NEXT GEN EMPLOYABILITY PROGRAM

Creating a future-ready workforce

Team Members

Student Name S.Mallika:
Student ID:au951221104019

College Name
JP college of engineering

CAPSTONE PROJECT SHOWCASE

Project Title

Voting Web Application using Django Framework

Abstract | Problem Statement | Project Overview | Proposed Solution | Technology Used | Modelling & Results | Conclusion

Abstract

The ‘Online Voting System’ is a web based voting platform for conducting elections online. This system seeks to use face recognition algorithm for voter identity authentication to enhance the security of the electioneering process and ultimately providing an online platform which enables all eligible voters to exercise this activity from any location. The user must sign in/login using their respective credentials and they will be logged in into the system only after the face recognition authentication is successful. Thereafter, the voter can cast their vote securely and logout of the system. Hence, this project based on Online Voting System could be used for conducting secure and fair elections online.

Source :

Problem Statement

The present voting system applicable has proved inefficient as the voters' registration process is slow, the manual collation of results takes time and gives room for result manipulation, also the inaccessible nature of election venues which includes the long distances to be covered by voters' to their registered location increases voters apathy towards the election processes, and finally the issues of ballot box snatching and damage

Source :

Project Overview

Voting as a term encompasses a broad range of voting systems that apply elements in one or more steps of the cycle. There are many levels to voting in a broad sense which could be collation, verification, internet voting, remote online voting e.t.c. Following the definition of a system as anything that takes an input and gives an output, an voting system is any system that can offers both electronic and online voting. It could also incorporate registration, verification, collation, remote online voting and real-time result display. An voting system generally comprises the following for it to work efficiently:

An interactive voting user interface on an device which provides a friendly environment for voters to authenticate and cast their votes, it also serves as a means of collection the individual votes and storing them in the local and central database.

An administrative dashboard for voters registration, details update and coordination and monitoring.

A database management system for the storage of election, voting and voters data.

A result display interface.

Source :

Proposed Solution

Maintain all information of all the candidates and votes.

Check whether voter have voted or not.

Increase the voting percentage.

To make voting an easy process by avoiding problems like.

- Security

Booth capturing

Source :

coding:

```
import os
import sys


def main():
    """Run administrative tasks."""
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'pollproject.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)


if __name__ == '__main__':
    main()
```

Source :

```python
from django.contrib import admin
from .models import Question, Choice

admin.site.site_header = "The Poll Mall"
admin.site.site_title = "Voting Admin Area"
admin.site.index_title = "Welcome to our Voting Admin Area"

class ChoiceInLine(admin.TabularInline):
    model = Choice
    extra = 3
class QuestionAdmin(admin.ModelAdmin):
    fieldsets = [(None, {'fields':['question_text']}), ('Date Information', {'fields': ['pub_date'], 'classes':['coll
    inlines = [ChoiceInLine]

admin.site.register(Question, QuestionAdmin)
```

Source :

edunet foundation

Technology Used

Front-end

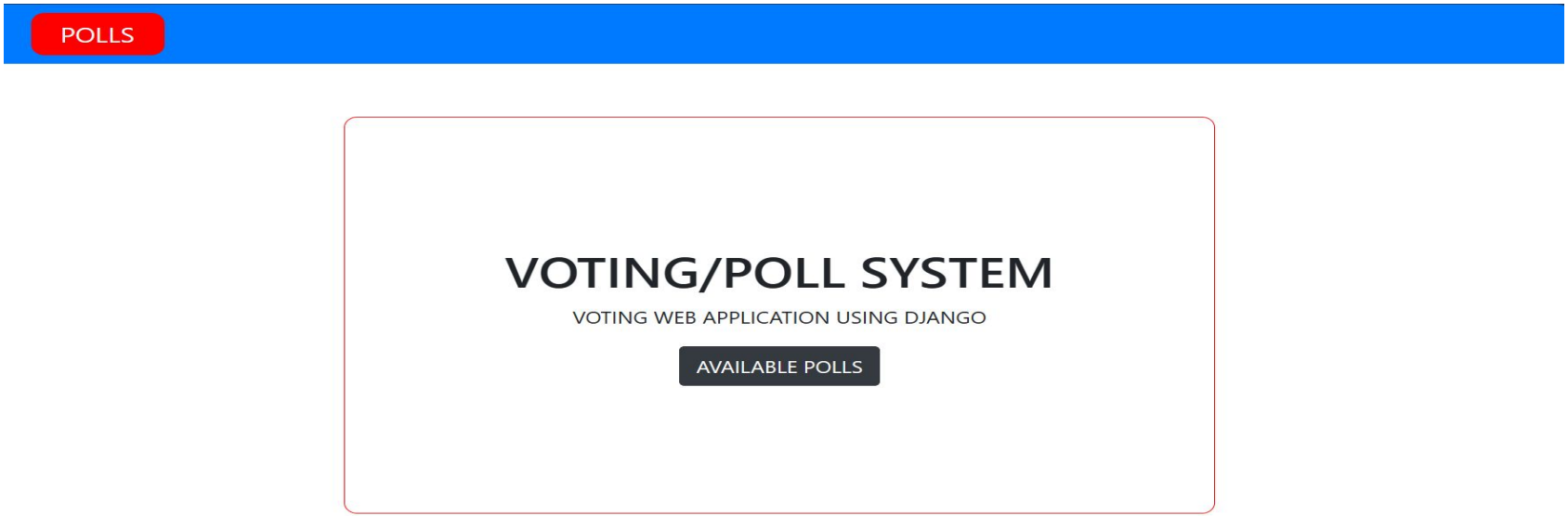
Back-end

Source :

