Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024
DOI : https://doi.org/10.33395/sinkron.v8i3.13734
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Student Organization Website with E-Voting Feature by Using Student Card Verification Concept Design

Yohanes Maria Jonathan Glenn Paskalis[1)], Karel Octavianus Bachri[2)*]
[1,2)]FakultasTeknik, Universitas Katolik Indonesia Atma Jaya, Jakarta, Indonesia
[1)] ymjonathangp2@gmail.com, [2)] karel.bachri@atmajaya.ac.id

Submitted : Jun 7, 2024 | **Accepted** : Jun 26, 2024 | **Published** : Jul 1, 2024

Abstract: Student organizations hold an election to decide their next head and vice head every year. The best voting method for student organizations is to use an independent website with a voting system. The voting system can use students' identity card and their student email as base for verification. OCR and face detection can be used for extracting all the needed information to validate the student card and verify it with the corresponding student email input. Other than the voting system, the website can be used to promote the student organization itself. The website was built using Nuxt for its front-end, Firebase for its back-end, and Cloud Vision API for its OCR and face detection module. There is a Lighthouse test, a stress test for the voting system, and a test to determine the optimal file size for the voting system. The results are a website that has an average Lighthouse score of 97.58. The stress test, which used a script that does submission repeatedly, results suggest that the voting system can handle up to 2000 voters at the same time. The optimal file size determined by the authors to be 500KB as the result of its test. The conclusions are a great performing website with a voting system can be built using Nuxt and Firebase, the voting system can be improved by adding another step of verification, and it's best to use and image with a file size above 250KB when using Cloud Vision API for optimal results.

Keywords: Cloud Vision; Firebase; Nuxt; OCR; Verification; Voting system; Website

INTRODUCTION

Election is a formal and organized choice by vote of a person for a political office or other position. An election is held at the end of a term of office. An election can be held by a country to decide the next president or by a student organization to decide their next head of the organization. Student organization in Atma Jaya Catholic University of Indonesia, which is later referred XU, is an organization whose members are active XU students. Every year, XU student organizations hold an election to decide the next head and vice head of the organization. For most XU student organizations, the people who have the right to vote are students inside the scope of the organization, such as university, faculty, or study program.

There are 7815 active undergraduate students studying in XU as of 2023 second half (*Universitas Katolik Indonesia Atma Jaya*, 2024). XU undergraduate students are divided into three campuses, which are Kampus 1 Semanggi, Kampus 2 Pluit, and Kampus 3 BSD (*Jelajahi*, 2024). From those data, an election must be held as efficiently as possible in the form of time, human resources, and cost.

An election that's held offline would take a long time and need a lot of cost and human resources. An online election would be relatively more efficient than an offline one. An online election could be done in many ways, such as by using an e-form, a polling feature in social media apps, or by an

*Author 2024

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i3.13734 p-ISSN : 2541-044X

independent website. Using an independent website would be the method compared to others for holding an online election.

The execution of an election using XU student identity card (KIM) and student email as verification components is an ideal method to be used on a student organization's independent website. Computer vision methods such as OCR and face detection can be used for extracting the information from the KIM image. Other than using it for election, the independent website could be used by the student organization to promote themselves outside of XU. The comparison between this research and the previous research can be seen in Table 1.

An online voting system method approached by (Pawar et al., 2020) was made to enhance security and save time compared to a traditional method. This system used vanilla HTMl, Bootstrap, jQuery, AJAX, MySQL, and Apache HTTP Server. In total, the system has six pages, which are a home page, a vote page, a result page, an about us page, a gallery page, and a question papers and syllabus page.

A similar voting system approached by (Fortuna & Khaeruzzaman, 2022) was made to add more convenience to the voters. It used a multi-task convolutional neural network, which was based on the FaceNet model, that was trained to verify the voters' faces using TensorFlow. The validation system compares the voters' faces during registration and during the election. This system used a citizenship identity card for verification. Other than that, the system used an OCR module from Firebase Machine Learning.

A voting machine method approached by (Yusoff et al., 2023) was made to improve the traditional voting system in a Malaysia's election. It used fingerprint biometric authentication instead of using password. In the system, instead of a fingerprint reader, the system uses a keypad matrix as an input. The voters first need to input their identity card (IC) number via the keypad matrix. The system then will verify the IC number using data from the database.

