



# NEXT GEN EMPLOYABILITY PROGRAM

| Creating a future-ready workforce

Student Name : AROCKIA LIVIN H  
Student ID : autb21csl002

College Name

Arasu Engineering College

# CAPSTONE PROJECT SHOWCASE

## Project Title

Django Vote software- AROCKIA LIVIN(4301,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:

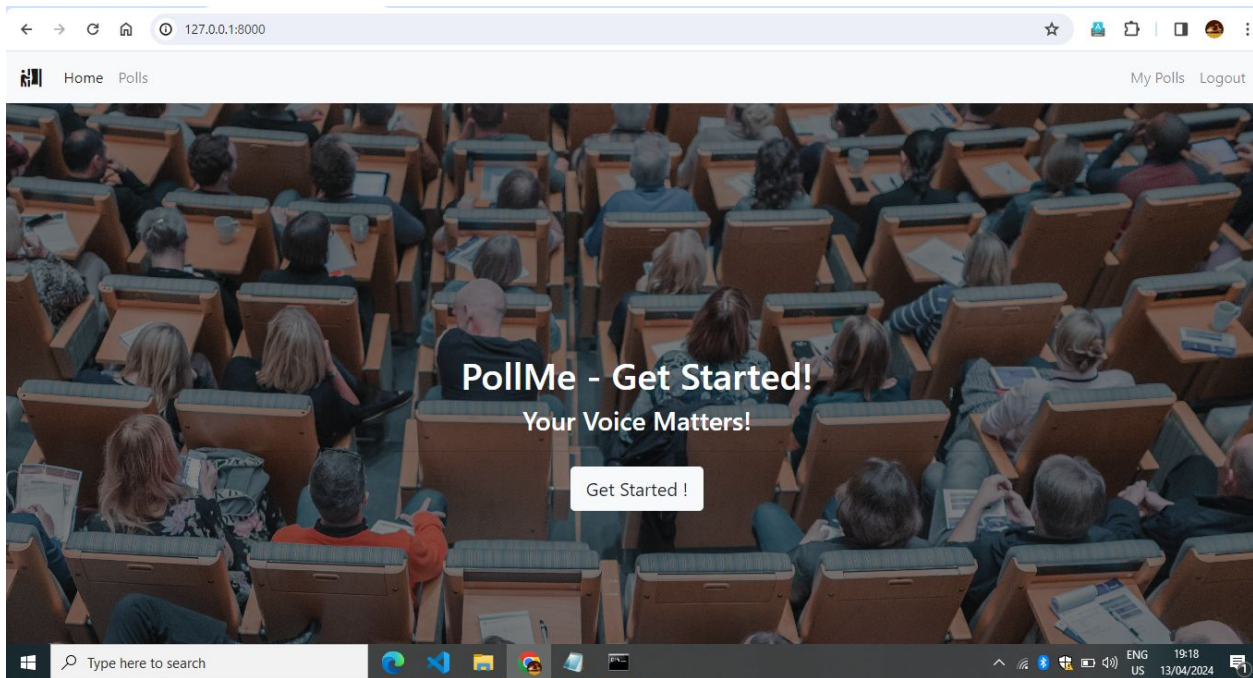
- 1. Setting up a Django Project:** Create a Django project to serve as the foundation for the voting application.
- 2. Designing the Database Schema:** Define the database structure to store user information, votes, and other relevant data.
- 3. Creating User Authentication:** Implement user authentication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

