CARITAS UNIVERSITY AMORJI-NIKE, EMENE, ENUGU STATE

Caritas Journal of Physical and Life Sciences

CJPLS, Volume 5, Issue 1 (2026)

***Article History*:** *Received:11th September, 2026, Revised: 29th September 2026, Accepted: 29th October, 2026*

Development Of Ipvote: a Secure Electronic Voting And Result Management System - a Case Study Of The Office Of The Accountant General Of The Federation

AKPAN Abundance Mbuotidem,
AMAZUE Christabel Oluchukwu,
OHAGBARA Pascal Chinonso
CHINAZOM Agu Miriam.
Department of Computer Science,
Caritas University, Amorji-Nike, Enugu State, Nigeria.

Abstract

Organization of elections manually by voting and result managing involves some security problems, such as time wasting, multiple-voting, ballot box snatching, ballot paper rigging, inaccurate vote counting, ballot-box stuffing, impersonation, and lack of electoral records. This research work is the development of IPVote: A Secure Electronic Voting and Result Management System for Elections, using Office of the Accountant General of the Federation (OAGF) as a case study, that will enhance the security and ease of election result compilation. The IPVote consists of a comprehensive, computerized registration system and voter authentication, candidate management, electronic voting, automated result compilation and management system for electronic voting, as well as security mechanism, which restricts the number of authenticated voters and prevents enumeration of the voting data. IPVote uses a web-based system architecture, whereby authorized persons will be able to conduct the election and view the results through the system. Requirements analysis, design implementation, and testing phases of the system development life cycle (SDLC) were undertaken to ensure that the system satisfy the main objectives of being reliable and easy to use. The finalized system is expected to reduce the time wasted, human errors, fraudulent activities, number of man-power used and to provide real-time result. The study concludes that organization of the voting and result management is a good system that can replace the existing manual system.

Keywords: ***Electronic Voting System; Secure E-Voting; Voter Authentication; Election Security; Result Management; Automated Vote Counting; Cybersecurity; Election Integrity.***

Introduction

Office of the Accountant General of the Federation (OAGF) – is a Nigerian government institution charged with the management of funds and accounts of the Nigerian federation. The Integrated Personnel and Payroll Information System (IPPIS) department manages the payroll of the government employees. To motivate staff and show them some appreciation, the IPPIS department conducts a monthly election to select the "Best Staff of the Month". Although this process used to involve paper-based voting by hand, the process was slow and very logistically challenging.
Having been exposed to all these difficulties during my 6-month industrial training stint in OAGF, it was suggested that we temporarily move the system to google forms. While this increased the engagement factor and is more accessible, it introduced a host of other issues, ranging from some employees facing difficulty in logging into the system to the key issue of zero transparency in collating results. Employees could not see the way in which the votes were being counted in real time. This project underscored the importance of a dedicated, secure, transparent electronic voting system with the peculiarities of a government institution like OAGF in mind. The paper describes the development of an "iPVote" web-based system to help streamline and make transparent the internal elections conducted by the IPPIS department.

Review of related works

Systems similar to iPVote can help determine the position of iPVote in the broader context and confirm that it is based on earlier ideas but designed to focus on the particular, defined need. Helios Voting system Helios is an online voting system designed at Princeton University with a focus on open-audit voting (Adida, 2008). It allows voters to verify their vote within the tally and also verify that their vote has been cast with the aid of cryptography, see fig. 1 below.

