Computer Network Design and Applications for West Java Legislative Elections

M L L Usman[1], A A Supriyadi[2], S D R Wijaya[3], B Herdianto[4]

[1] Institut Teknologi Telkom Purwokerto

[2-3] Universitas Pertahanan Republik Indonesia

[4] Lab STICC IMT atlantique, Brest, France

E-mail: lulu@ittelkom-pwt.ac.id[1], aadangsupriyadi@idu.ac.id[2], satriadhaniswara.rahsawijaya@gmail.com[3], bachtiar.herdianto@imt-atlantique.fr[4]

Abstract. The Legislative Election or Pileg is an annual plan held in Indonesia for five years. The concept of technology 4.0 has a huge opportunity to be applied in the legislative election process in this modern era. Legislative elections that use connected digital devices have more advantages than general elections that use conventional methods. The design of the legislative election system is made using a network where each TPS will be provided with one server to accommodate votes at the TPS. Then the server synchronizes the data to a central server to calculate the final results obtained. The application is web-based, and the programming languages used are HTML, PHP, CSS, and Javascript. This study uses the Software Development Life Cycle (SDLC) with the Waterfall model to develop BlackBox testing. The results of this study are a prototype of the legislative election application with valid BlackBox testing.

Keywords: Legislative Election; Application; Server; Web

1. Introduction

In the current era, changes in information technology change people's habits and governance [1]. The challenges of the technological era 4.0 as the development of modern civilization impact various aspects of life. The core concept of technology 4.0 is to combine automation technology with cyber technology connected and integrated via the internet. Artificial intelligence, automation, big data, the Internet of Things, and Cloud Computing, which created the 4.0 technology era, resulted in speedy changes. The rapid transition of the technological revolution influences human habits in carrying out their daily activities.
Information and communication technology has the potential to transform social organizations. Information and communication technology is used to collect, store, manipulate and transmit knowledge efficiently; technology also supports more efficient forms of organizational coordination and control and replaces traditional forms [2]. Information technology in the government sector can be an opportunity to improve service quality and effectiveness and cut budgets [3].

There are several studies on elections using e-voting, such as "Electronic Voting Using Blockchain And Smart Contracts: Proof Of Concept"[4], "On the Design and Implementation of a Blockchain-Enabled E-Voting Application Within IoT-Oriented Smart Cities"[5], and “An Electronic Voting Scheme Based on

Revised-SVRM and Confirmation Numbers"[6]. More details about the previous research can be seen in Table 1.

Table 1. Details of previous research

Title	Result	Research gap
Electronic Voting Using Blockchain And Smart Contracts: Proof Of Concept[4]	The concept of implementing Blockchain and smart contracts to conduct elections. The data obtained from the voting system is currently being simulated to evaluate the Blockchain.	This research is still in the form of a concept using simulation data from an existing system.
On the Design and Implementation of a Blockchain-Enabled E-Voting Application Within IoT-Oriented Smart Cities[5]	Introduced a secure and transparent e-voting mechanism through IoT devices using blockchain technology to detect and resolve the various threats caused by an intruder at multiple levels. Validation of the mechanism using analysis of various threats such as message alteration, Denial of Service (DoS), and Distributed Denial of Service (DDoS) attack and authentication delay	This research is still in the form of a scheme and needs to be realized as a product.
An Electronic Voting Scheme Based on Revised-SVRM and Confirmation Numbers[6]	Use of Revised-Simplified verifiable re-encryption mixnet (R-SVRM) and Confirmation Number (CN) for electronic voting schemes by excluding zero-knowledge proof (ZKP). Using a CN in an e-voting scheme can improve S-RVRM performance by reducing the number of items in each voting form and excluding candidate information items.	This research is still testing the use of Confirmation Number (CN) against Revised-Simplified verifiable re-encryption mixnet (R-SVRM) and has not been tested in e-voting applications that have implemented CN.