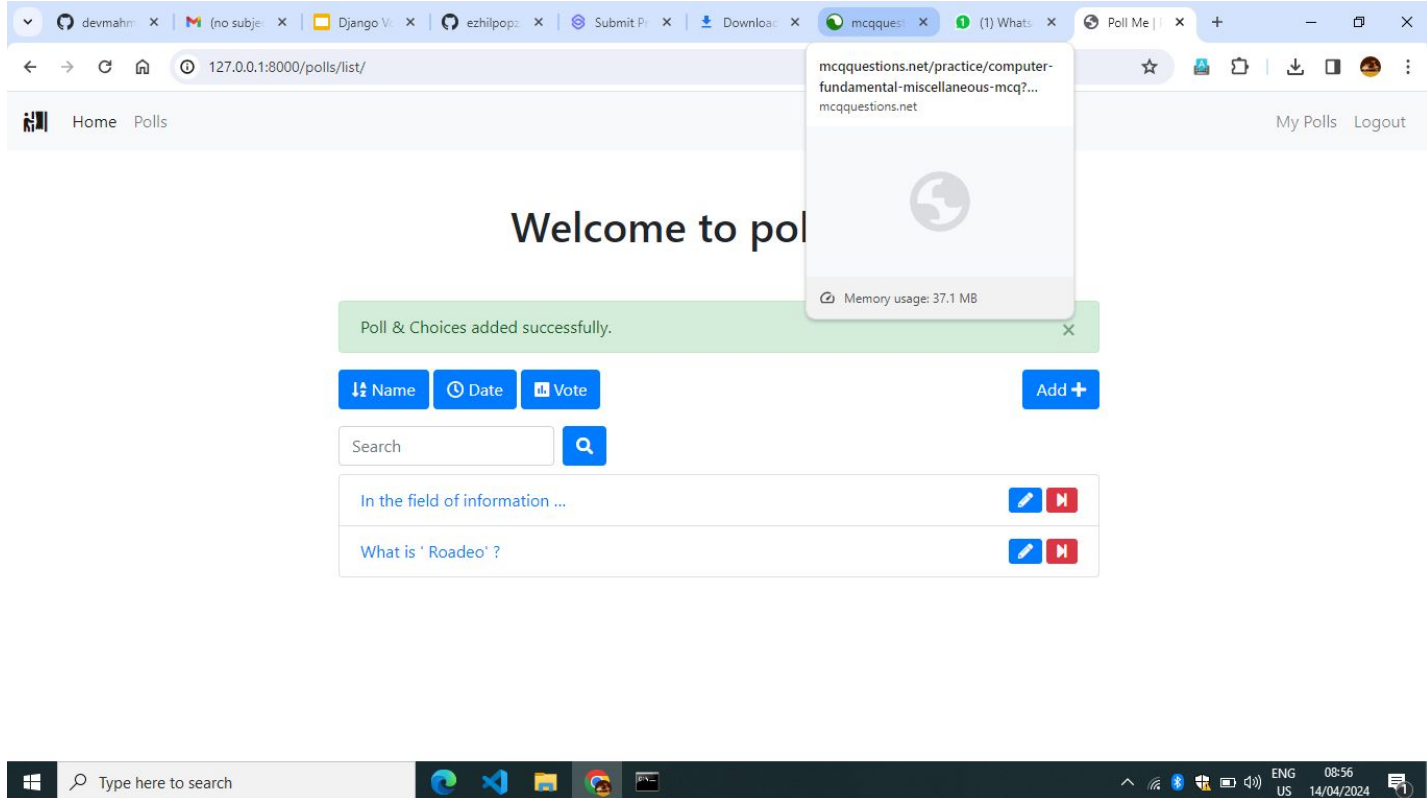
The proposed solution for a voting application using the Django framework is to create a secure and user-friendly online voting platform. The application will allow users to register, vote, and view real-time results. To build the application, the Django framework will be used as the foundation due to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



devmahm... (no subje... Django V... ezhilpop... Submit P... Download... mcquest... (1) Whats... Poll Me... + -

127.0.0.1:8000/polls/list/

Home Polls

Welcome to poll

My Polls Logout

Poll & Choices added successfully.

Name Date Vote Add +

Search

In the field of information ...

