



# NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name :Surya R  
Student ID :au820621104084

College Name

Arasu Engineering College

# CAPSTONE PROJECT SHOWCASE

## Project Title

Voting Application using Django Framework-Surya(4084,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:

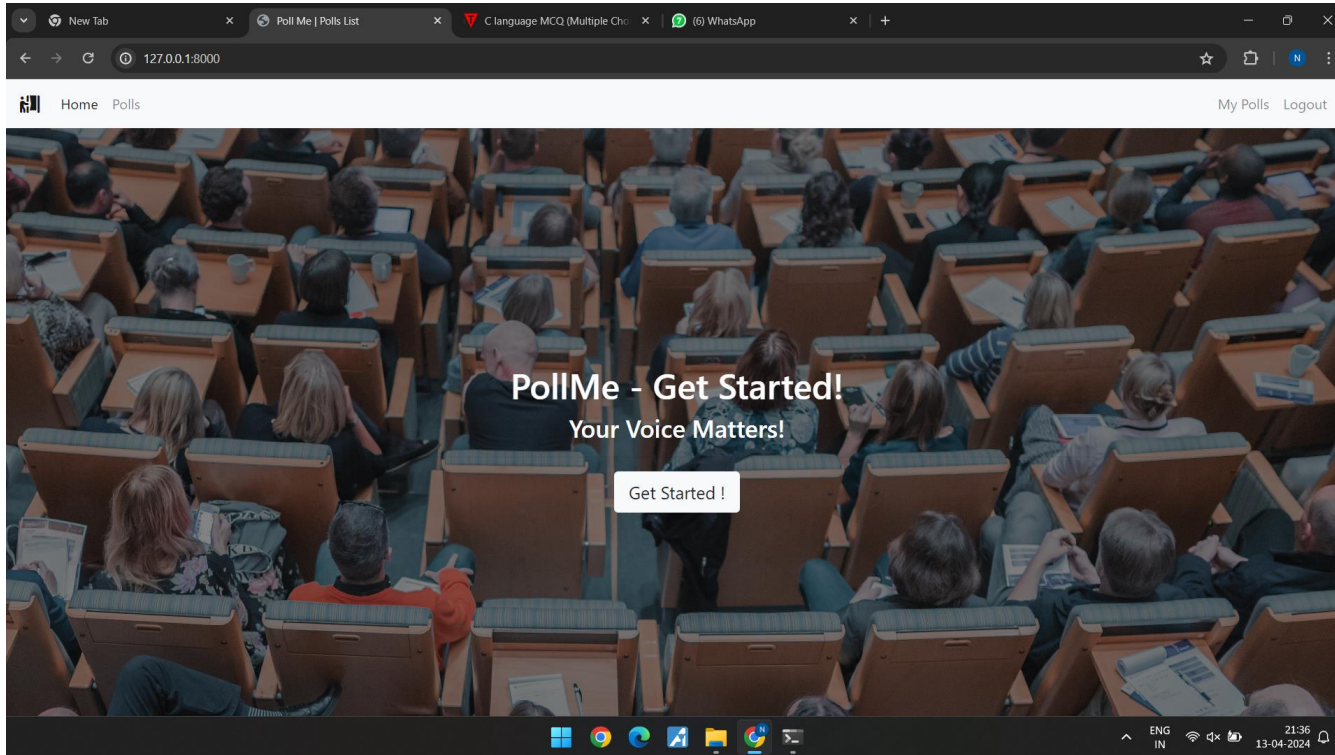
- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication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