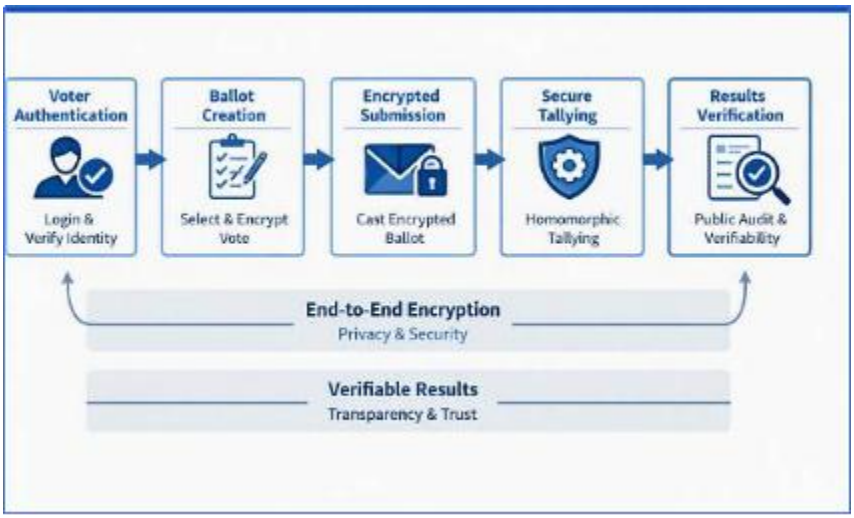


Fig. 1: Helios Voting System Flowchart

This approach is too complex for employees of the IPPIS and you need to deal with long cryptographic strings during verification that would be not be comprehensible to non-technical voters.

The important takeaway from this work for iPVote is the lesson of how important verifiability can be, achieved here by using a simpler, real-time dashboard. Commercial Platforms (Election Buddy, Simply Voting) Many organizations use a generic pay-per-use third-party solution for staff and board elections (Chukwunweike, 2021). They are expensive to buy every year, but stable and feature-rich. Most importantly, in contrast to the staff ID

of a Nigerian government office, they are generic and not tailored to the particular flow of work and identity scheme.

They do not give the real-time public transparency that iPVote does.

Blockchain-Based Voting System and Research This compares using a blockchain, a shared digital ledger, for voting, arguing that each vote is stored in a manner that makes it impossible to change and publicly verifiable (Tian, 2018). In an academic environment this is an interesting idea but the security advantages for electoral contests within small institutions remain a topic of discussion and it is complicated and potentially slow. An accessible web application is far more appropriate for the needs of the IPPIS section.

Google Forms (Current IPPIS Voting System) As the most similar to iPVote, this past system of use showed that a digital tool was highly desirable for its ease of use; analysis later showed that what were the essential needs of iPVote: a real-time public view of the results and stronger official identification with a real-time official ID of the voter. Official organizational requirements like auditability and secure access control (Chukwunweike, 2021) are better served by using a purpose-built tool.

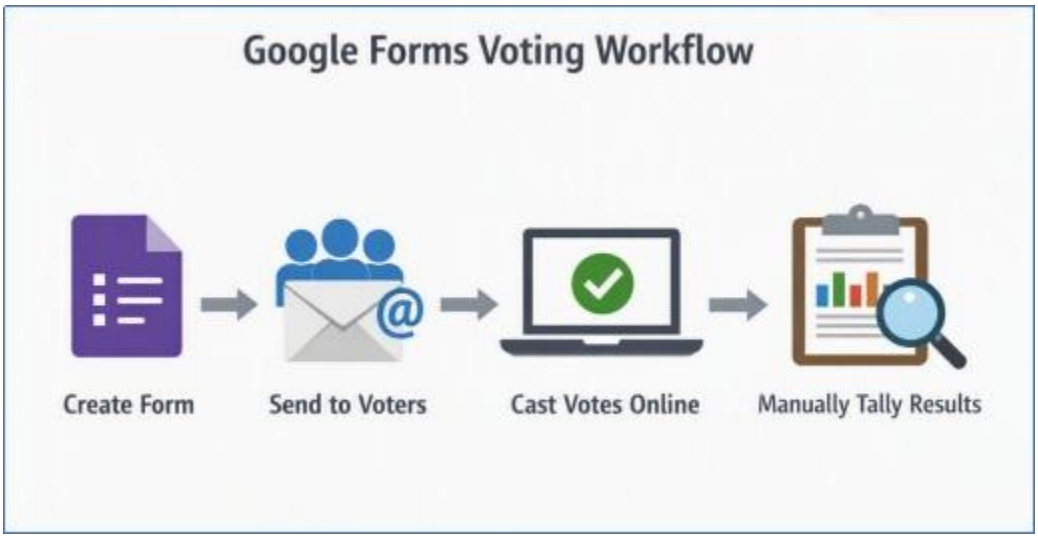


Fig. 2: Google Forms Voting Workflow

In sum, this thesis holds a special place: far more useful than complex academic or blockchain solutions, far more well-tailored to the organization than commercial systems and far more secure and organizationally aligned than Google Forms. It seeks to present a real-facing problem (company elections) to the specific, existing group of its users and respond to their expressed need.

