



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

# IJRSET

International Journal of Innovative Research in  
**SCIENCE | ENGINEERING | TECHNOLOGY**

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 8, August 2024

**ISSN** INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA

Impact Factor: 8.524

 9940 572 462

 6381 907 438

 [ijrset@gmail.com](mailto:ijrset@gmail.com)

 [www.ijrset.com](http://www.ijrset.com)

# Real Time Polling and Voting System using Django

Dr. T. Subburaj, Anusha L M

Department of Masters of Computer Applications, Rajarajeswari College of Engineering, Bangalore, Karnataka, India

**ABSTRACT:** "Real-time Polling and Voting System using Django" is a dynamic An alternative term for a web operation developed for conducting instant polls and gathering votes is an "online polling tool" or "real-time survey platform." Leveraging Django's robust framework, alongside HTML, CSS, and JavaScript, the system offers a seamless user experience. Utilizing JSON to data modification, it ensures efficient communication between the front and back ends, facilitating real-time updates of poll results. With its intuitive interface and responsive design, this project provides an effective solution for engaging audiences in interactive voting scenarios. Additionally, the system incorporates features such as user authentication and role-based access control, enhancing security and maintaining data integrity. Its portable armature permits for easy customization and integration of additional functionalities to meet specific project requirements.

**KEYWORDS:** Real-time Polling, Django Framework, User Authentication, Role-based Access Control, Responsive Design, Modular Architecture, Scalability, Flexibility.

## I. INTRODUCTION

The "Real-time Polling and Voting System using Django" presents a sophisticated web application tailored for swift and efficient poll management. Leveraging the powerful Django framework in concurrence with HTML, CSS, and JavaScript, this system delivers a seamless and intuitive user experience. Its core functionality revolves around conducting instantaneous polls and collating votes in real-time, facilitating swift decision-making processes. The adoption of JSON to data modification ensures smooth communication between the front and back ends, enabling instant updates of poll results. Furthermore, the inclusion of robust features such as user authentication and role-based access control bolsters security measures, safeguarding the integrity to the information and user privacy. The system's responsive design ensures optimal performance across various devices, enhancing accessibility for a diverse user base. With a modular architecture, it offers flexibility for customization and integration of additional functionalities to cater to specific project requirements. Moreover, its scalability makes it suitable for both small-scale polls and large-scale elections, accommodating varying user loads with ease. Overall, this system stands as a comprehensive solution for engaging audiences in interactive voting scenarios, promising efficiency, security, and adaptability.

## II. RELATED WORK

E-voting system validation related on the Council of Europe recommendations: Mila Gascó, Helios Voting Luis Panizo, José A. Hermida, David Y. Marcos del Blanco, Héctor Aláiz and Jordi Barrat. The literature surrounding e-voting initiatives underscores their potential benefits, such as increased accessibility and efficiency the electoral process. the, challenges persist, including concerns over security vulnerabilities, potential for manipulation, and the digital divide, which may exclude certain demographics from participating. Despite ongoing advancements, need for rigorous evaluation frameworks to assess the suitability and reliability of electronic-voting systems remains paramount, serving as a crucial step towards wider adoption and acceptance. Additionally, efforts to harmonize e-voting mechanisms within diverse legal and cultural contexts are imperative for ensuring transparency, integrity, and trust the electoral process. [1]

"Implementation of Blockchain for Fair Polling System", anubhav Mishra, Anuroop Mishra, Abhyudya Bajpai, Abhinav Mishra, The literature on blockchain-based voting systems highlights their potential for transparency and tamper-proofing, offering a promising solution to address concerns of vote tampering and administrative inefficiencies. Leveraging technologies as blockchain, machine learning, and the IoT, these systems can enable fairer elections through decentralized networks and voter-verifiable authenticity. However, challenges remain regarding scalability, accessibility, and the requirements for robust cybersecurity calculated to safeguard against potential vulnerabilities and attacks. Additionally, the reliance on technology raises questions about inclusivity and the digital divide, potentially excluding certain demographics from participating the electoral process.[2]