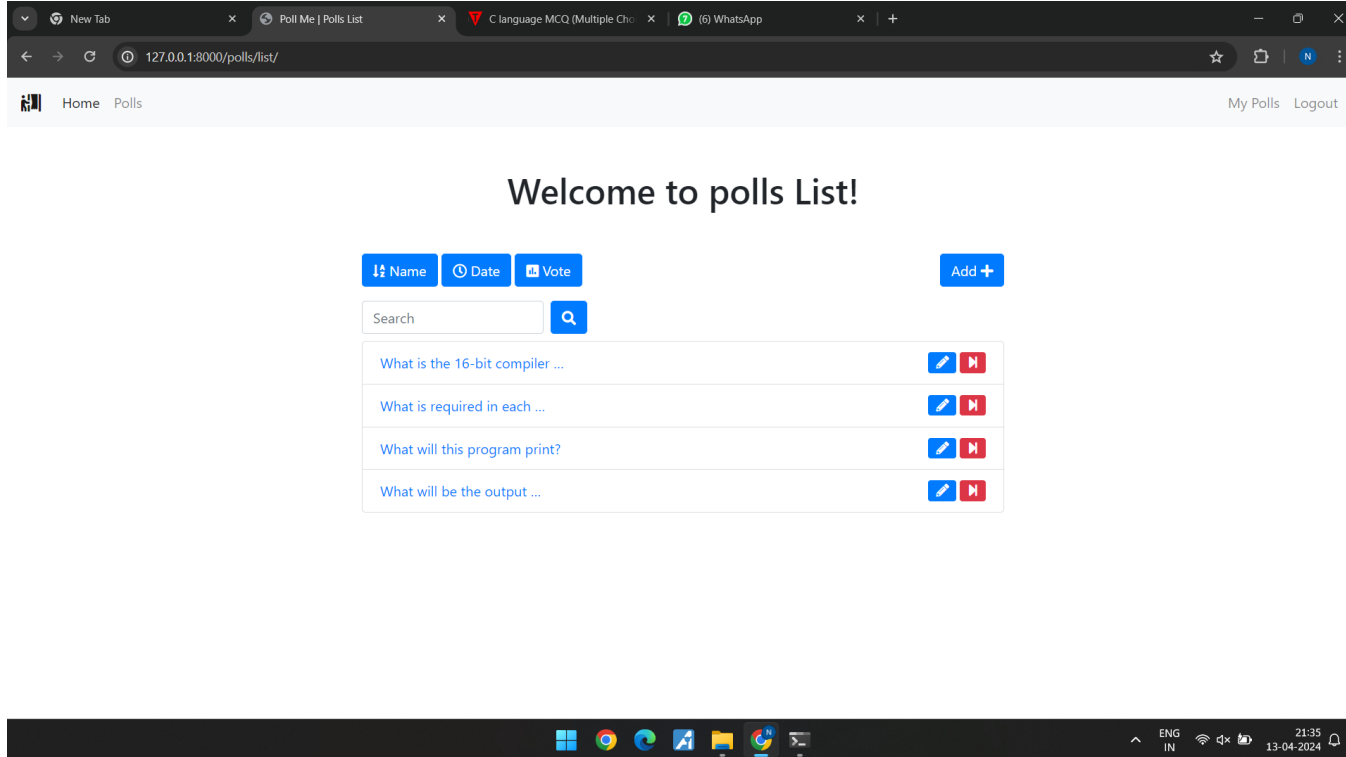
The proposed solution for a voting application using the Django framework is to create a secure and user-friendly online voting platform. The application will allow users to register, vote, and view real-time results. To build the application, the Django framework will be used as the foundation due to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



New Tab x Poll Me | Polls List x C language MCQ (Multiple Cho... x (6) WhatsApp x +

127.0.0.1:8000/polls/list/ ☆ 📄 N ⋮

Home Polls My Polls Logout

### Welcome to polls List!

🔼 Name 🕒 Date 🗳 Vote Add +

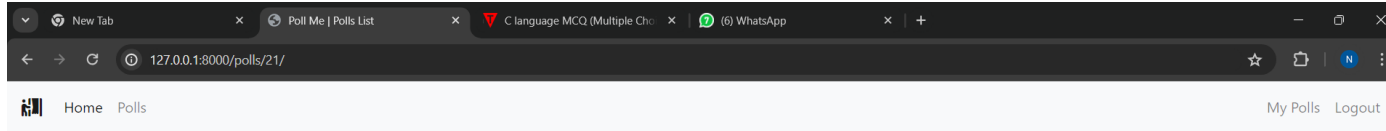
Search 🔍

What is the 16-bit compiler ...	✎ 🗑
What is required in each ...	✎ 🗑
What will this program print?	✎ 🗑
What will be the output ...	✎ 🗑

Windows taskbar: 21:35 13-04-2024



## Voting Page



### Polls details page

What is the 16-bit compiler allowable range for integer constants?