Legislative elections is an annual schedule routinely held in Indonesia for five years. The concept of technology 4.0 has a huge opportunity to be applied in the legislative election process in this modern era. Legislative elections that use digital devices in a system via the internet have more advantages than legislative elections that use conventional methods. The application of technology 4.0 in legislative elections has advantages in ineffectiveness, speed, and distribution of data, vote counting, and cost. For this reason, a general election system design is made to increase the effectiveness of general elections in these cases, namely the general election for the West Java DPRD. This research uses an Software Development Life Cycle (SDLC) system with a Waterfall model. Tests are carried out using BlackBox testing to test the suitability of the application with application access.

2. Literatur Review

2.1. General Election

The practice of democracy has the potential to encourage a country where general elections are one of the practices in democracy[7]. General elections held in Indonesia are carried out to elect members of the People's Representative Council, Level I Regional People's Representative Council, and Level II Regional People's Representative Council in the future referred to as DPR, DPRD I, and DPRD II, except for members of DPR, DPRD I, and DPRD II of the Armed Forces of the Republic of Indonesia (ABRI)[8], [9].

In the history of the general election for the president was first held in 2004 to elect the President and Vice President, wherein in the previous period (1999).. The general election in 2004 was a turning point in a democracy where the people were directly involved.

General elections, often abbreviated as elections in general, can be interpreted as a means of public participation in choosing people's representatives and become a form of democracy in society [10]. Elections allow people to choose their representatives and express how they will be governed [6]. The anonymity of the vote must be maintained where a person's choice must not be known by someone to preserve the nature of confidentiality [10]. This follows the principle of election: direct, general, accessible, honest, and fair [9].

2.2. Election Digitalization

E-voting is a system for digitally voting to assist decision-making more efficiently and effectively [5], [11]. The use of digital democracy has begun to be widely used, especially with the developments in the computer world, so that the development of traditional democracy into a more modern digital democracy has become a very developing issue. In digital democracy itself, several terms are usually known, one of which is E-Voting, where the voting process and vote counting use electronic devices [12].

3. Research Method

The research method uses the Software Development Life Cycles (SDLC) system with the Waterfall Model. The waterfall model was chosen because it is suitable at the beginning of the process, has a more superficial system complexity, and has a lower cost[13]. For self-testing, it is done using BlackBox testing. The black box testing methodology consists of seven main steps [14]: 1) System Requirement, 2) Software Requirement, 3) Analysis, 4) Program Design, 5) Coding, 6) Testing, 7) Operations. The Waterfall Model[15] has steps, as shown in Figure 1 Waterfall Model.

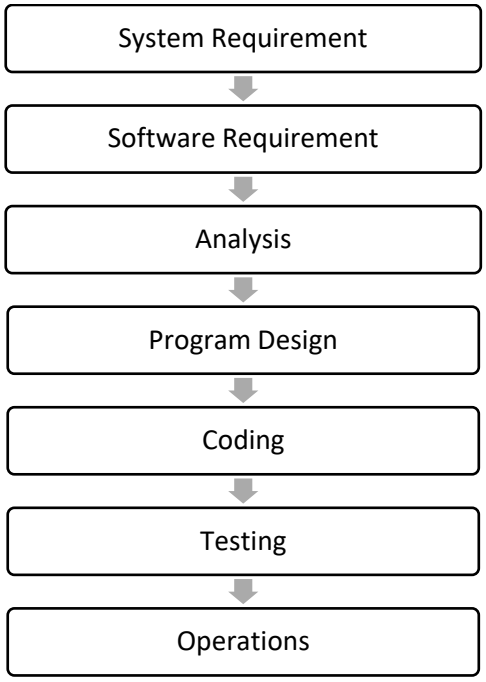

Figure 1. Waterfall Model[15]

3.1. System Requirement

At the system requirements stage, the author analyzes what system requirements are desired and what methods are needed. Analysis of system requirements, including the application, will run and based on

what. Other system requirements, such as the server used, whether the system uses an internet connection or not, include interactions between systems defined at this stage.

3.2. Software Requirement

At the software requirements stage, the author defines what software will be used to build the system based on the results at the system requirements stage. Software requirements including to perform system design or at the time of system creation.

