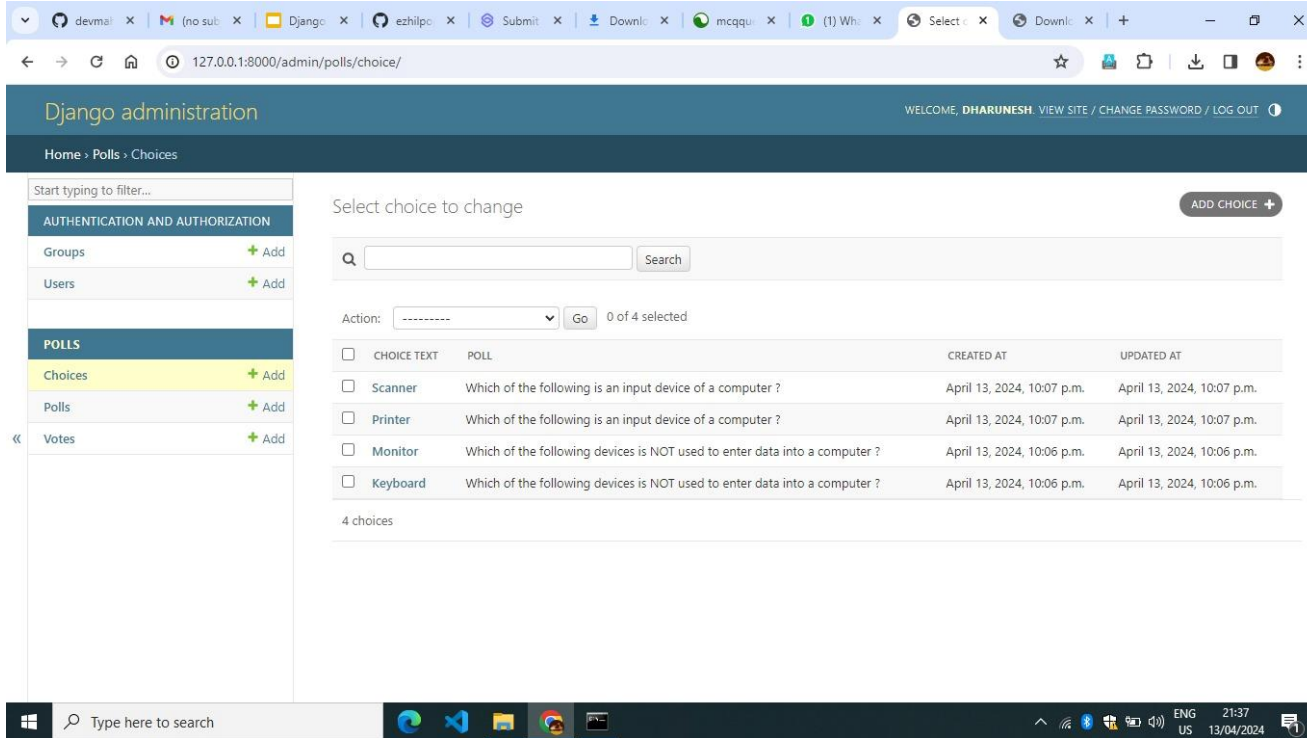
The main content area is titled "Site administration" and is divided into two sections:

- AUTHENTICATION AND AUTHORIZATION**: This section contains two rows of links for managing site administration. The "Groups" row has a green "+ Add" link and a yellow "Change" link. The "Users" row also has a green "+ Add" link and a yellow "Change" link.
- POLLS**: This section contains three rows of links for managing polls. The "Choices" row has a green "+ Add" link and a yellow "Change" link. The "Polls" row has a green "+ Add" link and a yellow "Change" link. The "Votes" row has a green "+ Add" link and a yellow "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 search bar with the text "Type here to search", several application icons (Edge, Teams, File Explorer, etc.), and system tray icons including the date and time (21:37, 13/04/2024).

## Authentication and Authorization Page



The screenshot shows the Django administration interface for the 'Polls' app, specifically the 'Choices' page. The browser address bar shows the URL '127.0.0.1:8000/admin/polls/choice/'. The page header includes the Django logo, the text 'Django administration', and a welcome message for 'DHARUNESH' with links to 'VIEW SITE', 'CHANGE PASSWORD', and 'LOG OUT'.

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

- Start typing to filter...
- AUTHENTICATION AND AUTHORIZATION
  - Groups [+ Add](#)
  - Users [+ Add](#)
- POLLS
  - Choices [+ Add](#)
  - Polls [+ Add](#)
  - Votes [+ Add](#)

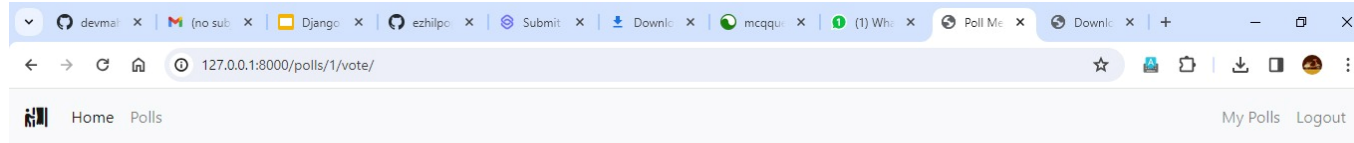
The main content area is titled 'Select choice to change' and features a search bar and an 'ADD CHOICE +' button. Below the search bar, there is an 'Action:' dropdown menu set to '-----' and a 'Go' button, indicating that 0 of 4 choices are selected.