☐ 3.4e38 to 3.4e38

☒ -32767 to 3276

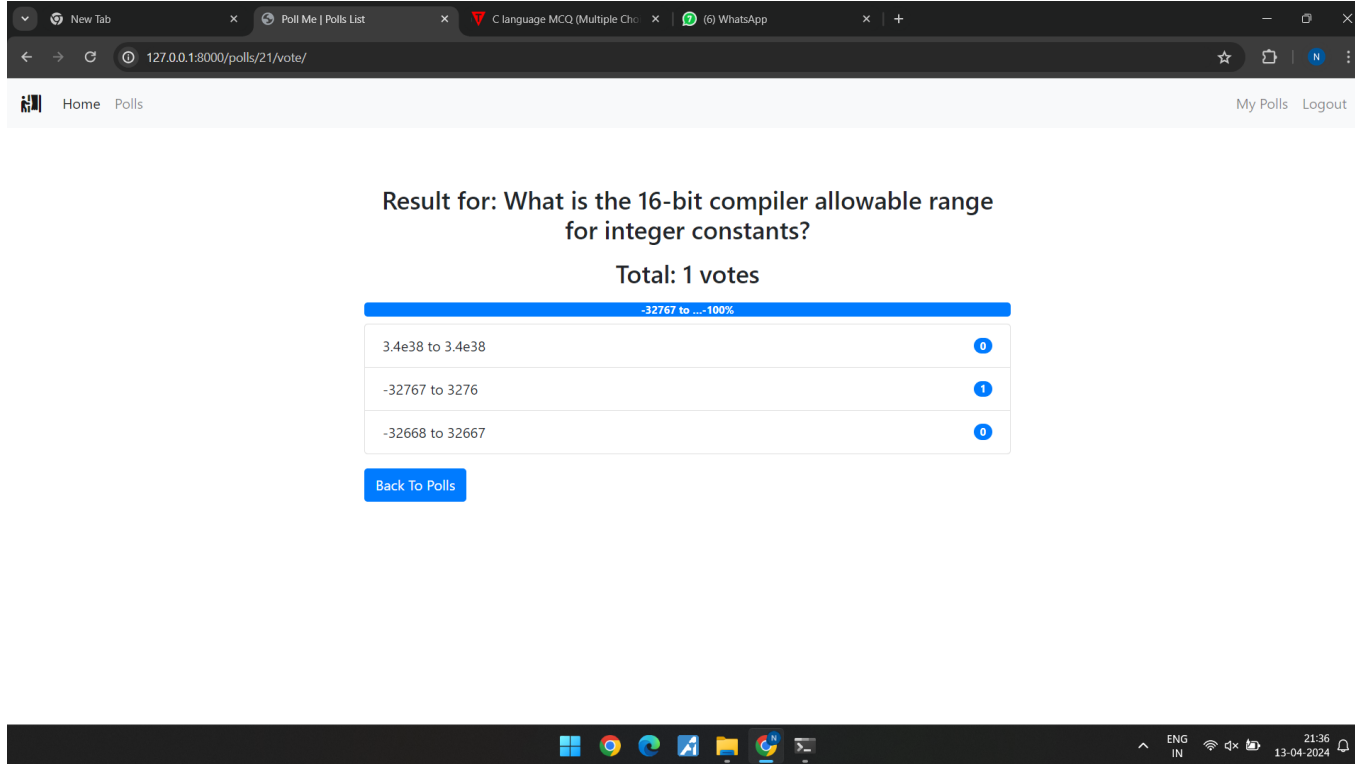
☐ -32668 to 32667

Vote

Cancel



## Voting Details Page



Browser tabs: New Tab, Poll Me | Polls List, C language MCQ (Multiple Cho..., (6) WhatsApp

Address bar: 127.0.0.1:8000/polls/21/vote/

Navigation: Home Polls My Polls Logout

Result for: What is the 16-bit compiler allowable range for integer constants?