Table 1 Comparison of current research and previous research

Previous Research	Current Research
The students need to register and log in using a unique ID, which was gotten after the registration, to vote (Pawar et al., 2020).	The students don't need to register and use their student email and student card to vote.
Is an Android-based mobile application and only made for a voting system (Fortuna & Khaeruzzaman, 2022).	Is web-based and has other pages other than the election page to promote the student organization.
The voting system use Arduino Uno to verify the voters (Yusoff et al., 2023).	The voting system use Firebase Cloud Functions to verify the voters.

METHOD

To design the website, first the authors need to design the flowchart for the users. The voting system have two verification functions as seen on

Fig. 1. The first verification is client side, by using JavaScript (JS). The front-end was built using Nuxt to simplify the process, which will be explained later. The second verification is done on cloud computing via Google Firebase by matching the email identity and the KIM identity.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i3.13734 p-ISSN : 2541-044X

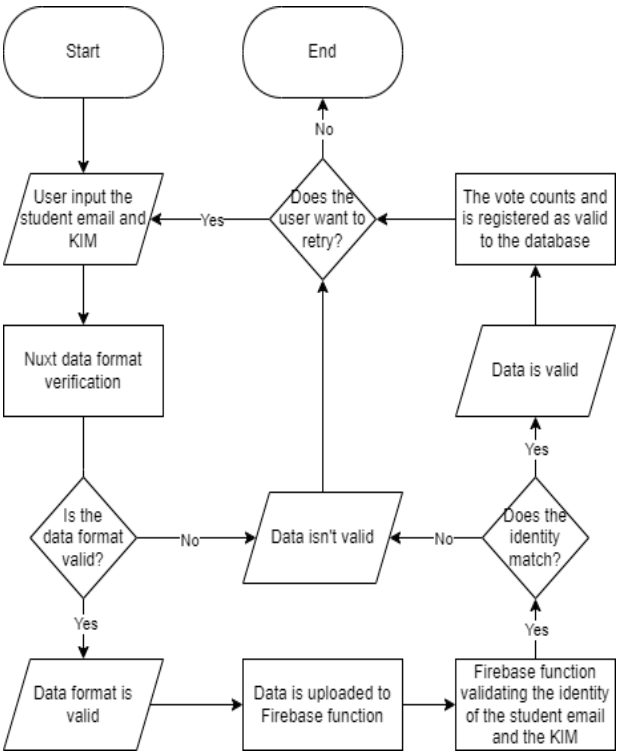

Fig. 1 System flowchart

XU students have their unique campus ID. Their student email username contains a maximum of seven letters of their first name continued by a dot and then their campus ID. XU student email domain is student.universityname.ac.id. For example, a student named Yohanes Maria with a student ID of 202004520031 would have a student which is yohanes.2020045200@student.universityname.ac.id.

An XU KIM has plenty of text that can be extracted using optical image recognition (OCR) model. OCR is a mechanism for converting scanned documents, printed text, or handwritten data into computer recognizable universal character encoding format (UNICODE) or machine-readable text over an optical mechanism (Karthikeyan et al., 2021). The important part is that it contains the student's name and their campus ID. Those two pieces of information are the main components for the online voting verification. Other than the identification, the voting system also checks the student card text format and if it contains only one picture of a face. To detect the face, the system uses a face detection application programming interface (API).

Other than the voting system, the website contains a few other pages for the student organization. The student organization website concept contains the home page, profile page, program page, contact us page, and election page. The home page is filled with a summary of all the information about the organization. The profile page explains the current ministry of the student organization. The program page contains all the organization's programs since it was first established. The election page is where the students go to vote for the next head and vice head of the organization.

The authors used NuxtJS as a front-end JS framework and Google Firebase as a back-end as a service. A front-end in the context of web application is the web design and the client-side rendering of web content (Mishra et al., 2021). Nuxt was used to design and build all the front-end of the website pages. According to a research by (Siahaan & Kenidy, 2023), Nuxt was slightly slower than Vue as a front-end JS framework. Therefore, the authors chose Nuxt, since it has plenty of built-in features to speed up the development process. Firebase is a comprehensive platform offered by Google that provides a set of back-end services and tools to help developers build and scale web and mobile applications quickly and efficiently (Zholdas et al., 2024). Firebase was used to host the website (Firebase Hosting) and as a database (Firebase Firestore). Firestore is more cost and time effective by using an items summary document instead of a document for every single item (Bahtiar Semma et al., 2023). A research by (Mumtahana, 2022) found that MonggoDB, which is a not only structured query language (NoSQL)

