



NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name :Naveenkumar A S
Student ID :au820621104057

College Name

Arasu Engineering College

CAPSTONE PROJECT SHOWCASE

Project Title

Voting Application using Django Framework-Naveenkumar(4057,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:

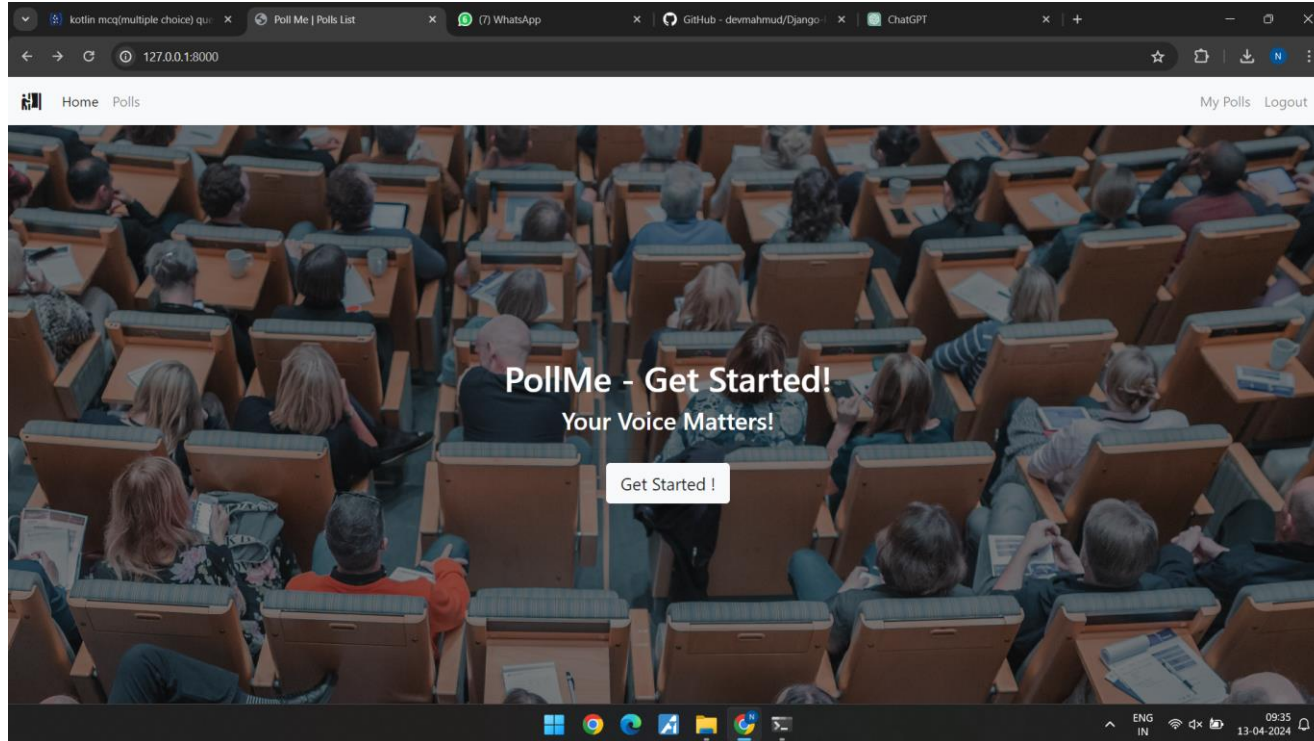
- 1.Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2.Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3.Creating User Authentication:** Implement user authentication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