Research Gap

Analysis of other electronic voting systems reveals many weaknesses that demand the design of a purpose-developed voting and result management platform for organizational elections. While the Google Form-based system used for the current IPPIS staff election has proven the basic convenience of allowing employees to vote remotely, it depends on personal Google accounts for authentication and can be inaccessible for employees without Google accounts, as well as challenging for those unable to recall their passwords or wishing to avoid connecting their accounts to official activities. What's more important is result transparency and verifiability that is lacking. According to the present system, elections will be held in a spreadsheet that's under the director

of the election manager, and voters don't have any way of tracking the progressive count or verifying how the ultimate outcome was reached.

It means that the election process is reasonably opaque and you can't independently verify the method. Of course, the literature also shows the ways that current e-voting systems tend to focus on one or more distinct priorities. Helios is designed around cryptography-based open-audit verification system but verification steps required may be too difficult for a company election of this size. Commercial voting systems are meant as generic solutions, rather than systems designed around the specific identity management and administration mechanisms of the IPPIS workplace.

Blockchain solutions are able to offer immutability and verifiability, but that might be an unnecessary level of complexity for internal company elections. These existing systems indicate the main reasons for designing a company election system: simplicity, official employee-ID authentication, requirement for single vote, result security, administration and publication of election results and visibly, but not necessarily immediately, observable collective result of the election. The election system designed in this thesis has the potential to fill this vital gap. The paper builds a secure host environment to replace the reliance on third-party survey systems and is a general election environment where voter management, candidate administration, election management and result tabulation are integrated together, see fig. 3 below.



Fig. 3: Research Gap Summary Diagram

Results and Discussions



Fig. 4: iPVote Admin Portal Sign In Page

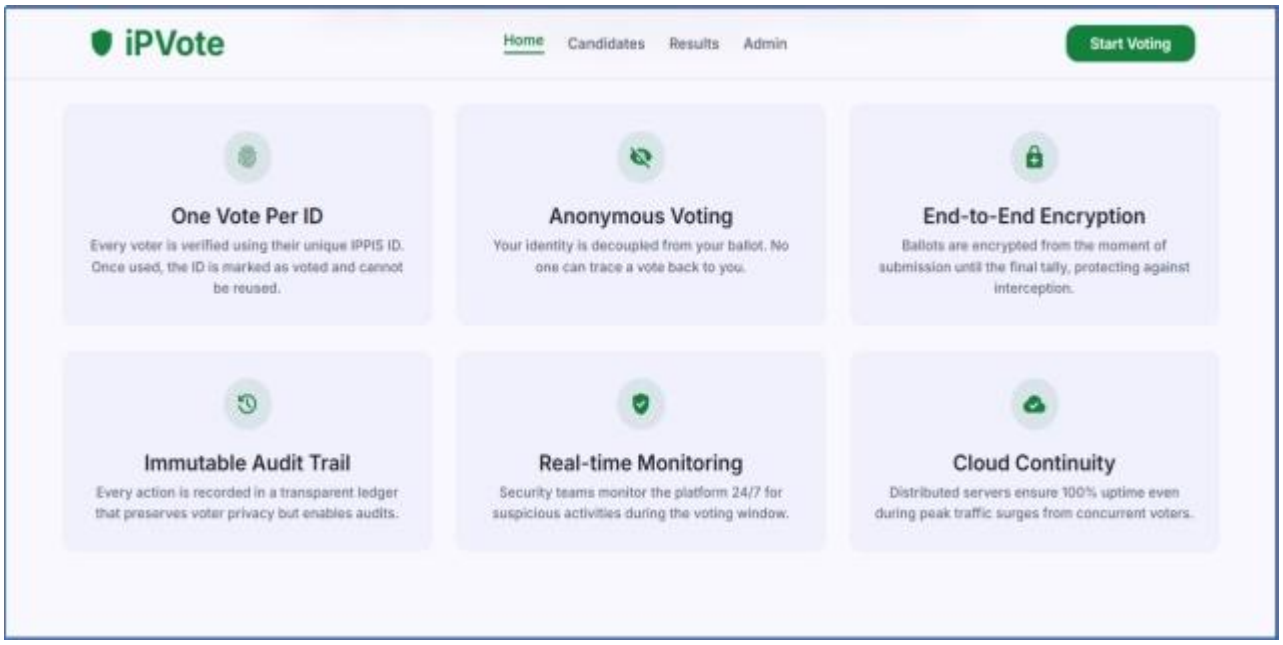
The iPVote system developed and tested successfully offers a simple web-based electronic voting and result management platform to conduct the IPPIS monthly staff election, see fig. 4. The system separates the voting and admin interfaces while combining an election creation process, candidate presentation, voter pre-registration, voting process and results presentation within a common application.