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i3.13734 p-ISSN : 2541-044X

database, is faster than MySQL, which is a structured query language (SQL) database. Firebase was also used for doing back-end logic for the voting verification system using Firebase Cloud Functions. The relation between Nuxt and Firebase for the voting system can be seen on Fig. 2. On the front-end, the HTML form input would include three data values, which are the student email, KIM, and the chosen candidate number. The form also has front-end verification for all the data inputs. The email format should match the university student email format explained before and the file must have jgp or jpeg extension. If the data format is verified, the user can submit the form. After the form has been submitted, the data is processed on the server. First the student email is uploaded as a Firestore document, then it will generate a unique id. Then the same id is used to name the vote value document in Firestore. After that the KIM image is uploaded to the Firebase Storage and named by the same id. One of the Cloud Functions function then does the identification process on the student email and KIM. If the verification is successful, then another Cloud Functions function will do an increment to the corresponding candidates vote count. The user then will be informed from the UI if the vote is valid or not.

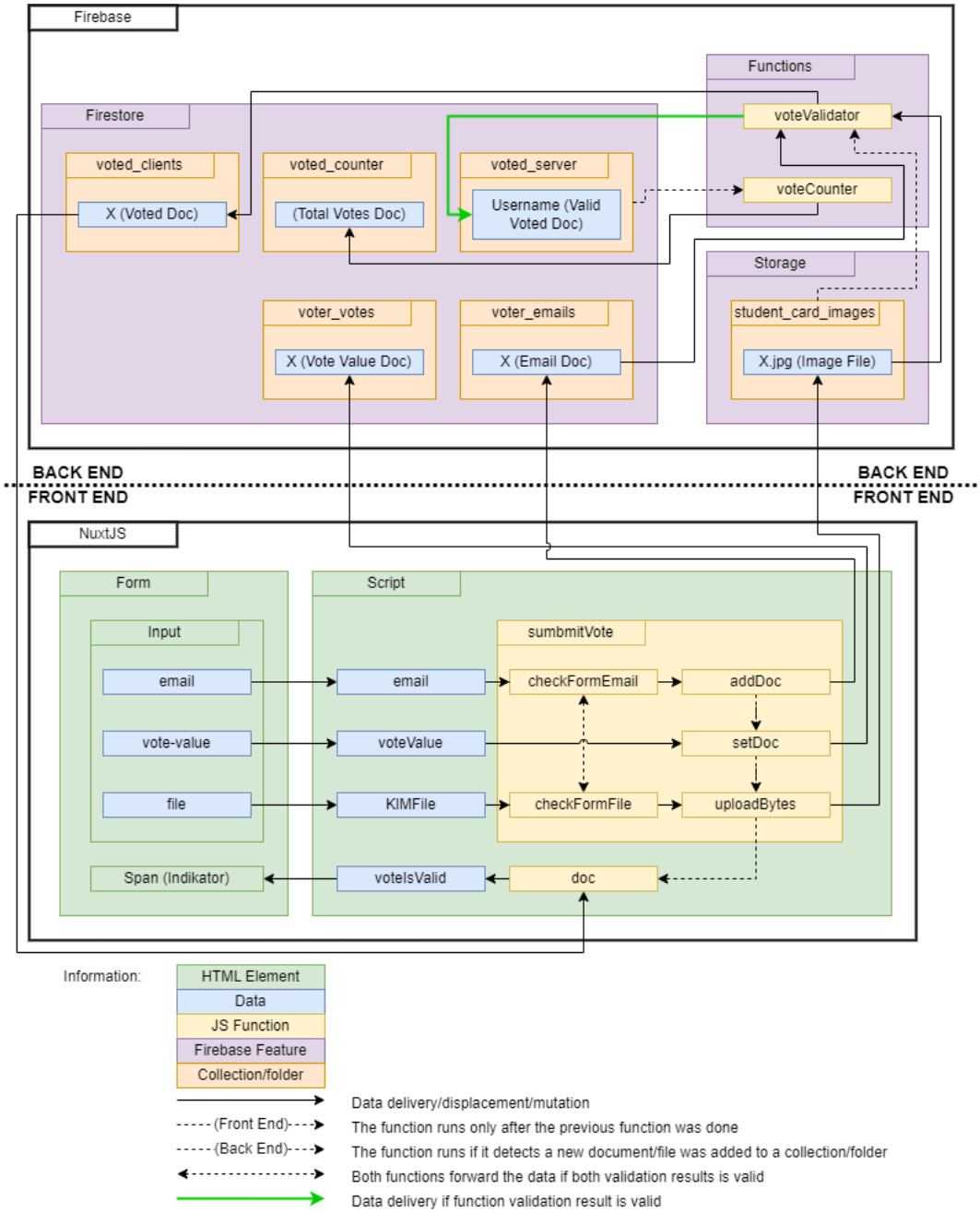

Fig. 2 Voting system front-end and back-end diagram

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i3.13734 p-ISSN : 2541-044X

The Cloud Functions function that's responsible for validating the votes has seven validation steps. It's configured using if else statement in the script. The Vote Validator Function flow chart can be seen on figure. The function is triggered every time a file has been uploaded to a specific Google Cloud Storage bucket. The first validation step checks if the uploaded file is in jpg or jpeg format. If it is in one of those formats, the second step will use Google Cloud Vision AI Text Detection OCR model to check if there is any text in the image. Other method, such