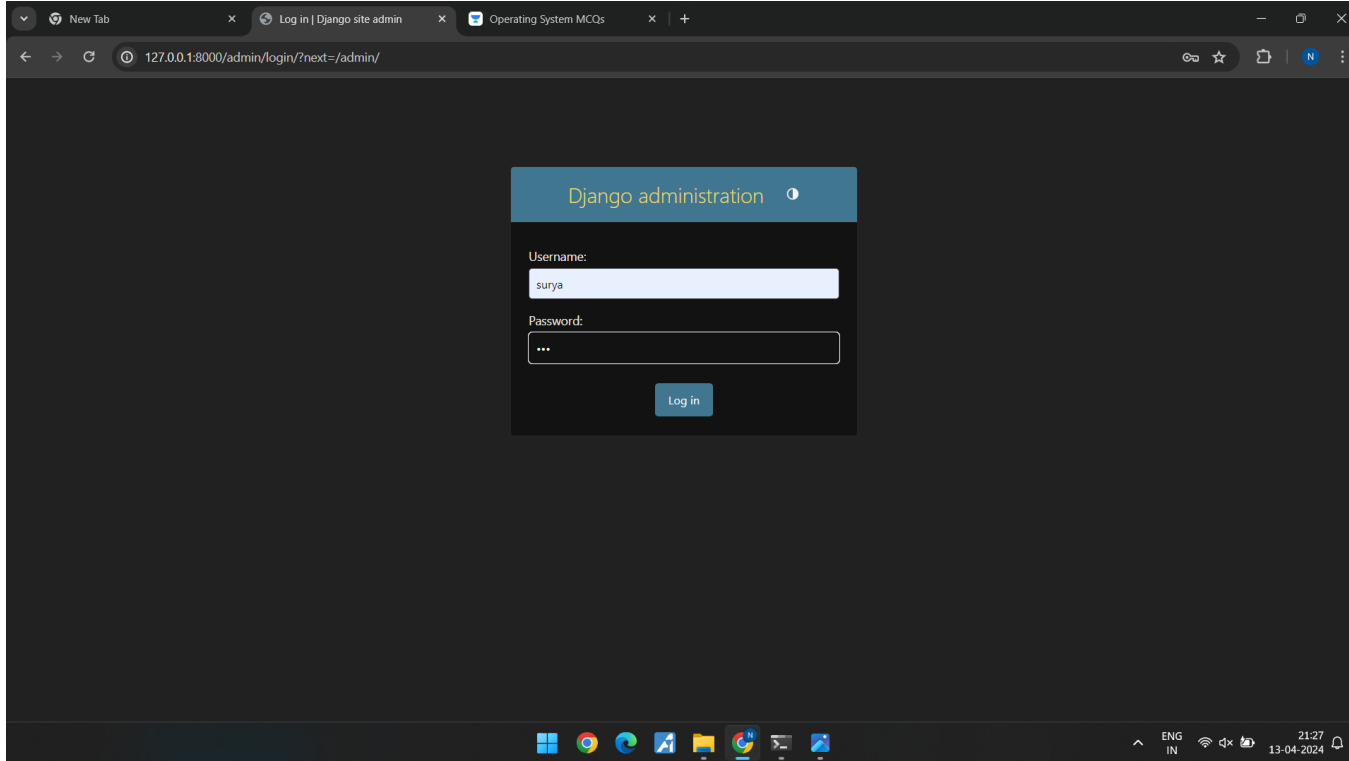
Total: 1 votes

-32767 to ...-100%	
3.4e38 to 3.4e38	0
-32767 to 3276	1
-32668 to 32667	0

Back To Polls

Windows taskbar: ENG IN, 21:36, 13-04-2024

