

PROJECT ABSTRACT

Online Campus Voting and Poll System

A secure web-based voting platform for student elections and campus polls with anonymous balloting, live turnout tracking and result dashboards. Delivered built-to-order with the full application, admin panel, report, PPT and viva kit.

01. Title

02. Introduction / Background

Student elections and opinion polls on most campuses still run on paper ballots or ad-hoc Google Forms. Paper voting needs booths, volunteers and manual counting; generic form tools cannot enforce one-student-one-vote, keep ballots secret while publishing turnout, or prevent duplicate votes. This project builds a purpose-made online voting platform: an authenticated voter roll, anonymous ballot casting with one-time vote tokens, and a live results dashboard that shows turnout and vote shares without ever revealing individual votes.

03. Problem Statement

Campus elections suffer from low turnout because voting means standing in a queue between lectures, and organizers struggle with counting, recount disputes and duplicate-vote prevention. Online polls run on generic tools leak voter identity to whoever owns the form and offer no audit trail. The project addresses this with a dedicated voting web application: verified voter rolls, anonymous ballot storage separated from voter identity, duplicate-vote blocking, and transparent provisional results that build trust in the outcome.

04. Objectives

- Implement a verified voter roll with one-time vote tokens so each eligible student can vote exactly once.
- Keep ballots anonymous: store cast/not-cast status separately from vote content.
- Provide a live results dashboard with vote shares and turnout breakdowns.
- Support multiple concurrent elections and quick opinion polls with scheduling and deadlines.
- Maintain an audit log (ballots issued, cast, duplicates blocked) without compromising secrecy.
- Deliver the complete student kit: application, admin panel, report, PPT and viva Q&A.;

05. Proposed Methodology

The system is developed in three stages. (1) Voter management: the admin imports the voter roll (name, college email, PRN), and the system issues a single-use token per voter per election. (2) Balloting: the voter authenticates, receives the ballot, selects a candidate and

submits; the token is consumed and the vote is stored with no link back to the voter. (3)
Counting and display: a results engine aggregates votes into shares and turnout statistics served on a public dashboard, while an audit view shows issuance/cast/duplicate-attempt counts for organizers.

06. System Architecture / Working

A three-tier web architecture: a responsive frontend (ballot pages, dashboards), an application backend exposing REST APIs for voter auth, ballot casting and result aggregation, and a relational database holding voters, tokens, elections, candidates and anonymized votes. Token validation middleware rejects reused or forged tokens before any vote is recorded. Role-based access separates voters, election officers (create elections, view audit) and read-only result viewers.

07. Hardware / Software Requirements

- Server: any machine or shared hosting running the backend (2 vCPU, 2 GB RAM adequate for campus scale)
- Backend: Python (Django/Flask) or Node.js with a relational database (PostgreSQL/MySQL)
- Frontend: HTML5, CSS3, JavaScript (responsive; works on phones for queue-free voting)
- Voter roll import: CSV upload; email service for token delivery and reminders
- HTTPS deployment for ballot secrecy in transit

08. Expected Outcomes

A working voting platform where an admin can create an election, import voters, open and close polls on schedule, and publish results. Expected behavior: duplicate votes are blocked at the token layer, results update as votes arrive, and the audit page reconciles ballots issued against ballots cast. Load expectations are design targets (a few thousand concurrent voters on modest hosting), verified by the buyer with the included load-test script rather than claimed as measured.

09. Applications

- Student council and department representative elections
- Hostel committee and mess-menu opinion polls
- Club leadership elections and event-theme voting
- Class-representative and survey polling for departments

10. Future Scope

- OTP-based second-factor voter authentication
- End-to-end verifiable voting with cryptographic receipts
- Ranked-choice and multi-winner counting methods

- Integration with the college ERP for automatic voter-roll sync