



## PEMBUATAN SISTEM APLIKASI *ELECTRONIC VOTING (E-VOTING)* BERBASIS WEB DENGAN PENDEKATAN METODE *WATERFALL*

Rendy Nurdiansyah<sup>1</sup>, Sayekti Harist Suryawan<sup>2</sup>, Abdul Rahim<sup>3</sup>

<sup>1,2,3</sup>Teknik Informatika, Universitas Muhammadiyah Kalimantan Timur  
Samarinda, Kalimantan Timur, Indonesia 75124

2011102441127@umkt.ac.id, shs500@umkt.ac.id, ar622@umkt.ac.id

### Abstract

*Electronic voting (e-voting) uses electronic devices to simplify the voting process, enabling quick selection, vote tabulation, and real-time access to information. Traditional voting methods have various drawbacks, such as high costs, risk of human error, lengthy counting processes, and vulnerability to fraud. Universitas Muhammadiyah Kalimantan Timur (UMKT) also faces similar challenges in its internal organizational elections. The e-voting system is designed to overcome these obstacles by enhancing efficiency and transparency and reducing errors through an accurate audit trail. This study aims to develop a web-based e-voting application that can address the problems associated with traditional voting methods. The application is developed using Python, the programming language of the Django framework, and MySQL for database management. The development method employed is the Waterfall model, which provides a clear structure and project schedule. The research results indicate that the black box testing of this application was successful. Furthermore, testing using the Technology Acceptance Model (TAM) shows that the Web-Based E-voting Application System has an average percentage score of 91%, indicating that most respondents agree or strongly agree with using this system. Thus, this e-voting system is expected to increase voter participation, reduce operational costs, and ensure that every vote is counted accurately while addressing the various issues faced by traditional voting methods.*

**Keywords:** *E-voting, Django, Traditional Voting, Python, Waterfall*

### Abstrak

Pemungutan suara elektronik (*e-voting*) menggunakan perangkat elektronik untuk menyederhanakan proses pemilihan, memungkinkan seleksi dan tabulasi suara yang cepat serta akses informasi secara *real-time*. Metode *voting* tradisional memiliki berbagai kelemahan, seperti biaya tinggi, risiko kesalahan manusia, proses perhitungan yang lama, dan kerentanan terhadap kecurangan. Universitas Muhammadiyah Kalimantan Timur (UMKT) juga menghadapi tantangan serupa dalam pemilihan internal organisasi mereka. Sistem *e-voting* dirancang untuk mengatasi kendala-kendala ini dengan meningkatkan efisiensi, transparansi, dan mengurangi kesalahan melalui jejak audit yang akurat. Penelitian ini bertujuan mengembangkan aplikasi *e-voting* berbasis web yang dapat mengatasi masalah-masalah *voting* tradisional. Aplikasi ini dikembangkan menggunakan bahasa pemrograman *Python* dengan *framework Django* dan *MySQL* untuk manajemen *database*. Metode pengembangan yang digunakan adalah model *Waterfall*, yang memberikan struktur dan jadwal proyek yang jelas. Hasil penelitian menunjukkan bahwa pengujian *black box testing* pada aplikasi ini berhasil dengan baik. Selain itu, pengujian menggunakan *Technology Acceptance Model (TAM)* menunjukkan bahwa Sistem Aplikasi *E-voting* Berbasis Web ini memiliki nilai persentase rata-rata sebesar 91%, yang berarti bahwa sebagian besar responden setuju atau sangat setuju terhadap penggunaan sistem ini. Dengan demikian, sistem *e-voting* ini diharapkan dapat meningkatkan partisipasi pemilih, menurunkan biaya operasional, dan memastikan setiap suara dihitung dengan akurat, sekaligus mengatasi berbagai masalah yang dihadapi dalam metode *voting* tradisional.

**Kata kunci:** *E-voting, Django, Voting Tradisional, Python, Waterfall*

### 1. PENDAHULUAN

Pemungutan suara adalah proses pengambilan keputusan dengan memilih dari antara kandidat yang telah dicalonkan. Setelah kandidat dipilih, pemungutan suara dilakukan untuk

menentukan siapa yang mendapatkan suara terbanyak, dan hasilnya ditampilkan dalam bentuk grafik[1]. Selain itu, pemungutan suara juga berperan dalam mengumpulkan aspirasi untuk mencari solusi suatu masalah[2]. Di

Indonesia, pemungutan suara dilakukan dengan mencoblos atau mencontreng pilihan, yang merupakan bagian penting dari praktik demokrasi. Sistem pemungutan suara ini dilakukan melalui mekanisme legislasi dan sistem voting tradisional non-elektronik[3].

Pemungutan suara tradisional masih memiliki berbagai masalah, seperti biaya yang tinggi, risiko kesalahan manusia yang besar, dan proses perhitungan yang memakan waktu lama, yang membebani petugas perhitungan. Selain itu, pemungutan suara tradisional juga rentan terhadap kecurangan, terutama dalam proses penghitungan suara, seperti penggelembungan hasil suara[4]. Universitas Muhammadiyah Kalimantan Timur (UMKT), sebuah lembaga pendidikan tinggi di Kalimantan Timur, juga menghadapi tantangan serupa. UMKT masih menerapkan sistem voting tradisional dalam pemilihan di berbagai organisasi internalnya.