What is 'Rodeo' ?

mcquestions.net/practice/computer-fundamental-miscellaneous-mcq?... mcquestions.net

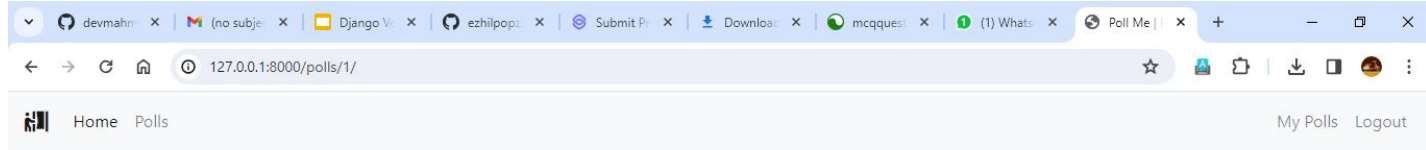
Memory usage: 37.1 MB

Type here to search

ENG US 08:56 14/04/2024



## Voting Page



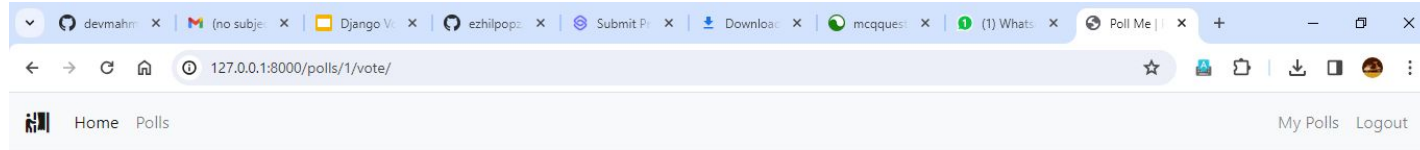
### Polls details page

In the field of information and communication technology, what is the full form of FDD ?

- ☐ Floppy Disk Drive
- ☐ Floppy Data Drive

[Vote](#) [Cancel](#)

## Voting Details Page



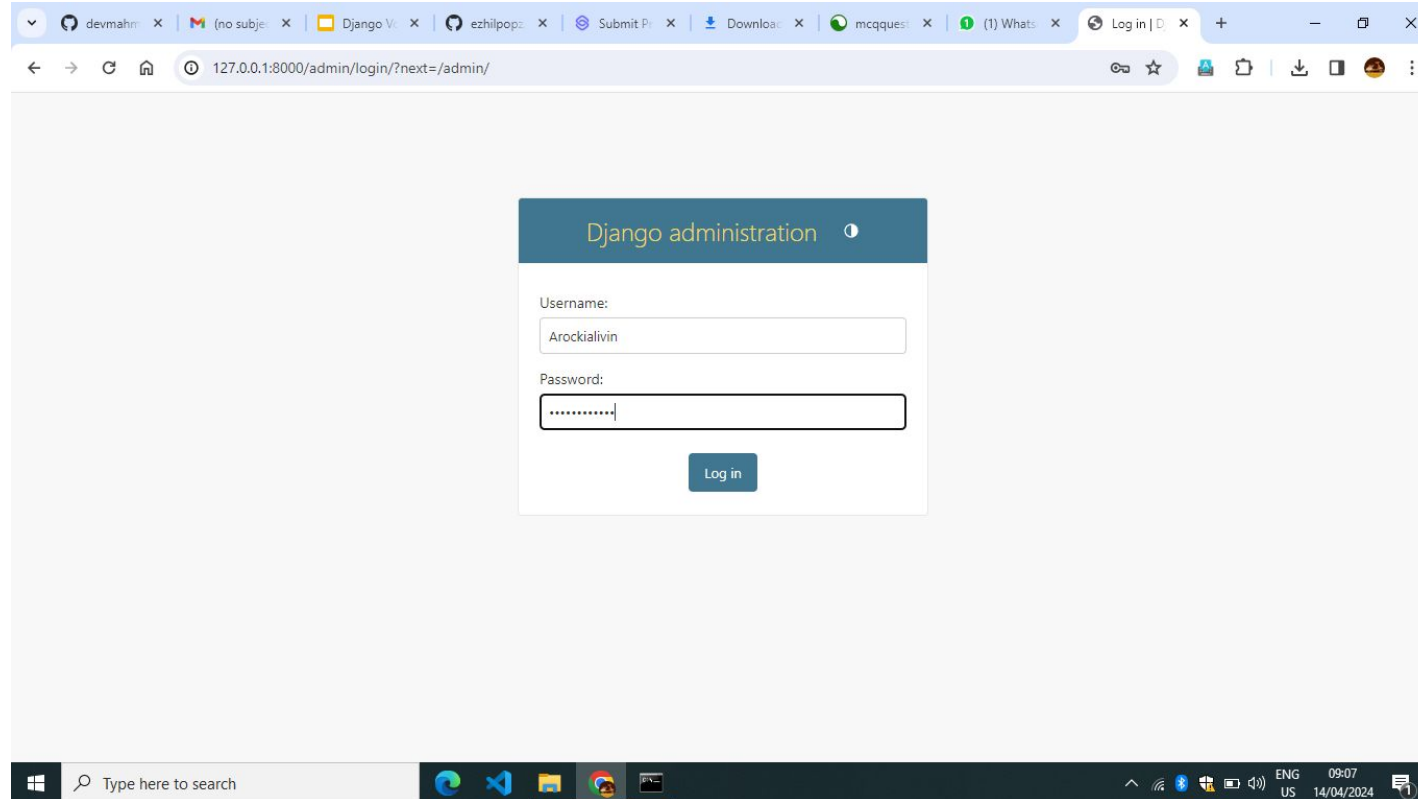
Result for: In the field of information and communication technology, what is the full form of FDD ?