3.3. Analysis

In the analysis stage, the writer analyzes the legislative election system, including the election flow and calculation method, by converting it into a logical flow that the system can translate. Supporting data such as the number of parties, candidates, and electoral districts are also collected at this stage.

3.4. Program Design

At the program design stage, the author makes the application flowchart based on the analysis results carried out in the previous step. A flowchart defines the workflow of the system to be created.

3.5. Coding

At the coding stage, the author translates the results of the design program into source code which can then be run as a web system. The languages used are PHP, HTML, MySQL, CSS, and JavaScript.

3.6. Testing

At the testing stage, testing is carried out to find out whether the system that has been built can work well or not. Testing uses Black Box Testing to determine whether the system is running well or not.

3.7. Operations

The operation stage is carried out to implement the results of the system, but because this is a research program, the system at this stage is made into a scientific paper.

A group of interconnected computers can be via a wired or wireless network. Computer networks also have functions for sharing resources such as sharing files, sharing printers, or sharing Internet Protocol or IP functions automatically (DHCP).

Applications are ready to use and can carry out functions and purposes following what is desired by the application maker [8]. Each application has its role according to the purpose of the maker of the application. Applications are also made with a specific definition to facilitate users doing a particular job.

4. Result and Discussion

The computer network is built with a star topology with one access point as the center. The TPS server can also be connected to the internet to perform on the KPU's central server, but the client can only click on the access point to access the TPS. For more details, see Figure *2*.

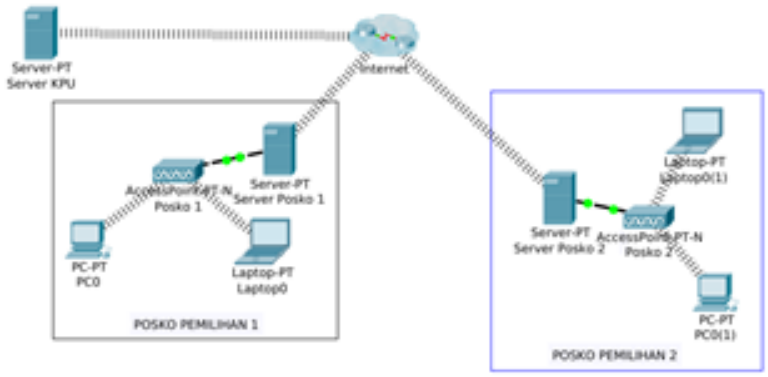

Figure 2. Computer Network Architecture

The general election application is built web-based using the PHP, HTML, CSS, and Javascript programming languages. In terms of appearance, the application used is made with the bootstrap framework. Applications are built and stored on a server created using a text-based Linux Ubuntu server operating system. The minimum server specifications can be seen in Table 2.

Table 2. The Minimum Server Specifications

Component	Specification
Hard disk	Minimum 8 GB
Ram	Minimum 128 GB
Processor	Minimum Single Core
NIC	Minimum 2

The election flow is for voters to come and then verify the data using an e-KTP or other identity. Then the officer gives an attendance signature, and then the officer provides Prototype fixes and improvements a token code that voters can use. Then, the voter makes an election using the officer's device. After voting, voters can leave the location of the general election. For clarity, the selection flow can be seen in Figure 3.

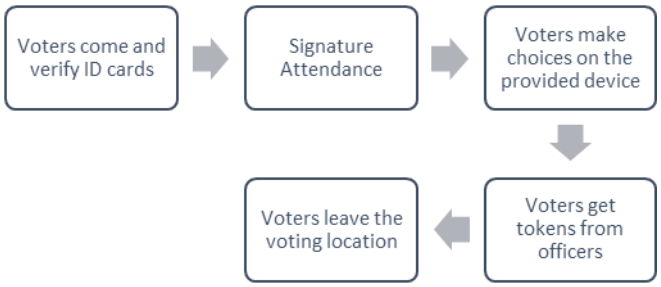

Figure 3. Selection Flow

The election flow in the application is where voters log in by entering their NIK and token code, then choose one of the legislative or party candidates by clicking the vote button. Choose a legislative candidate or party, and then the voter logs out. More details on the application flow can be seen in Figure 4.