## Admin Login Page



New Tab x Log in | Django site admin x Operating System MCQs x +

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

Django administration

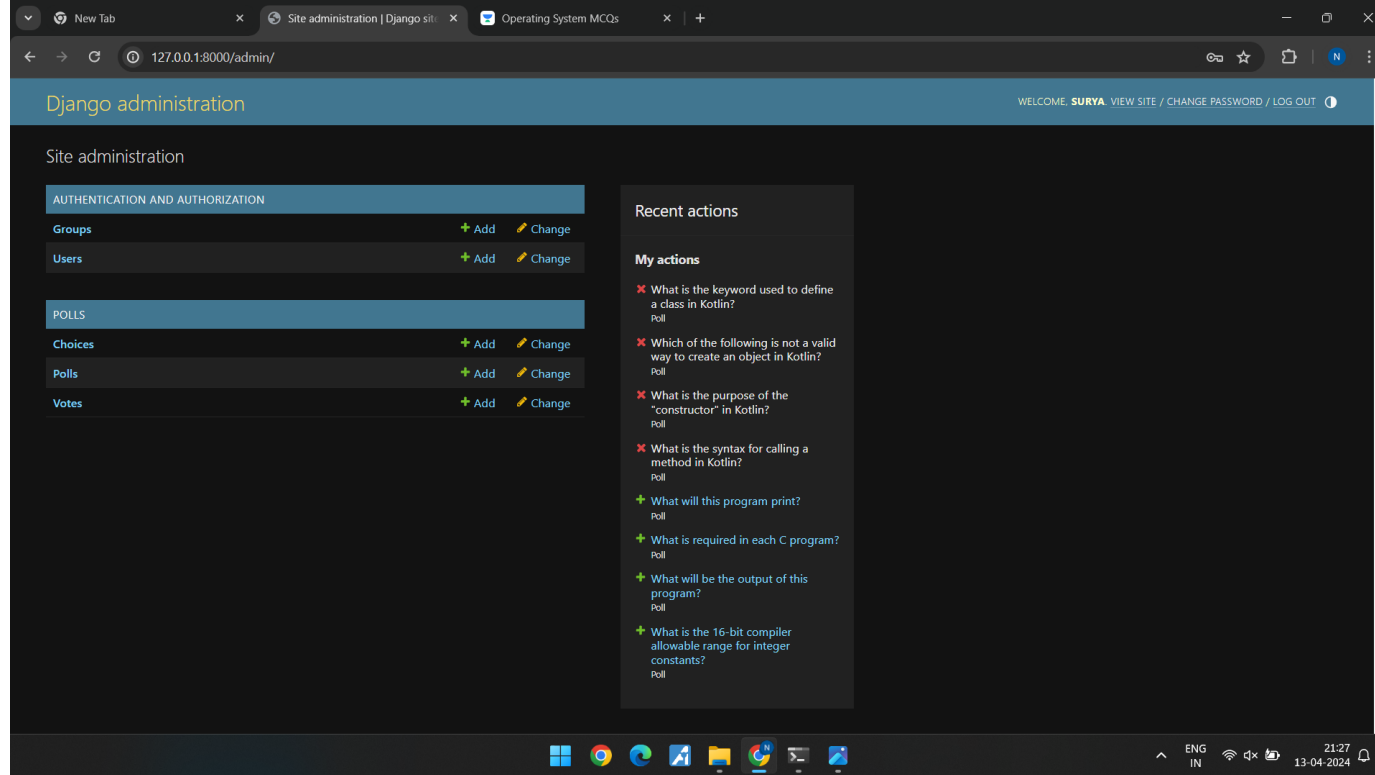
Username:  
surya

Password:  
...