Total: 1 votes

Floppy Disk ...-100%	
Floppy Disk Drive	1
Floppy Data Drive	0

[Back To Polls](#)

## Admin home



devmahin x (no subject) x Django V... x ezhilpop... x Submit P... x Download... x mcquest... x (1) Whats... x Log in | D... x

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

Django administration ⓘ

Username:

Arockialivin

Password:

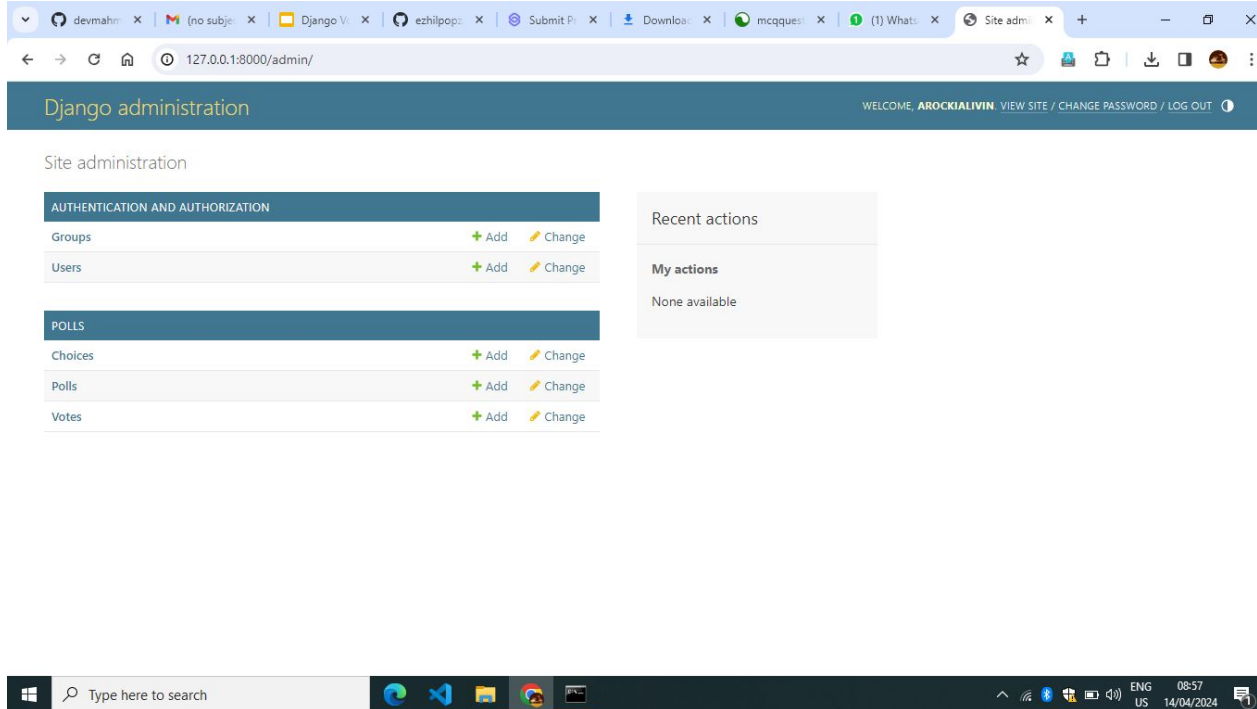
.....