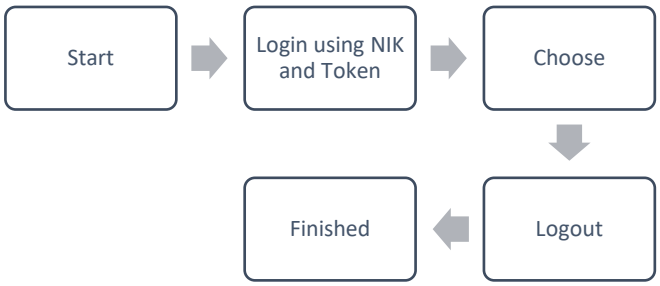

Figure 4. The Application Flow

The officers first synchronize the server management flow when the central KPU determines the synchronization time to download data to each TPS by entering the TPS code and password. During the election, the officer showed the witness that the number of incoming votes on the server was 0, and then the access was opened so that the election could be carried out. At the time of the election, the officer on duty is obliged to give tokens to voters to make an election. After the election was completed, the officers closed the access again, witnessed by witnesses and other components. The calculation is carried out by clicking the process button for the qualifying party to filter out the qualifying party, then clicking the process result button to see the seat acquisition for each party and the ranking of each candidate. After the witness's results appear and are known, the officer uploads the data using the internet and sends the data or services to the district, and then provincial levels to be synchronized manually to verify the results. The server management flow can be seen in Figure 5.

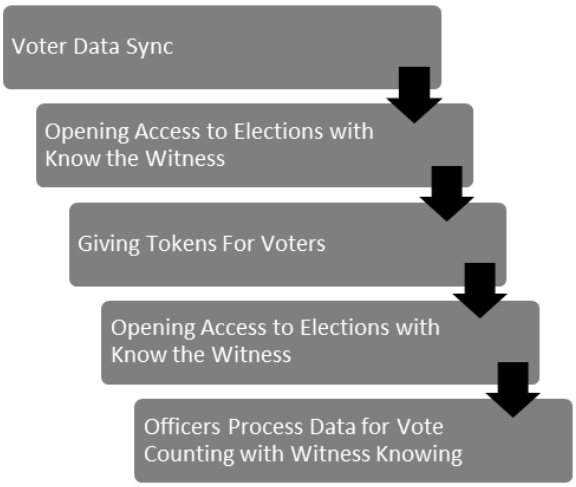

Figure 5. Server Management Flow

According to Law No. 7 of 2017, the calculation process is carried out by selecting parties that get a minimum vote of 4% to be able to participate in the seat count [8]. After obtaining the right to participate in the seat calculation, each party is calculated for seat acquisition by taking the maximum number of votes from each party and getting one seat. Then, this step is repeated with the party that gets one seat, and the total number of votes is divided by 3, 5, 7, and so on. Each time, the party gets one seat[16]–[19]. This method is also called the Sainte-League method where the division of seats in parliament uses a proportional system[20], [21]. The calculation of the Sainte-League can then be used as a formula as follows:

$$Y(n)=X(n)/(1+(2*Z(n))) \quad (1)$$

The explanation for the formula is Y(n)= The result for the total votes used in the calculation, X(n)=The development of the total votes obtained from voting, Z(n)= The seats that have been obtained from a party. Then the steps are repeated until the allocation of seats is entirely divided. This step follows Law No. 7 of 2017 articles 414 and 419 [8]. The application of the Sainte-League method in calculations can be made by calculating the following flowchart, as shown in Figure 6.