as the one researched by (Boutounte & Ouadid, 2021) used key points and convolutional neural network (CNN) to train the OCR model, which was proven to have greater efficiency than traditional method. A research by (Setiawan et al., 2023) also used other method for OCR, which was by using OpenCV, an open source computer vision library for Python. The authors used Cloud Vision API because it's more convenient than training and deploying a model from scratch. If the image file contains text, the third step will split the text based on every new line character. The OCR model would create a new line for every text group from the image. Then the third step checks if there are between seven and ten text groups. For example, the third step would have a valid output if there were eight or nine text groups. The fourth step checks if all the texts from every text group match the official XU KIM format. The fifth step uses Google Cloud Vision AI Face Detection model to check if there is one and only face on the image. Since the image file name is the same as the document name that was used to store the student email and vote value in Firestore, the sixth step checks if there is a student email document that has the same name as the KIM image file. If there is, the seventh step will check KIM identity matches the email identity from the user input. If it matches, the eight-validation step will check if a user with that identity has already voted before. If there hasn't been any user with that identity who voted before, the vote would've been valid and added by the Vote Counter function.

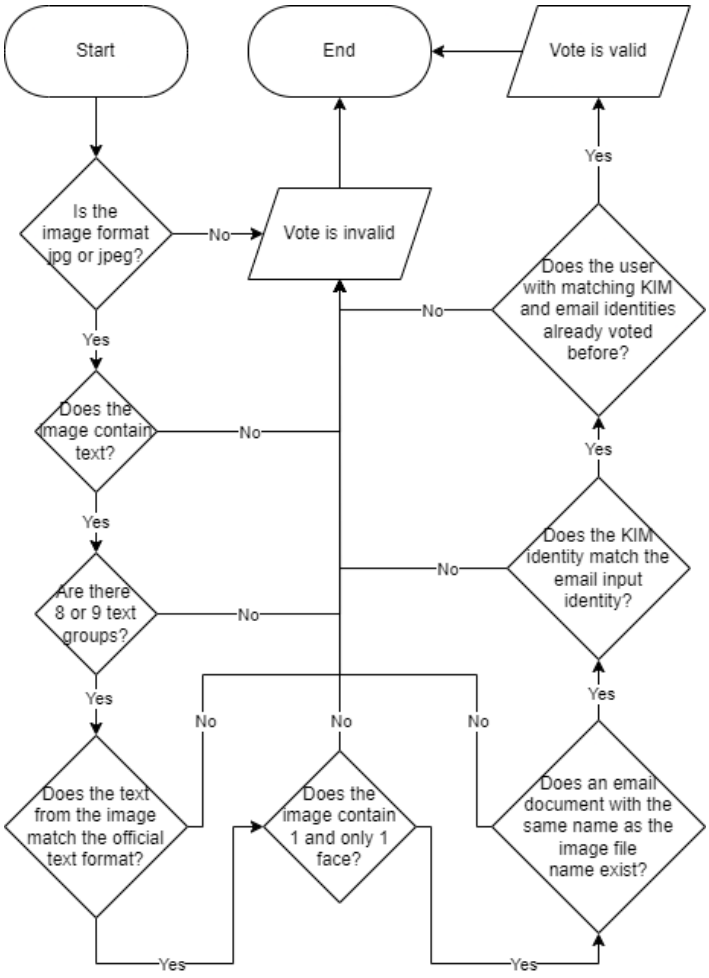

Fig. 3 Vote Validator function flowchart

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i3.13734 p-ISSN : 2541-044X

The only other page which has UI input feature is the contact us page. It has a form for submitting messages by the user. The form has name, email, subject, message, and type fields. The type field is a checkbox input to determine if the message is a student's aspiration or not. If it is, the form will require the user to input an XU student email format in the email field.