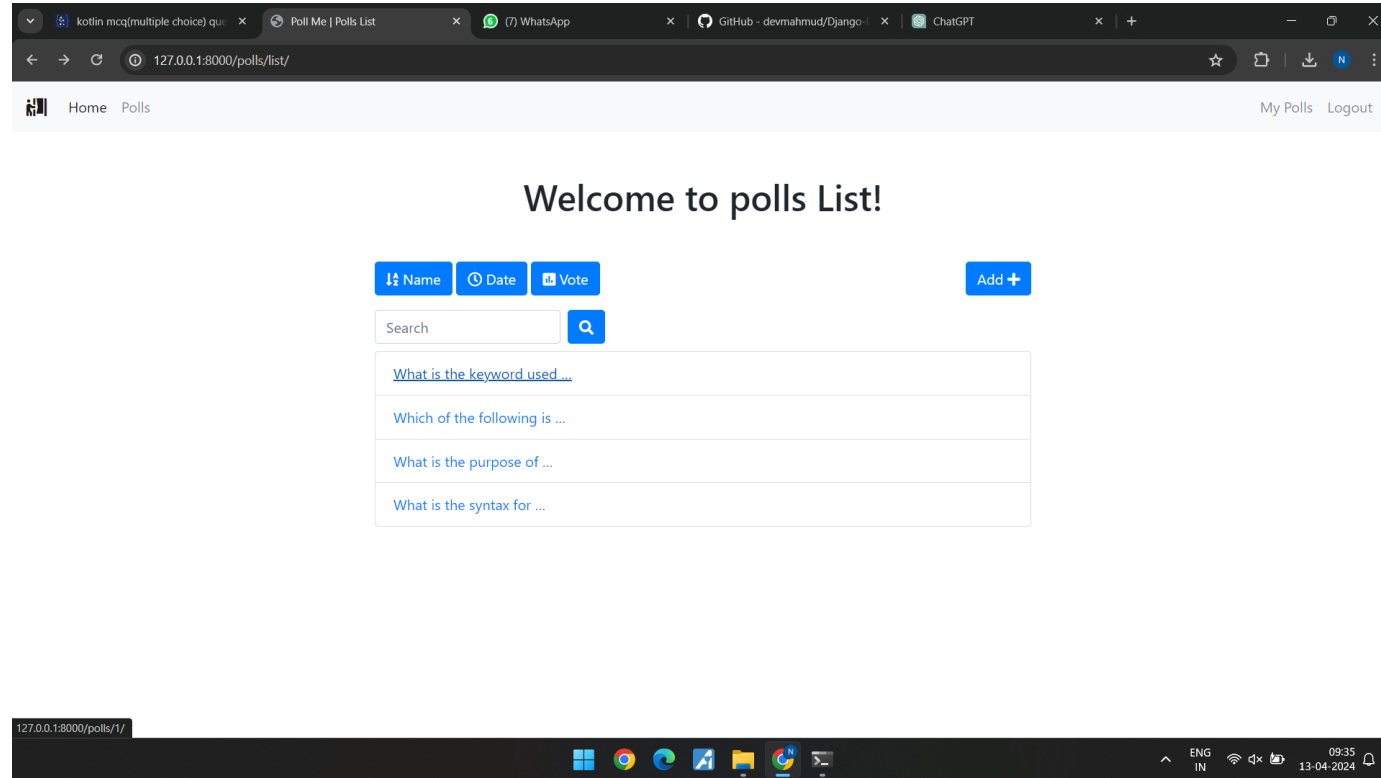
The proposed solution for a voting application using the Django framework is to create a secure and user-friendly online voting platform. The application will allow users to register, vote, and view real-time results. To build the application, the Django framework will be used as the foundation due to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

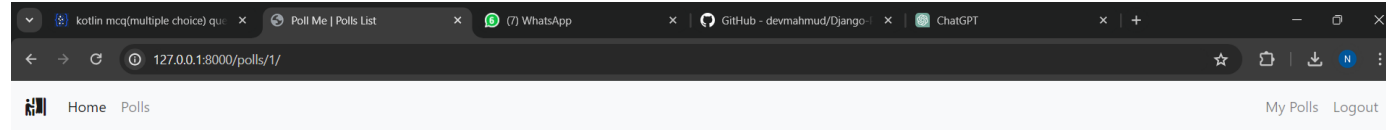


The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/polls/list/". The browser has several tabs open, including "kotlin mcq(multiple choice) qu...", "Poll Me | Polls List", "(7) WhatsApp", "GitHub - devmahmud/Django-", and "ChatGPT". The page features a navigation bar with "Home" and "Polls" links, and "My Polls" and "Logout" links on the right. The main content area displays a "Welcome to polls List!" message. Below this, there are four buttons: "Name" (with a person icon), "Date" (with a clock icon), "Vote" (with a ballot icon), and "Add +" (with a plus icon). A search bar with a magnifying glass icon is also present. Below the search bar, there is a list of four poll questions, each in a separate row:

- [What is the keyword used ...](#)
- [Which of the following is ...](#)
- [What is the purpose of ...](#)
- [What is the syntax for ...](#)

The bottom of the screenshot shows a Windows taskbar with various application icons and a system tray displaying "ENG IN", "13-04-2024", and "09:35".