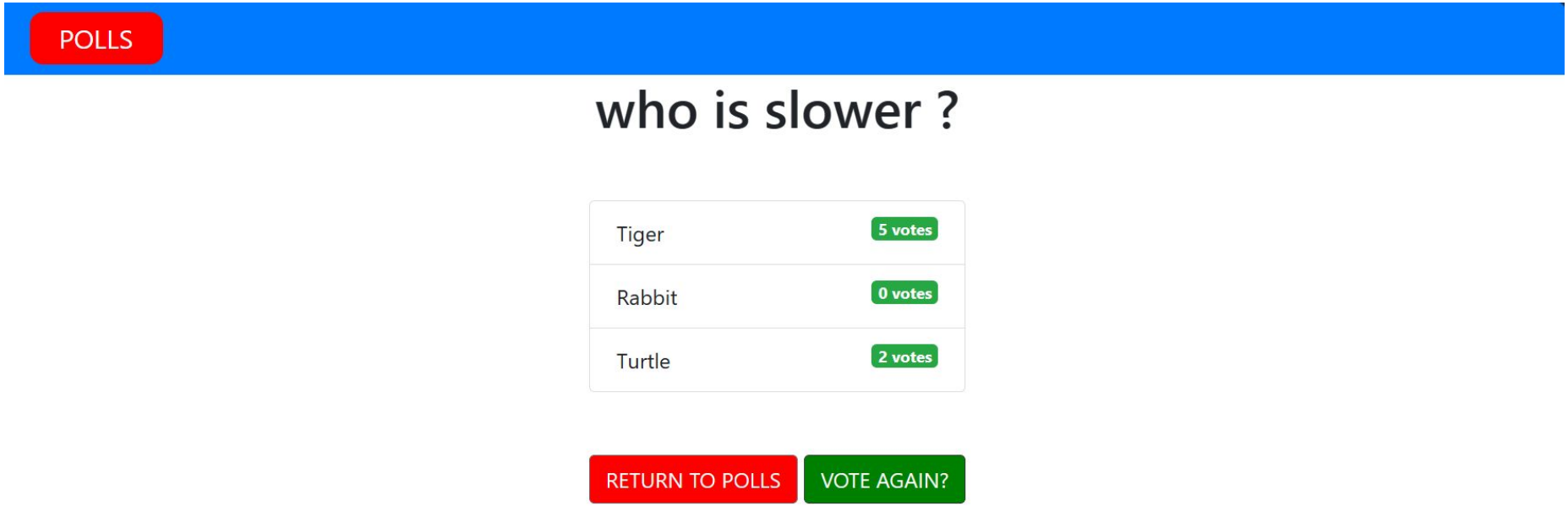
Modelling & Results

In the voting web application developed using the Django framework, the data model encompasses entities such as Users, Voting Events, Ballots, Candidates, and Votes. Users register and authenticate securely, gaining access to active voting events where they can cast
their votes on customizable ballots featuring various voting methods. Real-time result updates are provided, displaying the current status of the voting process and the accumulated
votes for each candidate. The application implements stringent security measures to protect the integrity of the voting process, including encryption techniques and authentication mechanisms.

Source :


POLLS
VOTING/POLL SYSTEM
VOTING WEB APPLICATION USING DJANGO
AVAILABLE POLLS


POLLS
POLLING QUESTIONS
who is slower ?
READY TO VOTE
WINNER!


POLLS
who is slower ?
Tiger
5 votes
Rabbit
0 votes
Turtle
2 votes
RETURN TO POLLS
VOTE AGAIN?

Future Enhancements:

For future enhancements, the voting web application built on the Django framework could incorporate advanced features such as blockchain-based voting for enhanced security and transparency, integration with biometric authentication systems for secure user verification, and support for mobile voting applications to increase accessibility and participation. Additionally, implementing machine learning algorithms could improve the accuracy of result predictions and identify potential anomalies or irregularities in voting patterns. Furthermore, enhancing the user interface with interactive data visualization tools could provide users with deeper insights into voting trends and results. Overall, these enhancements would further elevate the voting web application's capabilities, ensuring its effectiveness, security, and inclusivity in democratic decisionmaking processes.

ReplyForward

Add reaction

Conclusion

In conclusion, the voting web application developed using the Django framework represents a significant advancement in facilitating democratic decision-making processes. Through robust security measures, intuitive user interface design, and scalable architecture, the application provides a reliable platform for users to participate in voting events securely and transparently. The utilization of Django's built-in functionalities, coupled with potential future enhancements such as blockchain integration and machine learning algorithms, ensures the continuous evolution and effectiveness of the application in meeting the evolving needs of democratic societies.

Source :

edunet foundation

Thank You!