The rest of all the pages is filled with information about the student organization as mentioned before. It uses server-side rendering (SSR) and partial hydration for dynamic data. The data is stored on Firestore except the image which is statically stored in the public directory when hosted. For example, if an organization member information in Firestore changes, it will be automatically updated on the front-end without the need for the user to refresh the page.

For testing, the authors used a few different methods. The first method was using Lighthouse to test performance, accessibility, best practices, and search engine optimization (SEO). Lighthouse was used to compare front-end JS frameworks in a research by (Siahaan & Vianto, 2022), one of which was Vue, which is the framework that Nuxt is based on. The second method uses a script to stress test the voting system by submitting the form repeatedly for 50, 100, 500, 1000, and 2000 times.

The authors also tried to find the ideal file size for the KIM image. First the authors compressed the original KIM image which the system could extract its content successfully. The original image was compressed to a variety of sizes by adjusting the export resolution and quality using Adobe Photoshop. The compressed images were then used in the voting system and the output would be tested to see if it was still valid or not.

RESULT

These are the results of the student organization website concept which was made by the authors. Fig. 4 below show the top of the home page.

Fig. 4 Home page (https://ormawaxyz.id/)

Table 2 below shows the Lighthouse reports of all the pages. This test method was run by using the development tools in the internet browser. The Lighthouse report was run using navigation mode and desktop device options, also the display that were used was1920px wide.

Table 2 Website pages Lighthouse report results

Pages Name	Lighthouse Report Scores				
	Performance	Accessibility	Best Practices	SEO	Average
Home	96	95	100	100	97.75
Profile	99	94	100	100	98.25
Program	94	93	100	100	96.75
Individual program	95	94	100	100	97.25
Contact us	95	94	100	100	97.25
Election	98	95	100	100	98.25
Average	96.17	94.17	100	100	97.58

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i3.13734 p-ISSN : 2541-044X

Table 3 show the result of the vote system stress test. As mentioned before, this test uses a script to repeatedly submit the vote form so the system must validate different submissions at the same time. In the result column, "passed" means all the submissions validated correctly and successfully "failed" means otherwise. The Voted Clients Collection that stores the vote validation results can be seen on Fig. 5. From that figure, it can be seen all the unique ID of the documents every time a student votes.

Table 3 Vote system stress test results

Number of Submissions	Result
50	Passed
100	Passed
500	Passed
1000	Passed
2000	Passed