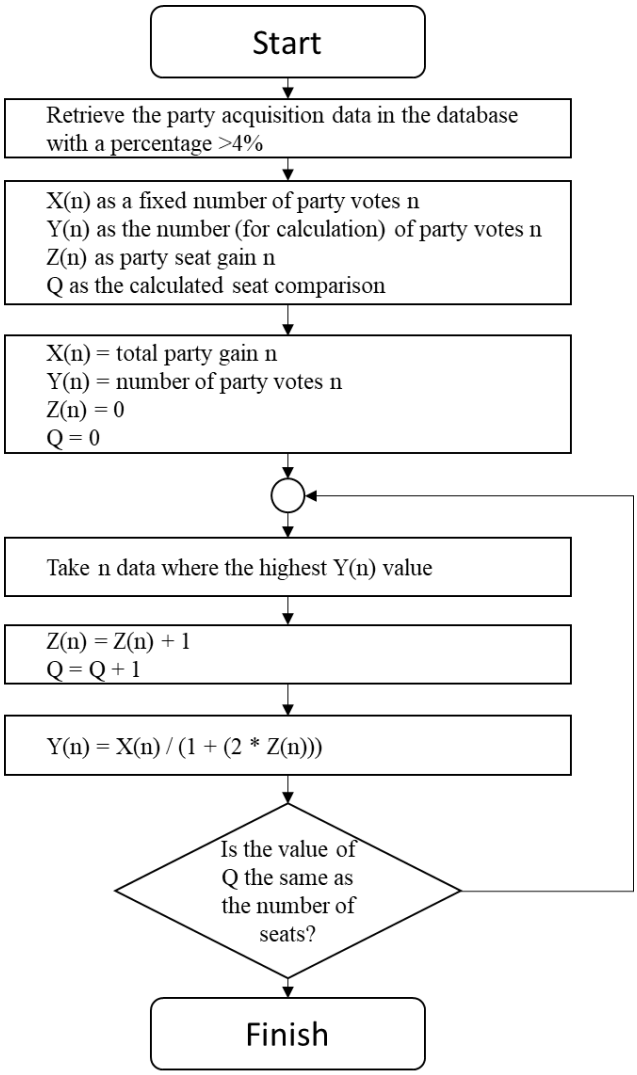

Figure 6. Application Flowchart of Sainte-League

The above flowchart can then be made a program to calculate the vote acquisition automatically. Using the PHP program, then coding can be seen in Table 3.

Table 3. The Minimum Server Specifications

Coding	Description
<?php $sql = "CALL makevariable()"; mysqli_query($con, $sql);	"Call makevariable()" is used to call a procedure to create a calculated table. Count tables are used for counting votes and processing
$jatah_kursi = 100;	Declaring the number of seats available
forketerangan($q=0; $q < $jatah_kursi; $q++) {	Repeat according to the number of seats
$sql = "SELECT id_partai from hitung where jumlah=(SELECT max(jumlah) FROM hitung) GROUP BY id_partai"; $call = mysqli_query($con, $sql); foreach ($call as $y) { $id= $y['id_partai'];	Take the party id with the largest ballots
$sql = "UPDATE kursi SET jumlah=jumlah+1 WHERE id_partai=".$id; mysqli_query($con, $sql);	Add 1 seat to the party with the largest number of seats
$sql = "update hitung set jumlah=(select jumlah/(2*(select jumlah from kursi where id_partai=".$id.")+1) from hitung_fix where id_partai=".$id.") where id_partai="**.$id;** mysqli_query($con, $sql); } } ?>	Divide the number of party votes that have already won seats by multiples of 3, 5, 7, and so on. Using the formula Y(n)=X(n)/(1+(2*Z(n)))

This application testing model uses a BlackBox test. Blackbox test is done by testing the functionality of a program based on needs and specifications [22]–[25]. BlackBox testing focuses on taking the results from the input entered [25], [26]. The election application itself has two main applications: the TPS officer segment and voters. The results of the application BlackBox test on the selector can be seen in Table 4.

Table 4. Blackbox Testing for Voters

No	Scenario	Test Case	Result	Validation
1	Login Voters	Username: Blank/False Token: Blank/False	Login failed	√
		Username: (NIK) Token: Blank/False	Login failed	√

No	Scenario	Test Case	Result	Validation
		Username: Blank/False Token: (Token)	Login failed	√
		Username: (NIK) Token: (Token)	Login Successfully	√
		Username: (NIK that has a different TPS code) Token: (Token)	Login failed	√
2	Voting	Elect a legislative candidate	Voting Successfully	√
		Elect a legislative party	Voting Successfully	√
3	Relogin	Relogin where is the voting status successful	Login failed	√
		Relogin where without voting before	Login Successfully	√

The results of the application blackbox test on TPS officers can be seen in Table 5.