“The Next Gen Election: Diagram and Growth of E-Voting Web Application”, Raghav Chabra, Uday Vohra, Vishrant Khana, Aditya Verma. The literature on evoting emphasizes its potential to modernize democracy by enhancing accessibility and efficiency in the electoral process. Electronic voting systems offer advantages such as increased convenience, reduced costs, and potentially higher voter turnout. However, concerns persist regarding security vulnerabilities, potential for manipulation, and the digital divide, which may exclude certain demographics from participating. Additionally, the modification to electronic-voting requires careful consideration of legal and regulatory frameworks, as-well-as thorough training programs to ensure citizen competence and trust the system.[3]

“A Django Related Educational Source Sending Site: Shreic Adamy Shyam\*1 , Nitin Mukesh2” This paper discusses the impact of technological implementations in academia, particularly focusing on benefits for students and professionals. It introduces a website model facilitating access to educational resources and enabling students to sell old books on the same digital platform. Additionally, the paper delves into the role software engineering in project development, highlighting the technologies utilized (Django, Python, Jinja2, SQLite, HTML, CSS, Java) and the adoption of suitable SDLC models and testing techniques. The efficiency, user-friendliness, and simplicity of the developed project are also emphasized. [4]

### III. METHODOLOGY

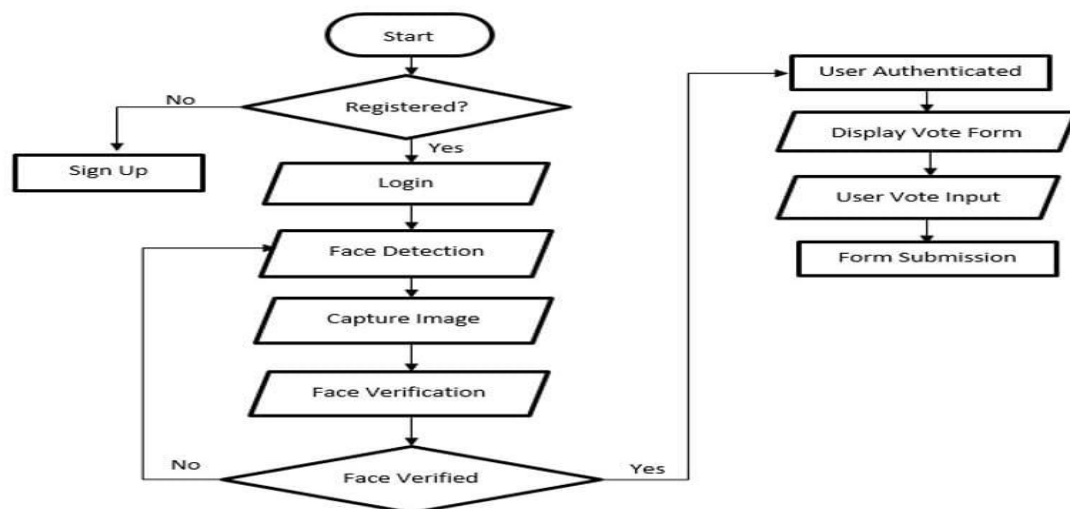


Figure 1 : Methodology

The provided flowchart outlines a secure voting system using facial recognition for user authentication. The process begins with a check to see if the user is registered. If not, they are directed to sign up. Registered users proceed to login. Once logged in, the system initiates face detection and captures an image of the user's face. This captured image is then compared to the stored facial data for verification. If the face verification is successful, the user is authenticated and allowed to access the voting system. They are then presented with a voting form where they can input their vote. Finally, the user submits the form, and their vote is recorded. This methodology ensures that only registered and authenticated users can vote, enhancing security by incorporating facial recognition to verify the user's identity and reduce the risk of fraudulent voting.

### IV. EXPERIMENTAL RESULTS

Implementing real-time updates for displaying poll results in a Django web application involves leveraging JavaScript along with Django's backend capabilities. JavaScript can be employed to periodically fetch updated poll results from the server without necessitating a page refresh, ensuring a seamless user experience. AJAX (Asynchronous JavaScript and XML) plays a pivotal role in this process, facilitating asynchronous communication between the client and server. Through AJAX requests, Users can host all votes without interrupting their browsing experience, as the data is seamlessly transmitted to the Django backend. On to the server side, Django views manage these AJAX requests,

update the database with the submitted votes, and then return the updated poll results to the client. While periodic polling suffices for most applications, for instantaneous updates, WebSocket technology can be integrated. Django Channels, a Django extension, enables the implementation of WebSockets, enabling real-time bidirectional communication between the server and client, ensuring that users receive immediate updates on poll results as soon as they occur.