Voting Page



Polls details page

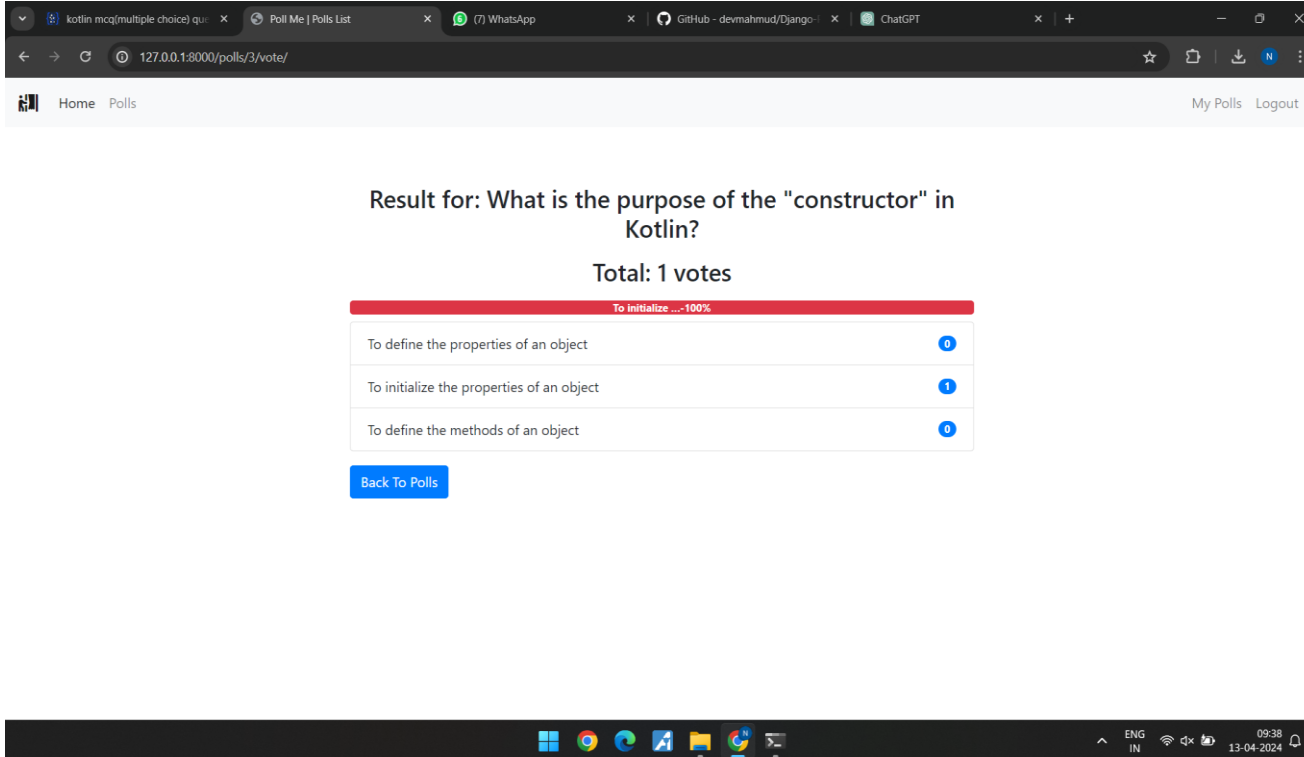
What is the keyword used to define a class in Kotlin?