Log in

ENG IN 21:27 13-04-2024

## Admin Home Page



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

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

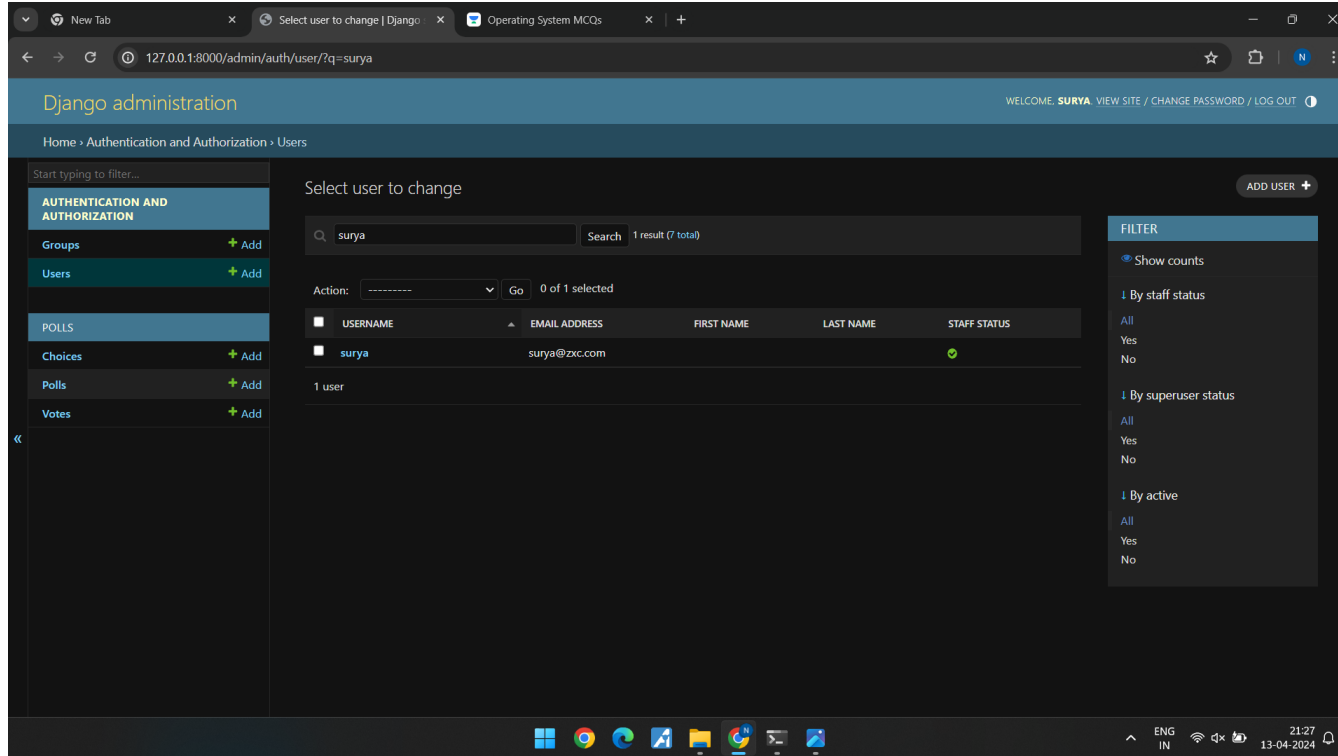
- AUTHENTICATION AND AUTHORIZATION**: This section contains two items, "Groups" and "Users", each with a green "+ Add" button and a yellow pencil "Change" icon.
- POLLS**: This section contains three items, "Choices", "Polls", and "Votes", each with a green "+ Add" button and a yellow pencil "Change" icon.

On the right side of the page, there is a "Recent actions" sidebar. It features a "My actions" header and a list of actions, each preceded by a red "X" icon. The actions listed are:

- What is the keyword used to define a class in Kotlin? Poll
- Which of the following is not a valid way to create an object in Kotlin? Poll
- What is the purpose of the "constructor" in Kotlin? Poll
- What is the syntax for calling a method in Kotlin? Poll
- What will this program print? Poll
- What is required in each C program? Poll
- What will be the output of this program? Poll
- What is the 16-bit compiler allowable range for integer constants? Poll

The bottom of the image shows a Windows taskbar with various application icons and a system tray on the right displaying the language as "ENG IN", the time as "21:27", and the date as "13-04-2024".

## 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/?q=surya`. The page header includes the Django logo, the text 'Django administration', and a welcome message for 'SURYA' 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 user to change'. It features a search bar with the text 'surya' and a 'Search' button, indicating '1 result (7 total)'. Below the search bar, there is an 'Action:' dropdown menu and a 'Go' button, with the text '0 of 1 selected'.