Mengatasi masalah yang ada dalam pemungutan suara tradisional dan memanfaatkan perkembangan teknologi, dibuatlah sistem yang dapat melaksanakan kegiatan voting sekaligus menyelesaikan kendala yang ada. Sistem yang dirancang adalah *Electronic Voting (E-voting)*, yakni metode pemungutan suara yang menggunakan perangkat elektronik untuk memudahkan proses pemungutan suara[5]. *E-voting* memiliki keunggulan seperti proses seleksi dan tabulasi suara yang lebih cepat[6]. Pemilihan dan perhitungan suara dapat dilakukan secara *real-time*, di mana setiap suara yang diterima akan ditampilkan di aplikasi. Ini memudahkan akses informasi[7], sehingga mampu mengurangi biaya operasional dalam penyelenggaraan pemilihan serta meminimalkan kelemahan-kelemahan dari pemungutan suara tradisional dengan surat suara.

Penelitian yang dilakukan oleh Willyandro[8] menjelaskan bahwa salah satu keuntungan dari *e-voting* adalah kemampuannya untuk meningkatkan partisipasi pemilih dengan membuat proses pemungutan suara lebih cepat dan nyaman bagi pemilih. Selain itu, penelitian dari Krismanto[2] mendukung bahwa proses *e-voting* yang telah terkomputerisasi menjadi lebih efisien dan mudah digunakan.

Salah satu keuntungan utama *e-voting* dibandingkan metode tradisional adalah kemampuannya untuk meningkatkan transparansi dan mengurangi kesalahan melalui penggunaan jejak audit, serta *e-voting* dianggap lebih akurat[9]. Jejak audit adalah rekaman digital rinci dari setiap aktivitas atau transaksi dalam sistem dan *database*[10]. Fungsi ini sangat penting untuk memastikan integritas pemilihan, memungkinkan pemeriksaan dan verifikasi. Hal ini membantu membangun kepercayaan pemilih terhadap proses pemungutan suara dan memastikan bahwa setiap suara dihitung dengan benar.

Dalam pembuatan sistem *e-voting*, *Python* sering dipilih karena kemudahannya, cocok untuk pemula maupun kebutuhan yang kompleks[11]. *Python* juga mendukung

*Django, framework* yang menurut Jalolov[12], memberikan keuntungan seperti pengembangan cepat, skalabilitas, keamanan, fleksibilitas, dan dukungan komunitas luas. Salah satu fitur penting *Django* adalah *layer ORM*-nya, yang sangat menyederhanakan interaksi dengan berbagai jenis *database*, menjadikannya pilihan utama untuk pengembangan aplikasi web.

Dalam pengembangan aplikasi berbasis web, salah satu pendekatan yang dapat digunakan adalah model *waterfall*. Model ini merupakan salah satu pendekatan pengembangan perangkat lunak yang bersifat linear dan berurutan, di mana setiap fase harus diselesaikan sebelum melanjutkan ke fase berikutnya. Pendekatan ini memiliki keunggulan dalam struktur yang terstruktur dan dapat diprediksi dalam pengembangan perangkat lunak, memungkinkan perencanaan yang jelas, pencapaian yang terdefinisi dengan baik, dan garis waktu proyek yang mudah dipahami[13]. Oleh karena itu, model *waterfall* sangat sesuai untuk proyek-proyek seperti pembangunan aplikasi baru atau proyek dengan risiko rendah dan jadwal pengembangan yang relatif cepat[14]. Penelitian oleh Aldi[15] yang mengembangkan aplikasi penerimaan siswa baru berbasis web dengan menggunakan metode *waterfall* menunjukkan bahwa metode ini cocok untuk proyek-proyek yang memerlukan dokumentasi yang lengkap dan rinci.

Berdasarkan latar belakang yang telah dijelaskan, penelitian ini bertujuan untuk mengembangkan sebuah sistem aplikasi *e-voting* berbasis web sebagai solusi atas tantangan dalam metode voting tradisional. Penelitian ini akan mengadopsi pendekatan model *Waterfall*, menggunakan bahasa pemrograman *Python* dengan dukungan dari *framework Django*, dan *MySQL* sebagai sistem manajemen *database*. Untuk meningkatkan keamanan dan keterjangkauan sistem, akan diterapkan fitur *Login RFID Passwordless* menggunakan *NodeMCU*. Selain itu, keamanan data akan dijamin dengan menerapkan kriptografi *Rivest Shamir Adleman (RSA)* untuk enkripsi data dan *Digital Signature Algorithm (DSA)* untuk validasi tanda tangan digital, sehingga menjaga integritas dan keaslian informasi dalam proses *e-voting*.