The table displays the following data:

<input type="checkbox"/>	CHOICE TEXT	POLL	CREATED AT	UPDATED AT
<input type="checkbox"/>	Scanner	Which of the following is an input device of a computer ?	April 13, 2024, 10:07 p.m.	April 13, 2024, 10:07 p.m.
<input type="checkbox"/>	Printer	Which of the following is an input device of a computer ?	April 13, 2024, 10:07 p.m.	April 13, 2024, 10:07 p.m.
<input type="checkbox"/>	Monitor	Which of the following devices is NOT used to enter data into a computer ?	April 13, 2024, 10:06 p.m.	April 13, 2024, 10:06 p.m.
<input type="checkbox"/>	Keyboard	Which of the following devices is NOT used to enter data into a computer ?	April 13, 2024, 10:06 p.m.	April 13, 2024, 10:06 p.m.

Below the table, it indicates '4 choices'.

## Questions Adding Section Page



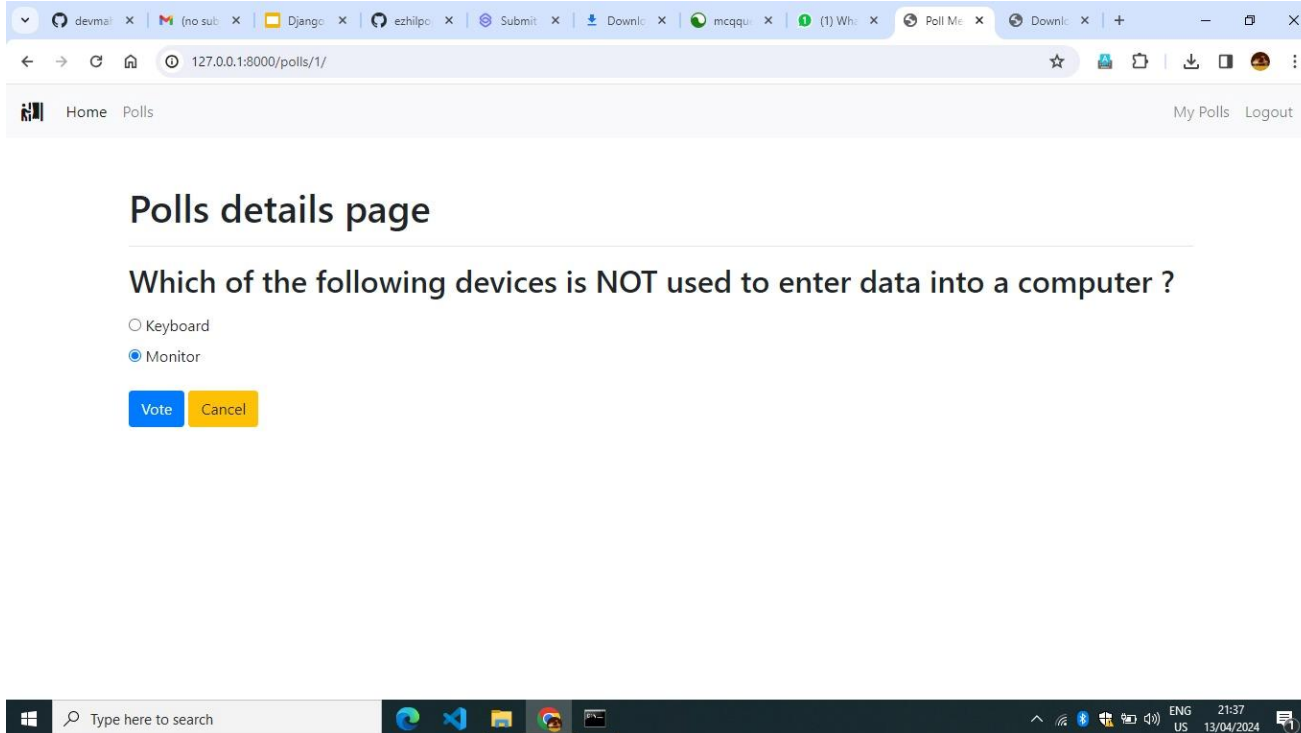
Result for: Which of the following devices is NOT used to enter data into a computer ?

Total: 1 votes

Monitor-100%	
Keyboard	0
Monitor	1

[Back To Polls](#)

## Voting Details Page



devmail x | (no sub: x | Django x | ezhilpo x | Submit x | Downlo: x | mcqqu: x | (1) Wh: x | Poll Me x | Downlo: x | +

127.0.0.1:8000/polls/1/

Home Polls My Polls Logout

### Polls details page

Which of the following devices is NOT used to enter data into a computer ?

☐ Keyboard

☒ Monitor

Vote Cancel

Type here to search

ENG 21:37  
US 13/04/2024

## 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!**