A table displays the search results:

USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
surya	surya@zxc.com			✓

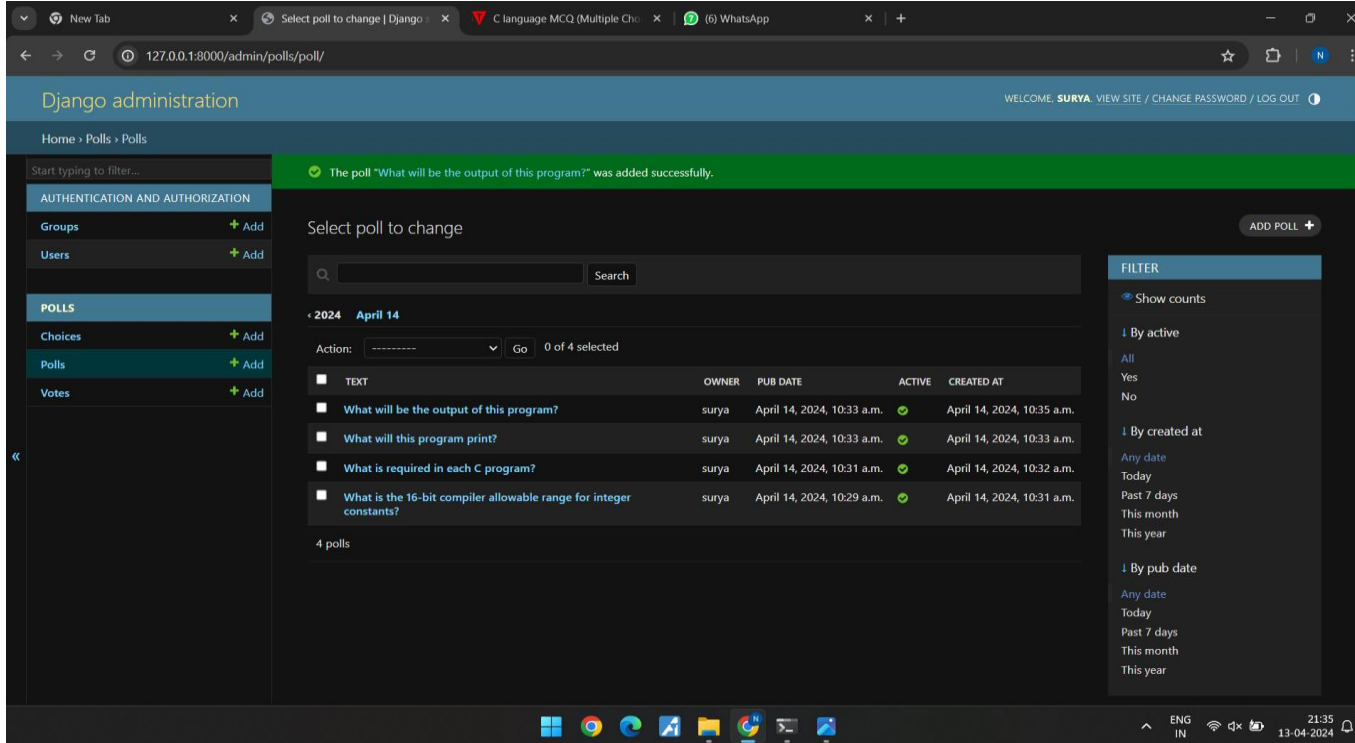
Below the table, it states '1 user'.

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

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

The Windows taskbar at the bottom shows the system clock as 21:27 on 13-04-2024, along with various system icons.

## Questions Adding Section Page



Django administration

WELCOME, SURYA. [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](#)