The most significant result is that the system replaces the reliance on third-party account authentication with pre-registration by the administrator who is in turn guided to verify the identity of each voter using official IPPIS staff codes, as illustrated in fig. 5 below.



Fig. 5: iPVote Admin Dashboard Showing No Active Election

Fig.6: Public Home Page Showing Election Day Summary Cards

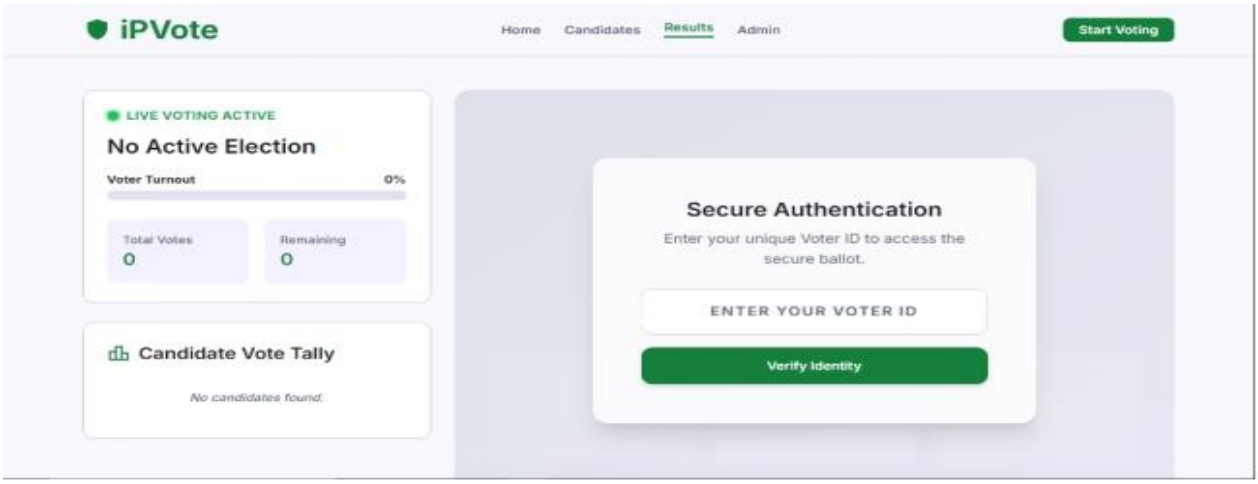


Individual or batch uploads of CSVs can be used to add verifiable voters who are automatically assigned to one or more elections. Duplicate voter entries for a given election are rejected by the database, see fig. 6 above. The system implements the principle of one person, one vote by updating the voter's used status once they have cast

a ballot; thereby preventing a second ballot being cast by the same person for a given election. The save operation and the update of the used status are performed as a single transaction to minimize the scope for a record to go out of sync. Multiple concurrent voting sessions were tested and are said to have confirmed the operation of the single-vote mechanism.

The system also delivers a real-time results dashboard immediately following a successful vote, showing that ballots are recorded and tabulated, and providing transparency that was lacking when the ballots were held in a Google Form survey. Security measures were built in, with passwords stored using PBKDF2-SHA256, routes protected via session-based auth, SQL parameterization used as a safeguard against SQL injection, upload filename sanitized and the hosting platform provides HTTPS. The database was configured with foreign-key enforcement and Write-Ahead Logging. The final application was then hosted on Pythonanywhere.com to avoid the need to install dedicated platforms or services on the users' computers, see fig. 7 below.

Fig. 7: Live Results Page with Secure Voter Authentication Panel



Discussion

The above results show that a lightweight purpose-built web application can improve upon the main procedural weaknesses identified in the existing set-up. Using official IPPIS staff identifiers rather than personal third-party accounts will reduce the reliance on individual accounts that the current Google Form process requires.

Allowing an administrator to directly upload multiple voters to an election creates a cross-reference with the other elections in which a given voter may be represented. The implementation of the single vote rule also addresses the weak area of preventing individual double-voting. This is achieved through the design of a web application that combines the presentation of the voter's registration status with a database transaction to enter a vote. The current live update of the vote count has been shown to be a tangible difference between the previous process of collecting votes via Google Forms and storing in a spreadsheet and the new process that makes the current vote count available to voters in real-time.