Table 5. Blackbox Testing on TPS Officer

No	Scenario	Test Case	Result	Validation
1	Login Admin	Username: Blank/False Token: Blank/False	Login failed	√
		Username: (Admin user of TPS) Token: Blank/False	Login failed	√
		Username: Blank/False Token: (Password)	Login failed	√
		Username: (Admin user of TPS) Token: (Password)	Login Successfully	√
		Username: (Admin user of different TPS) Token: (Password)	Login failed	√

No	Scenario	Test Case	Result	Validation
2	Voters	Check Voters Status	Voters Status Shown	√
		Check Voters List	Voters List Shown	√
3	Voting Result	Voting results after time is over	voting results displayed	√
		Voting results before the time is over	Not shown	√
4	Upload Data	No connection	Upload failed	√
		Connected	Upload Successfully	√

The results of the application black box test on the Central KPU officers can be seen in Table 6.

Table 6. Blackbox Testing on Central KPU Application

No	Scenario	Test Case	Result	Validation
1	Login Admin	Username: Blank/False	Login failed	√
		Token: Blank/False		
		Username: (Admin user of KPU)	Login failed	√
		Token: Blank/False		
		Username: Blank/False	Login failed	√
		Token: (Password)		
		Username: (Admin user of KPU)	Login Successfully	√
		Token: (Password)		
2	Input Voters	Input data voters based on TPS	Input Successfully	√
		Check Voters List	Voters List Shown	√
3	Result	Result Checking	Checking Successfully	√

No	Scenario	Test Case	Result	Validation
		Result Calculation	Calculation Result Successfully	√
		Result Displayed	Result displayed	√

5. Conclusion

This research produces an application that can be used in web-based legislative elections in the West Java DPRD. The web is placed on a server located at a TPS. Then, the data server is synchronized using the internet and sent to the KPU server. Applications were created to assist in legislative elections by adding technology in elections that still use the traditional way to use paper. The tests show that the time required for the system to complete each stage of the legislative process is relatively faster than the conventional legislature. So it can be said that the design of the electoral system has met the requirements to face the legislative process. However, the study still has some limitations, such as the absence of studies on usability and application security. Further research is still needed to discuss the usability and security of the application.

References

[1] I. Akman, A. Yazici, A. Mishra, and A. Arifoglu, "E-Government: A global view and an empirical evaluation of some attributes of citizens," *Gov. Inf. Q.*, 2005, doi: 10.1016/j.giq.2004.12.001.

[2] D. Robey and S. Sahay, "Transforming Work Through Information Technology: A Comparative Case Study of Geographic Information Systems in County Government," *Information Systems Research*. 1996, doi: 10.1287/isre.7.1.93.

[3] F. Lin, S. S. Fofanah, and D. Liang, "Assessing citizen adoption of e-Government initiatives in Gambia: A validation of the technology acceptance model in information systems success," *Gov. Inf. Q.*, 2011, doi: 10.1016/j.giq.2010.09.004.

[4] F. D. Giraldo, C. E. Gamboa, and others, "Electronic voting using blockchain and smart contracts: Proof of concept," *IEEE Lat. Am. Trans.*, vol. 18, no. 10, pp. 1743–1751, 2020.

[5] G. Rathee, R. Iqbal, O. Waqar, and A. K. Bashir, "On the design and implementation of a blockchain enabled e-voting application within iot-oriented smart cities," *IEEE Access*, vol. 9, pp. 34165–34176, 2021.

[6] K. M. R. Alam, S. Tamura, S. M. S. Rahman, and Y. Morimoto, "An Electronic Voting Scheme Based on Revised-SVRM and Confirmation Numbers," *IEEE Trans. Dependable Secur. Comput.*, vol. 18, no. 1, pp. 400–410, 2019.

[7] J. Pickering *et al.*, "Democratising sustainability transformations: Assessing the transformative potential of democratic practices in environmental governance," *Earth Syst. Gov.*, vol. 11, p. 100131, 2022.