**Success Message:** The poll "What will be the output of this program?" was added successfully.

Select poll to change

Search

2024 April 14

Action:  Go 0 of 4 selected

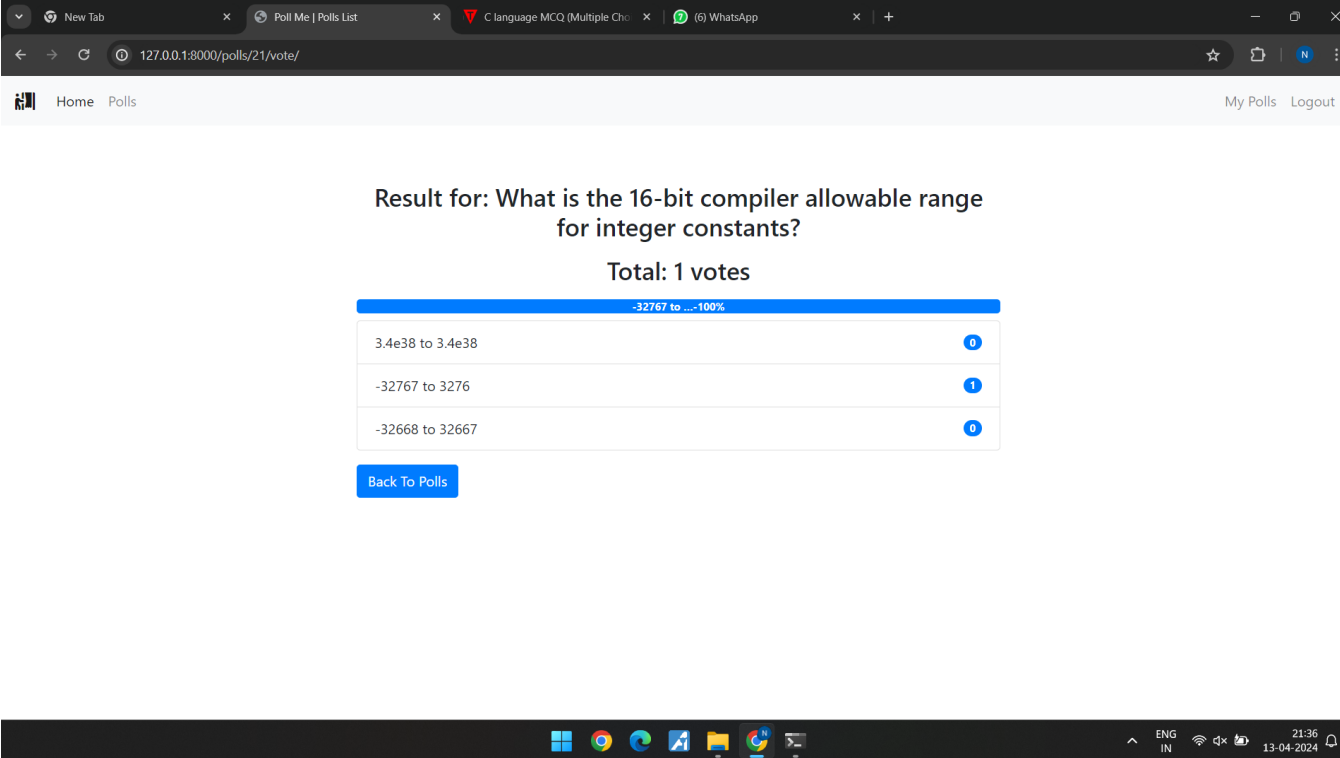
<input type="checkbox"/>	TEXT	OWNER	PUB DATE	ACTIVE	CREATED AT
<input type="checkbox"/>	What will be the output of this program?	surya	April 14, 2024, 10:33 a.m.	✓	April 14, 2024, 10:35 a.m.
<input type="checkbox"/>	What will this program print?	surya	April 14, 2024, 10:33 a.m.	✓	April 14, 2024, 10:33 a.m.
<input type="checkbox"/>	What is required in each C program?	surya	April 14, 2024, 10:31 a.m.	✓	April 14, 2024, 10:32 a.m.
<input type="checkbox"/>	What is the 16-bit compiler allowable range for integer constants?	surya	April 14, 2024, 10:29 a.m.	✓	April 14, 2024, 10:31 a.m.

4 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

## Voting Details Page



The screenshot shows a web browser window with the following details:

- Browser Tabs:** New Tab, Poll Me | Polls List, C language MCQ (Multiple Cho..., (6) WhatsApp.
- Address Bar:** 127.0.0.1:8000/polls/21/vote/
- Navigation:** Home, Polls, My Polls, Logout.
- Question:** Result for: What is the 16-bit compiler allowable range for integer constants?
- Total:** 1 votes
- Options and Votes:**

Option	Votes
3.4e38 to 3.4e38	0
-32767 to 3276	1
-32668 to 32667	0

[Back To Polls](#)

The Windows taskbar at the bottom shows the date and time as 13-04-2024, 21:36, along with system icons for language (ENG IN), network, and volume.

## 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!**