Log in

Type here to search

ENG US 09:07 14/04/2024

## Admin Home Page



The screenshot shows a web browser window with multiple tabs. The active tab is 'Site admin', and the address bar shows '127.0.0.1:8000/admin/'. The page title is 'Django administration'. A welcome message for 'AROCKIALIVIN' is displayed, along with links for 'VIEW SITE', 'CHANGE PASSWORD', and 'LOG OUT'. The main content area is titled 'Site administration' and contains two sections: 'AUTHENTICATION AND AUTHORIZATION' and 'POLLS'. The 'AUTHENTICATION AND AUTHORIZATION' section has two rows: 'Groups' and 'Users', each with '+ Add' and 'Change' links. The 'POLLS' section has three rows: 'Choices', 'Polls', and 'Votes', each with '+ Add' and 'Change' links. On the right side, there is a 'Recent actions' section with a 'My actions' link and a message 'None available'. The Windows taskbar at the bottom shows the search bar, taskbar icons, and system tray with the date '14/04/2024' and time '08:57'.

devmahm x | (no subje x | Django V x | ezhilpop x | Submit P x | Download x | mcqques x | (1) Whats x | Site admin x

127.0.0.1:8000/admin/

Django administration

WELCOME, **AROCKIALIVIN** VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

**AUTHENTICATION AND AUTHORIZATION**

Groups	+ Add	Change
Users	+ Add	Change

**POLLS**

Choices	+ Add	Change
Polls	+ Add	Change
Votes	+ Add	Change

Recent actions

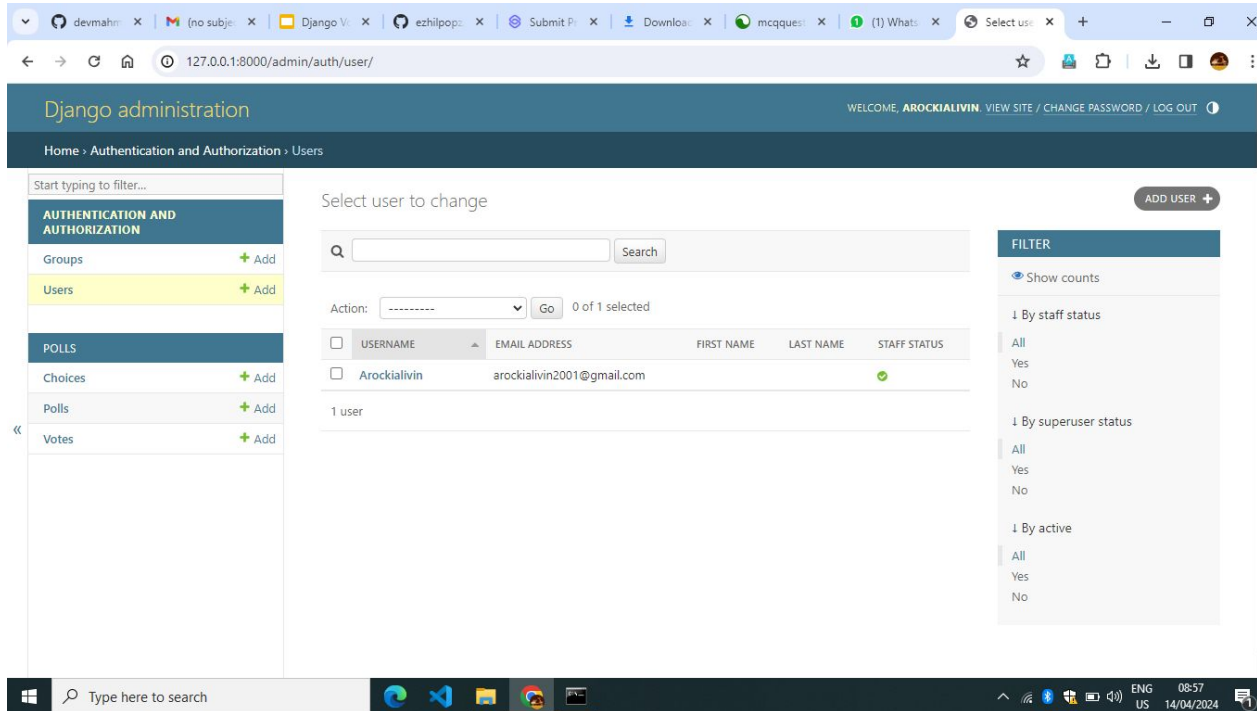
**My actions**

None available

Type here to search

ENG 08:57 14/04/2024

## Authentication and Authorization Page



The screenshot displays the Django administration interface for the 'Authentication and Authorization' section. The browser's address bar shows the URL '127.0.0.1:8000/admin/auth/user/'. The page header includes the Django logo and the text 'Django administration', along with a welcome message for 'AROCKIALVIN' and links for '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

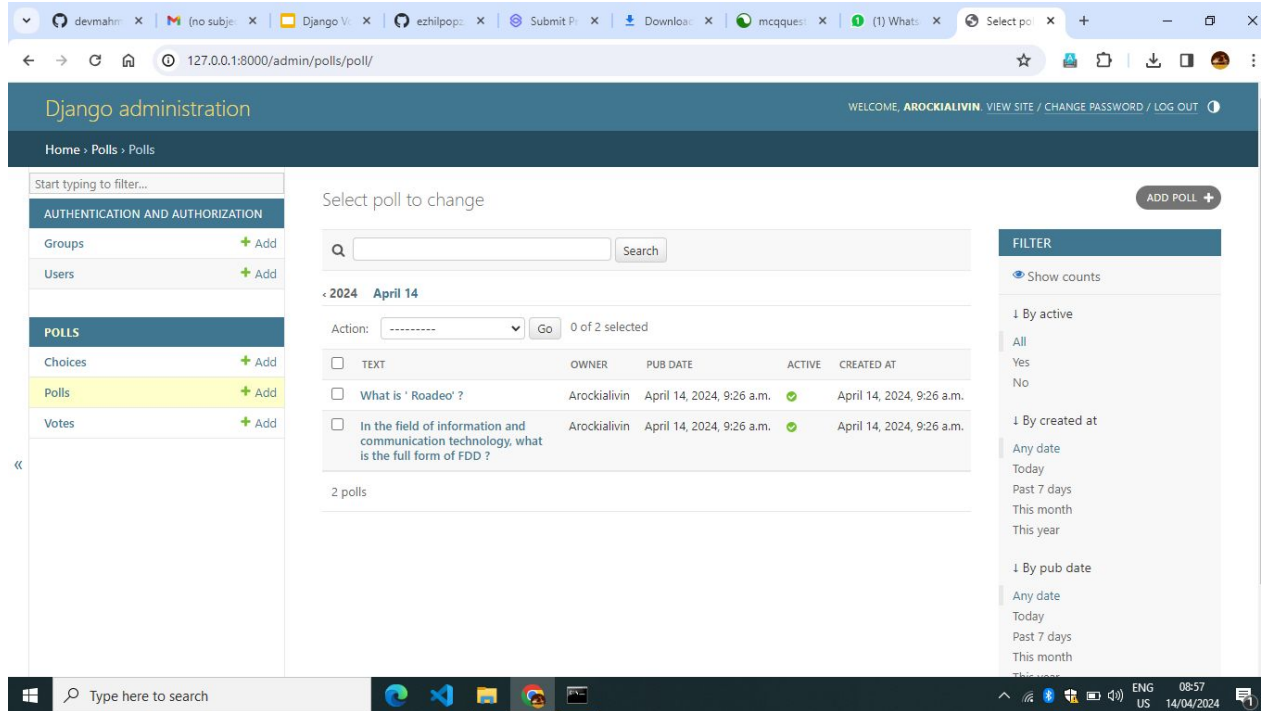
The main content area is titled 'Select user to change'. It features a search bar and a table of users. The table has columns for 'USERNAME', 'EMAIL ADDRESS', 'FIRST NAME', 'LAST NAME', and 'STAFF STATUS'. The user 'Arockialvin' is listed with the email 'arockialvin2001@gmail.com' and a staff status of 'Yes' (indicated by a green checkmark). 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 search bar, taskbar icons, and system tray information including the date '14/04/2024' and time '08:57'.