The above results demonstrate that a suitable technology stack comprising Python, Flask, SQLite, and web technologies can be combined effectively to create a lightweight, stand-alone web solution for the single or limited number of elections that the OAGF aims to conduct.

However, note must be taken of the results' limitations. It was not feasible to verify the application on the Ministry of Finance servers with the full institutional information set. Testing was carried out with simulated

voter data stored on the platform; hence, the results should be interpreted as demonstrating the prototype's technical viability rather than performance during a live election.

Conclusion

The secure web-based electronic voting and result management application, iPVote, was conceived as a solution to the discovered shortcomings in the voting process used for the Office of the Accountant General of the Federation (OAGF) IPPIS monthly staff election. Previous analysis showed that, although more accessible than paper-based voting, the existing system was crippled by authentication issues and severely limited by transparency and election management controls. The iPVote implementation offers a registration mechanism under administrator control, management of candidates and poll choices, electronic vote-casting, enforcement of single votes, secured data persistence and presentation of real-time results.

The project has shown how a purpose-written web application can serve as a suitably tailored alternative to a generic online survey application for the purpose of organizational electoral activities.

An implementation based on Python Flask, SQLite, PHP, HTML and related web technologies was able to run on a web server and be accessed by most established web browsers while incorporating adequate security, key-logging and data validation controls including session management, password hashing, parameterized database queries, secure uploading of files and HTTPS. However, iPVote should still be regarded as a prototype implementation rather than an officially approved electoral infrastructure, because it was evaluated within a simulated environment and legal, organizational and security considerations must be addressed before it can be used by the OAGF as an electoral platform.

Recommendations

Based on the above observations and the limitations specified earlier, the following recommendations are made:

1. Two-factor authentication should be used by changing the user login authentication mechanism to include an off-line message- or email-delivered password (e.g., on-time password) which is not stored on the server.

2. The administrator function should be extended to distinguish different administration roles for result viewing, poll management, and voting supervision.

3. An OAGF official email communication system should be used to notify all eligible staff when a voting session is open.

4. An exportable audit trail containing relevant administration and voting events should be provided to aid staff to perform election audits.

5. If the application is to handle a significantly increased number of voters, then the system migration should be considered from SQLite to a more scalable database implementation such as PostgreSQL.

6. The organization should consider security evaluations, security audits and integration with the current staff database as part of the initiative to use iPVote in production.

References

Adida, B. (2008). Helios: Web-based open-audit voting. In Proceedings of the 17th USENIX Security Symposium (pp. 335 348). USENIX Association.

Akinyokun, N. A., & Teague, V. (2017). Security requirements for electronic voting systems. Computer Science Review, 25, 1 25. https://doi.org/10.1016/j.cosrev.2017.05.003

Chukwunweike, J. N. (2021). Web application development with Python and Flask: A project-based approach. Pearsons Nigeria.

Flask Development Team. (2023). Flask documentation (version 3.x). Pallets Projects. https://flask.palletsprojects.com/

Independent National Electoral Commission (INEC). (2019). Framework and regulations for the use of electronic voting machines in Nigeria. INEC Press.

Mba, C. G. (2020). Digital transformation challenges in the Nigerian civil service: A case study of the Office of the Accountant General of the Federation. Journal of Public Administration and Governance, 10(2), 112 128. https://doi.org/10.5296/jpag.v10i2.16845

Mercuri, R. (2002). A better ballot box? IEEE Spectrum, 39(10), 46 50. https://doi.org/10.1109/MSPEC.2002.1031832

National Information Technology Development Agency (NITDA). (2019). Nigeria Data Protection Regulation 2019. https://nitda.gov.ng/wp-content/uploads/2024/02/NigeriaDataProtectionRegulation2019.pdf

Office of the Accountant General of the Federation (OAGF). (2023). Human resources policy manual: Staff recognition and awards. Internal Publication.

Okonigene, R. E., & Adewale, O. S. (2019). Database management systems: Design and implementation for Nigerian institutions (2nd ed.). Lagos University Press.

Olatunji, K. (2022). User-centered design for low-bandwidth government applications in Nigeria. In Proceedings of the 2022 Nigeria Computer Society Annual Conference (pp. 45 52). Nigeria Computer Society.

Python Software Foundation. (2023). sqlite3 DB-API 2.0 interface for SQLite databases. Python 3 Documentation. https://docs.python.org/3/library/sqlite3.html