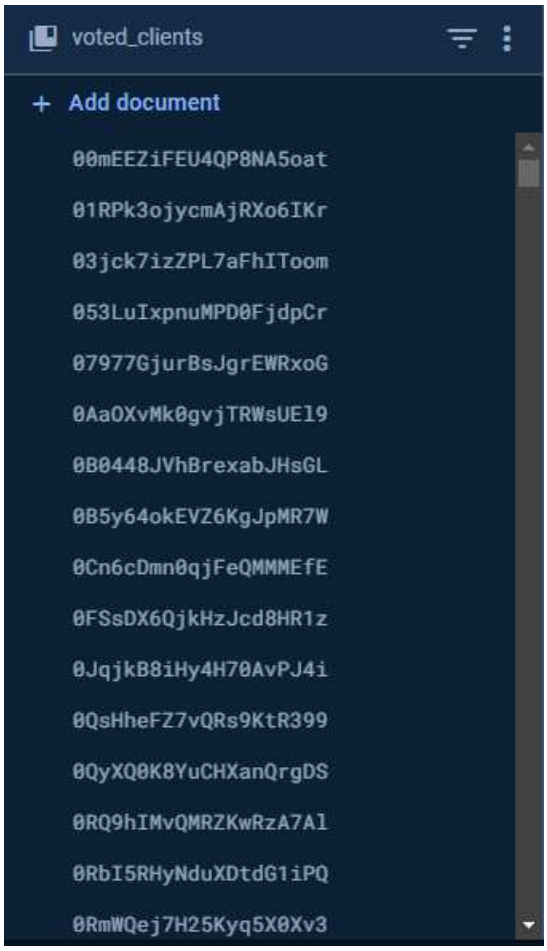

Fig. 5 Voted Clients Collection after stress test

As mentioned before, the authors want to find an optimal KIM file size so it could be stored cost efficiently. The result for each compressed file of the original image with different resolution and quality options can be seen on Table 4 below. In the result column, "passed" means that the KIM image is clear enough thus its content could be extracted perfectly by the Text Detection and the Face Detection model.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 3, July 2024
DOI : https://doi.org/10.33395/sinkron.v8i3.13734
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Table 4 Compressed images readability test results

Export Option		File Size	Result
Resolution	**Quality**		
100% (2532px * 1579px)	7	866 KB	Passed
	6	456 KB	Passed
	5	330 KB	Passed
	4	221 KB	Passed
	3	146 KB	Passed
	2	129 KB	Passed
	1	85 KB	Passed
90% (2279px * 1421px)	7	1244 KB	Passed
	6	534 KB	Passed
	5	314 KB	Passed
	4	180 KB	Passed
	3	127 KB	Passed
	2	114 KB	Passed
	1	74 KB	Passed
80% (2026px * 1263px)	7	1028 KB	Passed
	6	454 KB	Passed
	5	268 KB	Passed
	4	154 KB	Failed
	3	108 KB	Failed
	2	98 KB	Failed
	1	62 KB	Failed

DISCUSSIONS

The results obtained from the Lighthouse test were good (above 90). According to a research by (Siahaan & Kenidy, 2023), the Lighthouse test performance using Nuxt is about 72, which was calculated by an online Lighthouse score calculator. The comparison of the previous research test results and the current research average test results can be seen on Table 5. As seen from that table, the website that was developed in the current research is significantly faster. The performance, accessibility, best practices, and SEO scores suggest the XU student organization website can be used by the average users without any issues in those categories. These results also show that Nuxt and Firebase can be used together to make a great performing website. Although the results are great, the performance and accessibility scores could still be improved. The performance score can be improved by reducing the size of each page. Since Tailwind was used to make the website, the class attribute of most of the HTML elements is bloated with class names. Tailwind is a utility-first responsive CSS framework and it provides developers full privilege for customization (Salmi, 2023). These class names can be reduced by using a simpler UI design. Other issues that are affecting performance are images long render time. This could be solved by compressing the images more or by showing the heavily compressed images first and then showing the original images when it finished rendering. On the other hand, the accessibility score can be improved by increasing the contrast between texts and their background. Although the results scores from the Lighthouse test were considered good it has its limitations. The accessibility test has some additional items to be manually checked by developers. Other than that, the test was only run on a desktop device, not including mobile. Therefore, it maybe has lower scores on mobile devices, especially the performance score.

Table 5 Results comparison of the previous and current research

Metrics	Previous Research	Current Research (Average)
First Contentful Paint	0.7s	0.48s
Largest Contentful Paint	7.7s	1.2s
Total Blocking Time	1.6s	0s
Cumulative Layout Shift	0	0.0365
Speed Index	4.8s	1.03s

The results from the vote system stress test suggest the system can work perfectly fine for XU which has around 7000 active voters. The XU active voters who are also XU students would be unlikely to vote at the same time. Even if all the students vote at the same time, the verification system would very likely be having no issues because it uses Firebase Cloud Functions, which is a part of Google Cloud that scales automatically. The authors suggest that the voting system should accept submission for a day to be ideal for the voters and the system.

The results from the compressed image readability test suggest that an image resolution is more crucial than its quality (in terms of Adobe Photoshop export options). One thing that can be concluded is that the content of all the images that have a file size above 250KB can be read by the voting system perfectly. With that information, the authors set the maximum file size on the vote form to be 500KB to make sure most of the user inputs won't have file readability issues. The readability issues could have been a KIM photo that has a large angle to the horizontal axis, a KIM file that has poor compression algorithm, or a KIM photo that has too low of a resolution.

CONCLUSION

From this research, the authors have a few conclusions. A great performing student organization website that has a voting system can be achieved by using Nuxt as a front-end JS framework and Firebase as a back-end as a service. Although it's great the website can be improved by minimizing its HTML file size and reducing its image quality to reduce render time. The contrast between the website text and its background should also be increased to help people with difficulty in seeing. The voting system validation itself can be improved by adding another step of verification via the corresponding student email. The last conclusion from this research is for the Google Cloud Vision AI Text Detection and Face Detection model, it's best to use an image size above 250KB.