## Questions Adding Section Page



The screenshot displays the Django administration interface for the 'Polls' section. The browser address bar shows the URL '127.0.0.1:8000/admin/polls/poll/'. The page header includes the Django administration logo and a welcome message for 'AROCKIALIVIN' 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 (highlighted)
  - Votes + Add

The main content area is titled 'Select poll to change'. It features a search bar and a table of polls. The table has columns for 'TEXT', 'OWNER', 'PUB DATE', 'ACTIVE', and 'CREATED AT'. Two polls are listed:

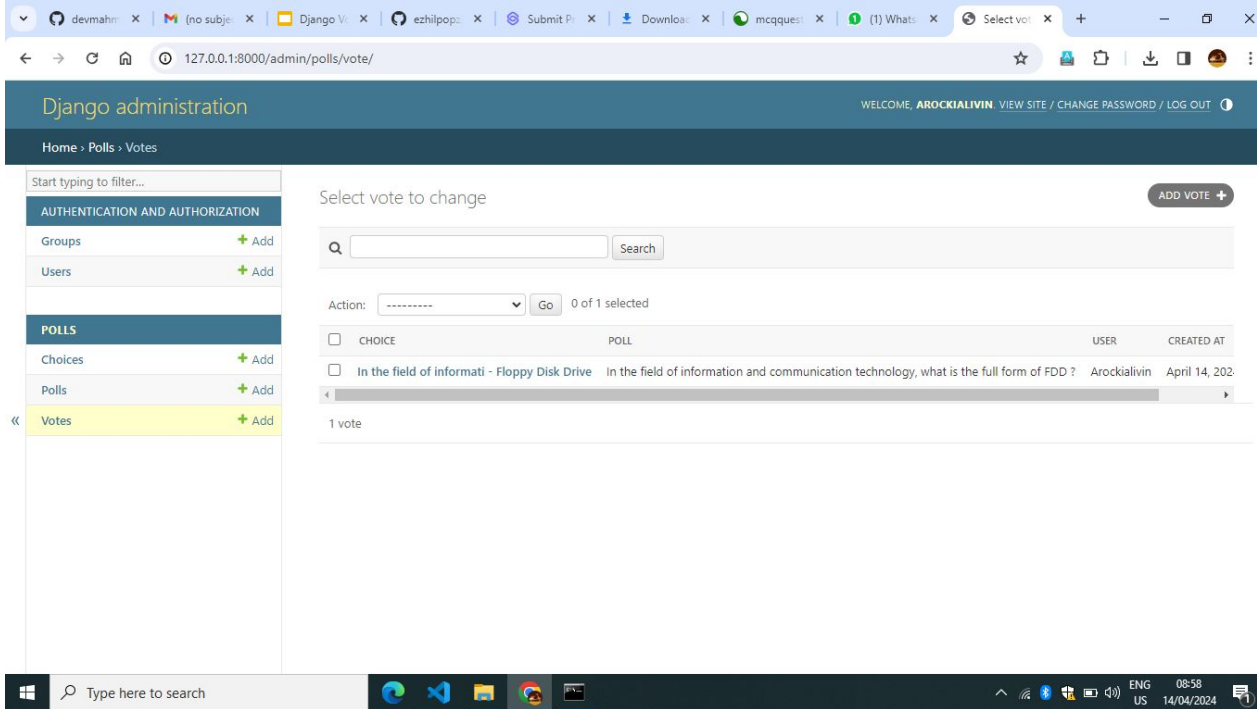
TEXT	OWNER	PUB DATE	ACTIVE	CREATED AT
<input type="checkbox"/> What is ' Roadeo' ?	Arockialivin	April 14, 2024, 9:26 a.m.	<input checked="" type="checkbox"/>	April 14, 2024, 9:26 a.m.
<input type="checkbox"/> In the field of information and communication technology, what is the full form of FDD ?	Arockialivin	April 14, 2024, 9:26 a.m.	<input checked="" type="checkbox"/>	April 14, 2024, 9:26 a.m.