- ☐ class
- ☐ struct
- ☐ object

[Vote](#)

[Cancel](#)

Voting Details Page



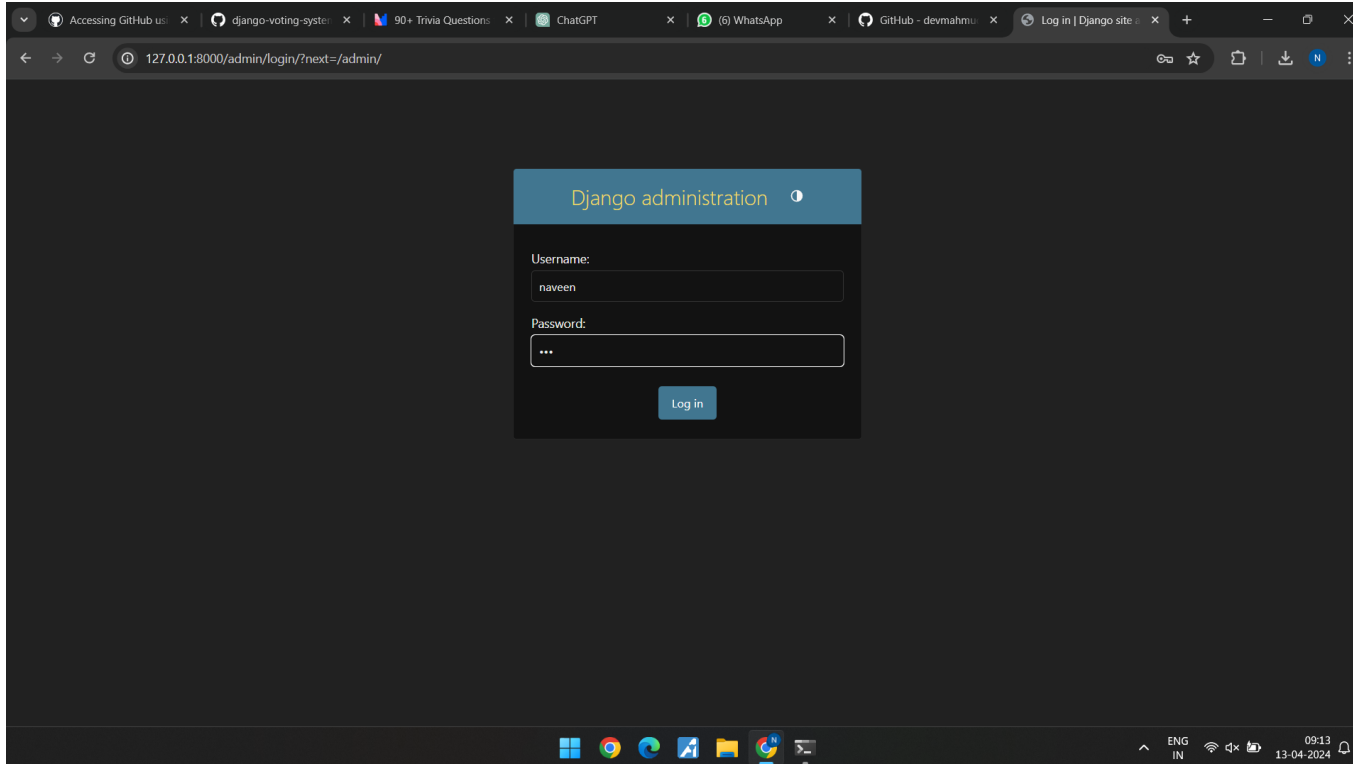
The screenshot shows a web browser with multiple tabs open. The active tab is 'Poll Me | Polls List' with the URL '127.0.0.1:8000/polls/3/vote/'. The page displays the result for a poll titled 'What is the purpose of the "constructor" in Kotlin?'. The total number of votes is 1. A progress bar indicates 'To initialize ...100%'. The poll options and their respective vote counts are as follows:

Option	Votes
To define the properties of an object	0
To initialize the properties of an object	1
To define the methods of an object	0

Below the table is a blue button labeled 'Back To Polls'.

The Windows taskbar at the bottom shows the system clock as 09:38 on 13-04-2024, with the language set to ENG IN.