[8] *Undang – Undang No 7 Tahun 2017 Tentang Pemilihan Umum*. .

[9] *UNDANG-UNDANG REPUBLIK INDONESIA NOMOR 3 TAHUN 1999 TENTANG PEMILIHAN UMUM*. .

[10] T. Kohno, A. Stubblefield, A. D. Rubin, and D. S. Wallach, "Analysis of an electronic voting system," 2004, doi: 10.1109/SECPRI.2004.1301313.

[11] L. P. Alonso, M. Gasco, D. Y. M. del BLANCO, J. Á. H. Alonso, J. Barrat, and H. A. Moreton, "E-voting system evaluation based on the Council of Europe recommendations: Helios Voting," *IEEE Trans. Emerg. Top. Comput.*, vol. 9, no. 1, pp. 161–173, 2018.

[12] T. Haryadi, A. Nurmandi, I. Muallidin, D. Kurniawan, and others, "Implementing 'SIREKAP' Application Based on Election for Improving the Integrity of Election Administrators and Increasing Public Trust," in *International Conference on Human Interaction and Emerging Technologies*, 2021, pp. 159–165.

[13] R. Malhotra and A. Chug, "Comparative analysis of agile methods and iterative enhancement model in assessment of software maintenance," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 1271–1276.

[14] D. Marijan, N. Teslic, M. Temerinac, and V. Pekovic, "On the effectiveness of the system validation based on the black box testing methodology," in *2009 IEEE Circuits and Systems International Conference on Testing and Diagnosis*, 2009, pp. 1–4.

[15] S. Liu and J. Wei, "The Satelite on the Research of the Software Life Cycle: V+ Iterative Waterfall," in *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2018, pp. 1716–1720.

[16] Z. Daskalovski, "Proportional Electoral Systems and Number of Parties in Parliament Evidence from Macedonia," *Polit. Cent. Eur.*, vol. 15, no. 3, pp. 459–478.

[17] M. M. Kaminski, "Spoiler effects in proportional representation systems: evidence from eight Polish parliamentary elections, 1991--2015," *Public Choice*, vol. 176, no. 3, pp. 441–460, 2018.

[18] M. Mart\'\inez-Panero, V. Arredondo, T. Peña, and V. Ram\'\irez, "A new quota approach to electoral disproportionality," *Economies*, vol. 7, no. 1, p. 17, 2019.

[19] M. Brill, J.-F. Laslier, and P. Skowron, "Multiwinner approval rules as apportionment methods," *J. Theor. Polit.*, vol. 30, no. 3, pp. 358–382, 2018.

[20] E. Boldyreva and Y. Duisembina, "Changes in swedish foreign policy after 2018 general elections," in *Proceedings of Topical Issues in International Political Geography*, Springer, 2021, pp. 228–235.

[21] J. Medzihorsky, "Rethinking the D'Hondt method," *Polit. Res. Exch.*, vol. 1, no. 1, pp. 1–15, 2019.

[22] S. Nidhra, "Black Box and White Box Testing Techniques - A Literature Review," *Int. J. Embed. Syst. Appl.*, 2012, doi: 10.5121/ijesa.2012.2204.

[23] M. Sholeh, I. Gisfas, M. A. Fauzi, and others, "Black Box Testing on ukmbantul. com Page with Boundary Value Analysis and Equivalence Partitioning Methods," in *Journal of Physics: Conference Series*, 2021, vol. 1823, no. 1, p. 12029.

[24] G. W. Sasmito and M. Nishom, "Testing the Population Administration Website Application Using the Black Box Testing Boundary Value Analysis Method," in *2020 IEEE Conference on Open Systems (ICOS)*, 2020, pp. 48–52.

[25] J. Kang and J. H. Park, "A secure-coding and vulnerability check system based on smart-fuzzing and exploit," *Neurocomputing*, vol. 256, pp. 23–34, 2017.

[26] A. P. Estrada-Vargas, E. López-Mellado, and J.-J. Lesage, "A black-box identification method for automated discrete-event systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 3, pp. 1321–1336, 2015.