REFERENCES

Bahtiar Semma, A., Ali, M., Saerozi, M., Mansur, M., & Kusrini, K. (2023). Cloud computing: Google firebase firestore optimization analysis. *Indonesian Journal of Electrical Engineering and Computer Science*, *29*(3), 1719. https://doi.org/10.11591/ijeecs.v29.i3.pp1719-1728

Boutounte, M., & Ouadid, Y. (2021). Characters recognition using keys points and convolutional neural network. *Indonesian Journal of Electrical Engineering and Computer Science*, *22*(3), 1629. https://doi.org/10.11591/ijeecs.v22.i3.pp1629-1634

Fortuna, I., & Khaeruzzaman, Y. (2022). Implementation of OCR and Face Recognition on Mobile Based Voting System Application in Indonesia. *IJNMT (International Journal of New Media Technology)*, 20–27. https://doi.org/10.31937/ijnmt.v9i1.2658

Jelajahi. (2024). [Education]. Unika Atma Jaya. https://www.atmajaya.ac.id/id/

Karthikeyan, D., P., A. V., Surendhirababu, K., Selvakumar, K., Divya, P., Suhasini, P., & Palanisamy, R. (2021). Sophisticated and modernized library running system with OCR algorithm using IoT. *Indonesian Journal of Electrical Engineering and Computer Science*, *24*(3), 1680. https://doi.org/10.11591/ijeecs.v24.i3.pp1680-1691

Mishra, D. P., Rout, K. K., & Salkuti, S. R. (2021). Modern tools and current trends in web-development. *Indonesian Journal of Electrical Engineering and Computer Science*, *24*(2), 978. https://doi.org/10.11591/ijeecs.v24.i2.pp978-985

Mumtahana, H. A. (2022). Optimization of Transaction Database Design with MySQL and MongoDB. *SinkrOn*, *7*(3), 883–890. https://doi.org/10.33395/sinkron.v7i3.11528

Pawar, B. M., Patode, S. H., Potbhare, Y. R., & Mohota, N. A. (2020). An Efficient and Secure Students Online Voting Application. *2020 Fourth International Conference on Inventive Systems and Control (ICISC)*, 1–4. https://doi.org/10.1109/ICISC47916.2020.9171063

Salmi, H. A. (2023). Comparative CSS frameworks. *Multi-Knowledge Electronic Comprehensive Journal For Education And Science Publications*, *66*, 35.

Setiawan, F. B., Muntaha, I., Pratomo, L. H., & Riyadi, S. (2023). Pattern Recognition on Automated Guided Vehicles Two Wheel Drive (AGV 2WD) Robot for Location Detection Based on Raspberry Pi 4 Model B. *Sinkron*, *8*(1), 338–347. https://doi.org/10.33395/sinkron.v8i1.11990

Siahaan, M., & Kenidy, R. (2023). Rendering performance comparison of react, vue, next, and nuxt. *Jurnal Mantik*, *7*(3), 10.

Siahaan, M., & Vianto, V. O. (2022). Comparative Analysis Study of Front-End JavaScript Frameworks Performance Using Lighthouse Tool. *Jurnal Mantik*, *6*(3), 7.

Universitas Katolik Indonesia Atma Jaya. (2024). [Government]. PDDikti - Pangkalan Data Pendidikan Tinggi. https://pddikti.kemdikbud.go.id/data_pt/NUM1MEIxNEQtN0VBMC00QzBELUEzRTAtOEQ0QzdGNTAyNEY4

Yusoff, Z. M., Yusnoor, Y., Markom, A. M., Nordin, S. A., & Ismail, N. (2023). Fingerprint biometric voting machine using internet of things. *Indonesian Journal of Electrical Engineering and Computer Science*, *30*(2), 699. https://doi.org/10.11591/ijeecs.v30.i2.pp699-706

Zholdas, N., Postolache, O., Mansurova, M., Belgibaev, B., Kunelbayev, M., & Sarsembayeva, T. (2024). Development of a wearable monitor to identify stress levels using internet of things. *Indonesian Journal of Electrical Engineering and Computer Science*, *33*(3), 1486. https://doi.org/10.11591/ijeecs.v33.i3.pp1486-1499