Below the table, it indicates '2 polls'. On the right side, there is a 'FILTER' panel with the following options:

- 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

The Windows taskbar at the bottom shows the system clock as 08:57 on 14/04/2024.

## Voting Details Page



The screenshot displays the Django administration interface for the 'polls' app, specifically the 'Votes' page. The browser's address bar shows the URL '127.0.0.1:8000/admin/polls/vote/'. The page header includes the Django administration logo and a welcome message for 'AROCKIALIVIN' with links to 'VIEW SITE', 'CHANGE PASSWORD', and 'LOG OUT'. The left sidebar contains a navigation menu with categories: 'AUTHENTICATION AND AUTHORIZATION' (Groups, Users), 'POLLS' (Choices, Polls), and 'VOTES' (Votes). The main content area is titled 'Select vote to change' and features a search bar and an 'ADD VOTE +' button. Below the search bar, there is an 'Action:' dropdown menu and a 'Go' button. A table lists the available votes, with columns for 'CHOICE', 'POLL', 'USER', and 'CREATED AT'. The table contains one entry: 'In the field of informati - Floppy Disk Drive' in the 'CHOICE' column, 'In the field of information and communication technology, what is the full form of FDD ?' in the 'POLL' column, 'Arockialivin' in the 'USER' column, and 'April 14, 202' in the 'CREATED AT' column. Below the table, it indicates '1 vote'.

devmahrn x (no subje x Django V x ezhilpop x Submit P x Download x mcquest x (1) Whats x Select vot x

127.0.0.1:8000/admin/polls/vote/

Django administration

WELCOME, AROCKIALIVIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Polls > Votes

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

POLLS

Choices + Add

Polls + Add

Votes + Add

Select vote to change

ADD VOTE +

Q Search

Action: Go 0 of 1 selected

CHOICE	POLL	USER	CREATED AT
<input type="checkbox"/> In the field of informati - Floppy Disk Drive	In the field of information and communication technology, what is the full form of FDD ?	Arockialivin	April 14, 202

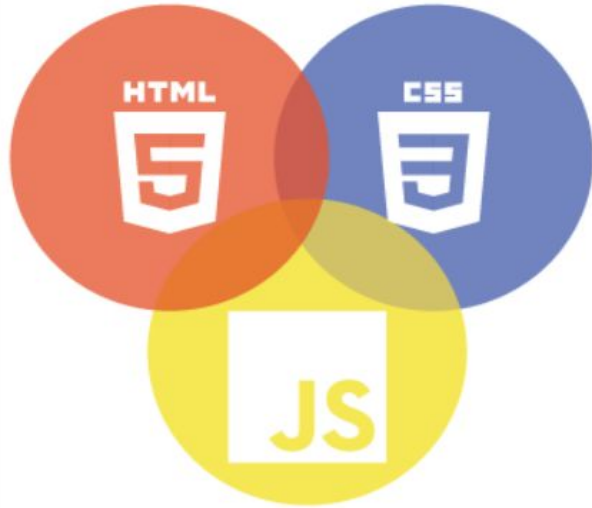
1 vote

Type here to search

ENG 08:58 14/04/2024

## 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!**