Admin Login Page



Accessing GitHub us... x | django-voting-syste... x | 90+ Trivia Questions... x | ChatGPT... x | (6) WhatsApp... x | GitHub - devmahm... x | Log in | Django site... x

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

Django administration ⓘ

Username:

naveen

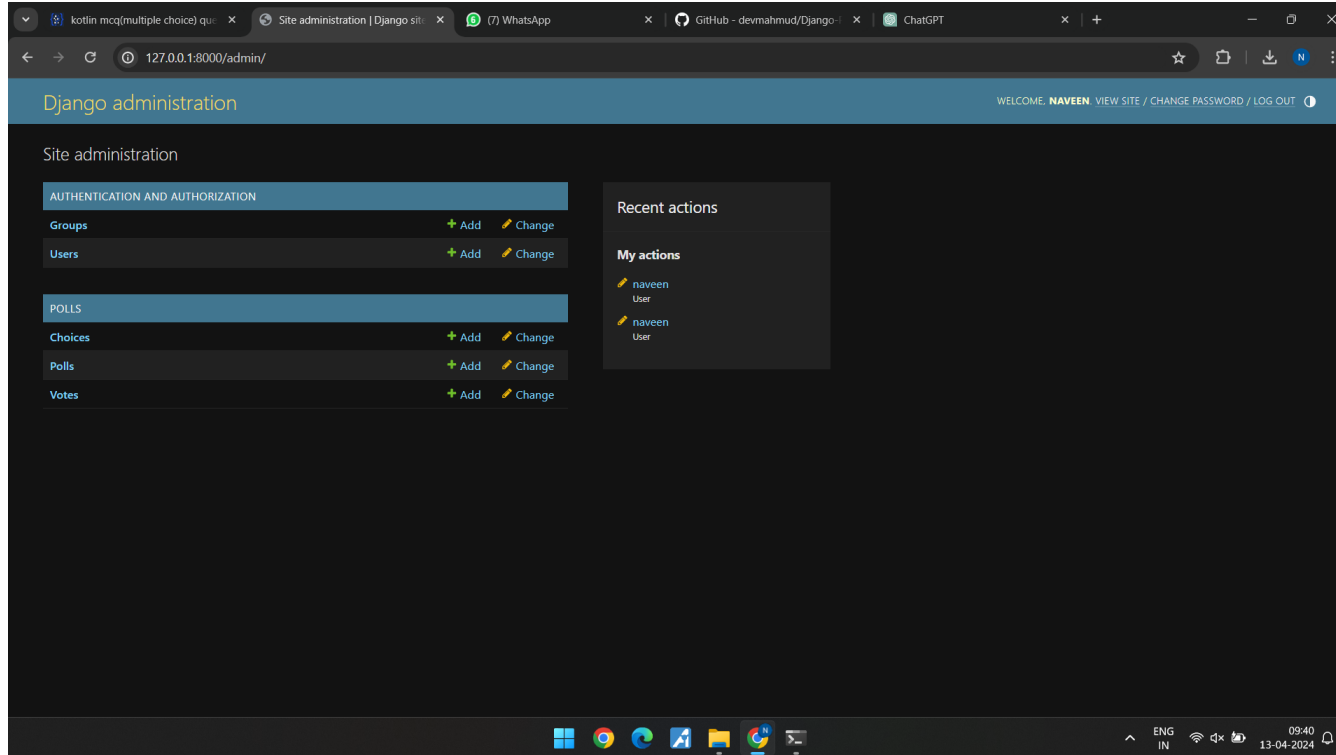
Password:

...

Log in

Windows taskbar: ENG IN, 09:13, 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 title is "Django administration". A welcome message at the top right reads "WELCOME, **NAVEEN** [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)".