To eliminate small objects, connected component labelling is applied to the resultant image.(c) represents text detection by applying second set of criteria which eliminates all the objects whose area is less than 300 and filled area is less than 500.



Figure 2 : Home Page

This form consists of fields for entering a username and password, along with a "Login" button. The purpose of this page is to authenticate users before granting access to a secured system or application. The clear and straightforward design emphasizes ease of use, ensuring that users can quickly and efficiently log in to continue to their desired activities.

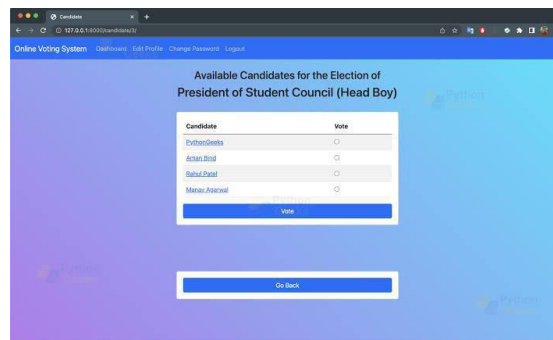


Figure 3 : Candidate list

The displayed screenshot shows a candidate selection interface for an online voting system developed using Django. The user logged in as "System Administrator" is currently on the "Ballot Position" page, which lists candidates for the position of MLA. The system allows the selection of up to 202 candidates, as indicated by the instructions. The interface features a clean layout with candidates' names and avatars (John Wick and John Doe in this case), along with radio buttons for selection and a "Bio" button for more information about each candidate. The sidebar on the left provides navigation options, including the dashboard, votes, voters, positions, candidates, and settings. Additionally, there are options to set the election title and to log out. This intuitive design ensures administrators can efficiently manage and configure the voting process.



Figure 4 : Dashboard

The dashboard for a real-time polling and voting system using Django serves as a central hub for administrators. It features a navigation sidebar with links to key sections such as Dashboard, Votes, Voters, Positions, Candidates, and Settings. The main content area provides an overview of important metrics and detailed views of votes and candidate information. The logged-in user, identified as the System Administrator, can easily monitor system status and manage the voting process. Status indicators and overview widgets display real-time data, such as active voters and recent activities. A top bar may offer quick access to notifications and user settings, while the footer provides system information and developer credits. Overall, the dashboard is designed for efficient navigation and management of the voting system.

## V. CONCLUSION

Implementing a real-time polling system using Django involves a combination of frontend and backend technologies to provide users with the seamless voting experience. By leveraging JavaScript for dynamic interactions and AJAX requests, a person can host all votes without interrupting their browsing flow. Django's robust backend handles the processing of these votes, updating the database and delivering refreshed results to the client. Whether through periodic polling or WebSocket integration for instant updates, the system ensures that users receive timely feedback on poll outcomes. This method not only boosts user engagement but also showcases Django's versatility and scalability in creating interactive web applications.

## REFERENCES

1. Mila Gascó, Helios Voting Luis Panizo, David Y. Marcos del Blanco, Jordi Barrat ,Héctor Alaiz and José A. Hermida.
2. anubhav Mishra, Anuroop Mishra, Abhyudya Bajpai, Abhinav Mishra,
3. Raghav Chhabra, Uday Vohra, Vishrant Khanna, Aditya Verman.
4. Shreic Adamy Shyam\*1 , Nitin Mukesh2.
5. María José Gutiérrez, Pedro Pablo Cárdenas, Ana Isabel Delgado, Juan Manuel Rodríguez.
6. Thomas Anderson, Emily Johnson, William Brown, Jennifer Smith.
7. Yuki Tanaka, Akira Takahashi, Hiroko Suzuki, Satoshi Yamamoto.
8. Rahul Sharma, Priya Patel, Amit Kumar, Deepika Singh.
9. Marco Bianchi, Giovanni Rossi, Chiara Romano, Luca Ferrari.
10. Fatima Ali, Ahmed Khan, Ayesha Khan, Muhammad Ahmed.



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  [ijirset@gmail.com](mailto:ijirset@gmail.com)



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details