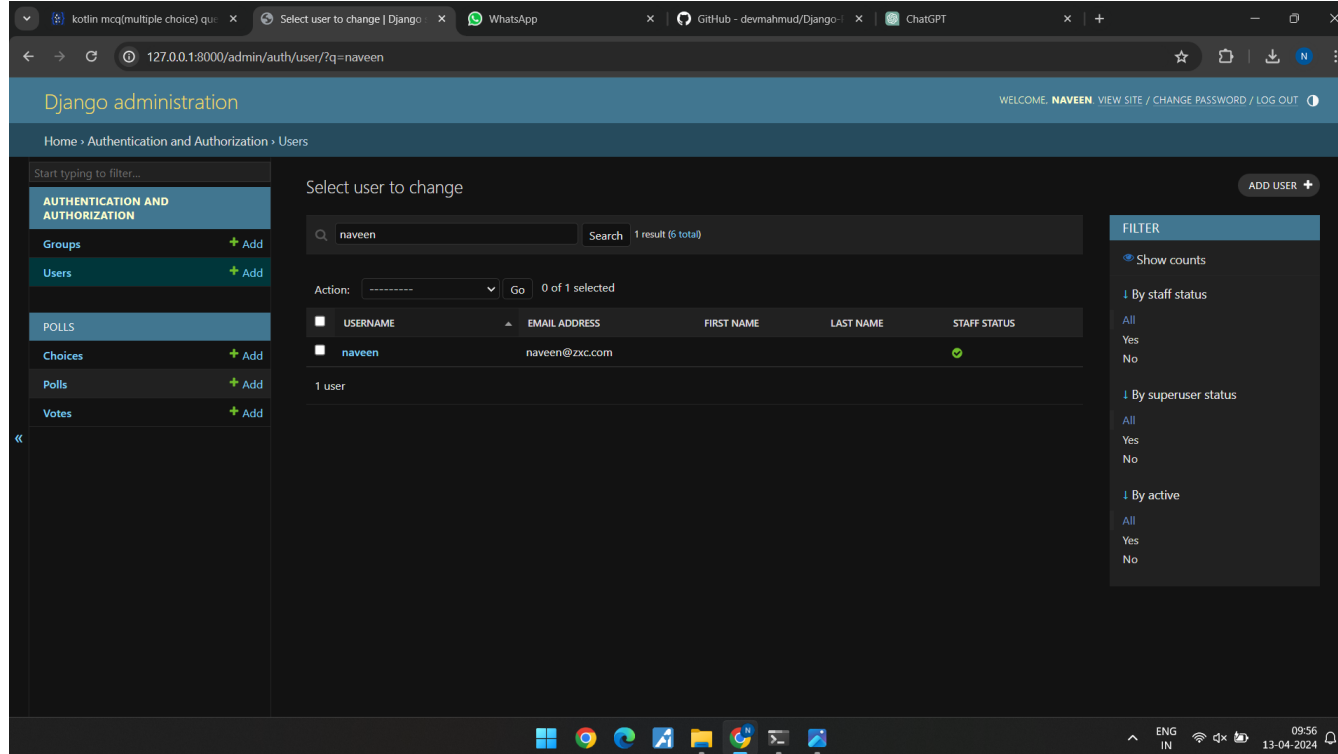
The main content area is titled "Site administration" and is divided into two columns. The left column contains two sections: "AUTHENTICATION AND AUTHORIZATION" and "POLLS".

- Authentication and Authorization:** Includes links for "Groups" and "Users", each with "Add" and "Change" options.
- Polls:** Includes links for "Choices", "Polls", and "Votes", each with "Add" and "Change" options.

The right column contains a "Recent actions" section and a "My actions" section. The "Recent actions" section lists two actions: "naveen User" and "naveen User", each with a "Change" link. The "My actions" section is currently empty.

The Windows taskbar at the bottom shows the time as 09:40 on 13-04-2024, along with system icons for language (ENG IN), network, and volume.

Authentication and Authorization Page



Django administration

WELCOME, **NAVEEN** 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: naveen 1 result (6 total)

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

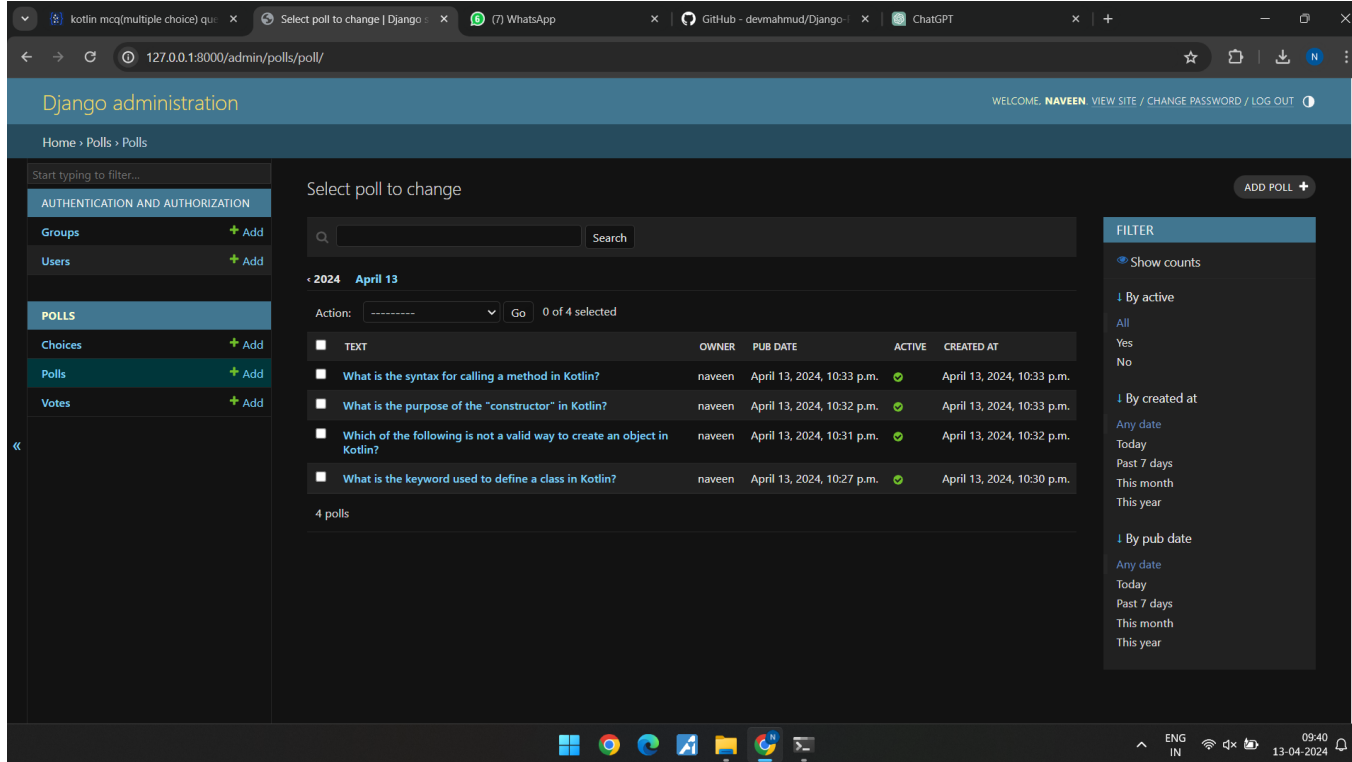
<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	naveen	naveen@zxc.com			✓

1 user

FILTER

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

Questions Adding Section Page



Django administration

WELCOME, NAVEEN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Polls » Polls

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

POLL'S

Choices + Add

Polls + Add

Votes + Add

Select poll to change

Search

2024 April 13

Action: Go 0 of 4 selected

TEXT	OWNER	PUB DATE	ACTIVE	CREATED AT
<input type="checkbox"/> What is the syntax for calling a method in Kotlin?	naveen	April 13, 2024, 10:33 p.m.	✓	April 13, 2024, 10:33 p.m.
<input type="checkbox"/> What is the purpose of the "constructor" in Kotlin?	naveen	April 13, 2024, 10:32 p.m.	✓	April 13, 2024, 10:33 p.m.
<input type="checkbox"/> Which of the following is not a valid way to create an object in Kotlin?	naveen	April 13, 2024, 10:31 p.m.	✓	April 13, 2024, 10:32 p.m.
<input type="checkbox"/> What is the keyword used to define a class in Kotlin?	naveen	April 13, 2024, 10:27 p.m.	✓	April 13, 2024, 10:30 p.m.

4 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

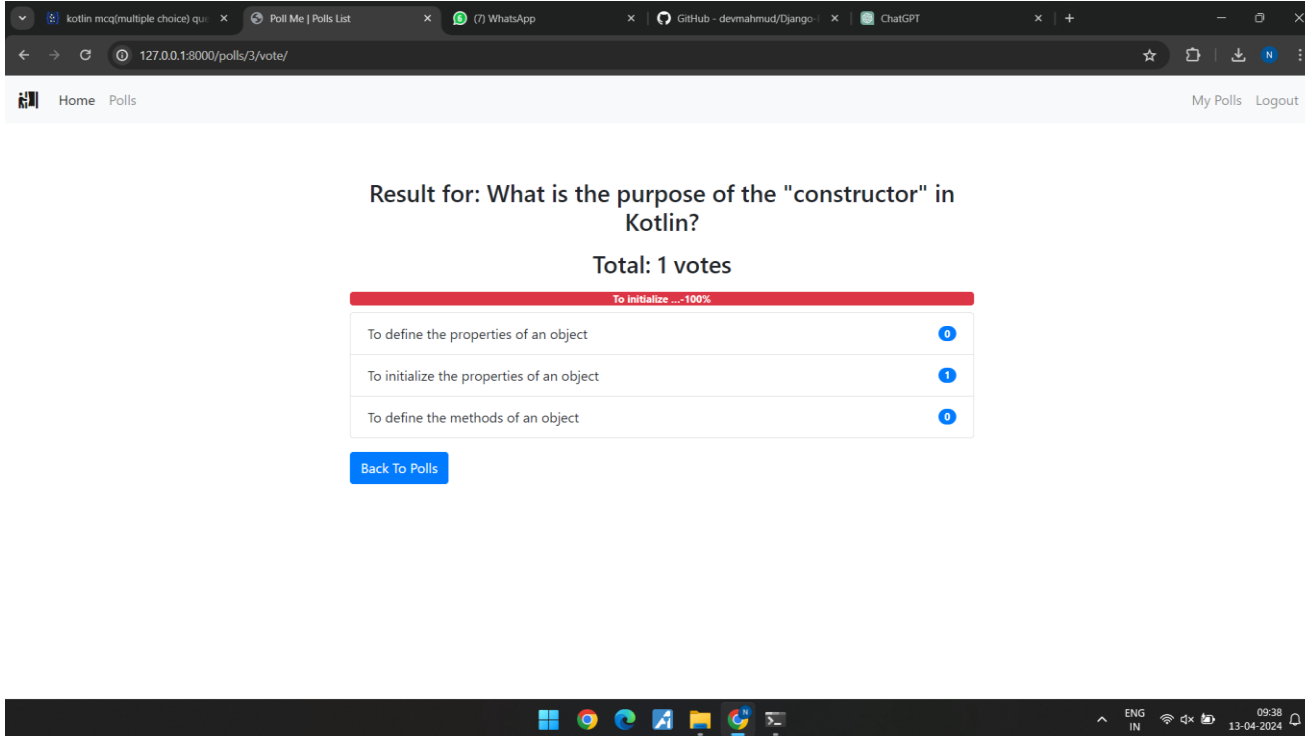
Today

Past 7 days

This month

This year

Voting Details Page



The screenshot shows a web browser window with multiple tabs. The active tab is 'Poll Me | Polls List'. The address bar shows the URL '127.0.0.1:8000/polls/3/vote/'. The page has a navigation bar with 'Home' and 'Polls' links, and 'My Polls' and 'Logout' links on the right. The main content area displays the poll result for the question 'What is the purpose of the "constructor" in Kotlin?'. It shows 'Total: 1 votes' and a progress bar indicating 'To initialize100%'. Below the progress bar, there is a table with three options and their respective vote counts:

To initialize100%	
To define the properties of an object	0
To initialize the properties of an object	1
To define the methods of an object	0

At the bottom of the table, there is a blue button labeled 'Back To Polls'.

The Windows taskbar at the bottom shows the time as 09:38 on 13-04-2024, with the language set to ENG IN.

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!