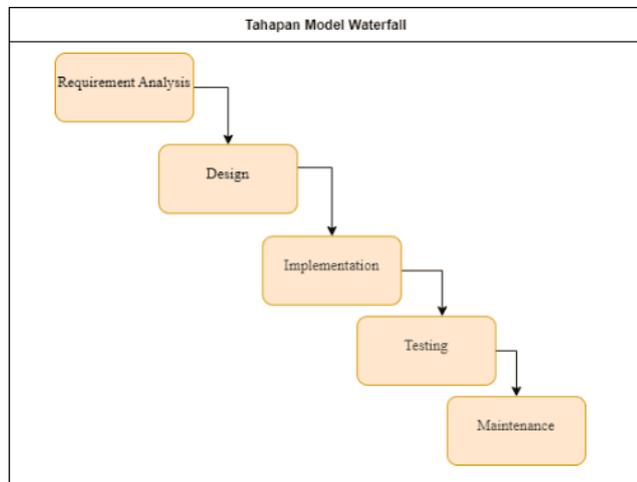
## 2. METODE PENELITIAN

Penelitian ini menggunakan metode campuran kuantitatif dan kualitatif. Pendekatan kuantitatif berfokus pada eksperimen untuk menguji dan mengevaluasi aplikasi *e-voting* berbasis web. Eksperimen ini dirancang untuk mengembangkan sistem *e-voting*, dengan menguji variabel *input* dan *output* untuk mengukur efisiensi dan efektivitas kinerja sistem. Proses eksperimental mencakup perancangan dan pembuatan komponen-komponen utama aplikasi *e-voting*, termasuk diagram alur dan pengembangan perangkat lunak. Pendekatan kualitatif difokuskan pada studi kasus di Universitas Muhammadiyah Kalimantan Timur, dengan melakukan wawancara untuk mengumpulkan data yang relevan. Metode yang digunakan dalam pengembangan sistem ini adalah metode *Waterfall*

yang merupakan bagian dari *System Development Life Cycles (SDLC)*. Model *Waterfall* terkenal dalam pengembangan perangkat lunak karena pendekatannya yang sistematis dan berurutan.

## 2.1 Tahapan Penelitian

Tahapan penelitian berdasarkan metode *waterfall* yang dilakukan dapat dilihat pada gambar 1 berikut:



Gambar 1. Tahapan Model Waterfall

### a. Analisis Kebutuhan (*Requirement Analysis*)

Tahap awal ini fokus pada pengumpulan data komprehensif tentang kebutuhan sistem *e-voting*, termasuk fungsionalitas dasar dan fitur-fitur spesifik. Prosesnya melibatkan penelitian literatur dan wawancara mendalam dengan dosen, mahasiswa, dan pemangku kepentingan lainnya untuk memastikan semua kebutuhan teridentifikasi dengan jelas sebagai dasar pengembangan sistem.

### b. Perancangan (*Desain*)

Setelah mengidentifikasi kebutuhan, langkah selanjutnya adalah merancang solusi teknis untuk sistem *e-voting*. Tahap ini mencakup pengembangan arsitektur sistem, termasuk desain *database*, antarmuka pengguna, dan pemilihan teknologi yang akan digunakan. *Wireframes* sistem *e-voting* juga dibuat menggunakan *Balsamiq Wireframes* sebagai panduan awal bagi tim pengembang.

### c. Implementasi (*Implementation*)

Tahap implementasi adalah saat di mana desain sistem dikodekan menjadi program yang berfungsi. Penelitian ini menggunakan bahasa pemrograman *Python* dengan dukungan dari *framework Django* untuk pengembangan dan integrasi fitur-fitur yang telah direncanakan. *Visual Studio Code* digunakan sebagai editor teks untuk pengkodean yang efisien.

### d. Pengujian (*Testing*)

Setelah implementasi, langkah berikutnya adalah pengujian sistem untuk memastikan fungsionalitas dan kebutuhan pengguna terpenuhi. Pengujian menggunakan *black box testing*, fokus pada fungsionalitas tanpa perlu pengetahuan detail tentang struktur atau kode sumber sistem[16]. Tujuannya adalah memastikan semua fitur berjalan dengan baik dan menghasilkan *output* sesuai dengan *input*. Selain itu, dilakukan juga pengujian dengan *Focus Group Discussion (FGD)* menggunakan kuesioner berbasis *Technology Acceptance Model (TAM)*, yang menjelaskan bagaimana persepsi pengguna mempengaruhi penggunaan teknologi. Ini memungkinkan pengembang dan *tester* memverifikasi semua aspek sistem sesuai dengan spesifikasi dan beroperasi sesuai rencana.

### e. Pemeliharaan (*Maintenance*)

Pada tahap ini, sistem telah diimplementasikan dan aktif digunakan. Kegiatan pemeliharaan meliputi perbaikan kesalahan yang tidak terdeteksi sebelumnya, peningkatan unit-unit implementasi sistem, serta penyesuaian layanan sistem dengan kebutuhan baru yang muncul. Proses ini penting untuk memastikan sistem tetap efektif, efisien, dan relevan dengan kebutuhan pengguna serta lingkungan operasional yang terus berubah.

## 3. HASIL DAN PEMBAHASAN

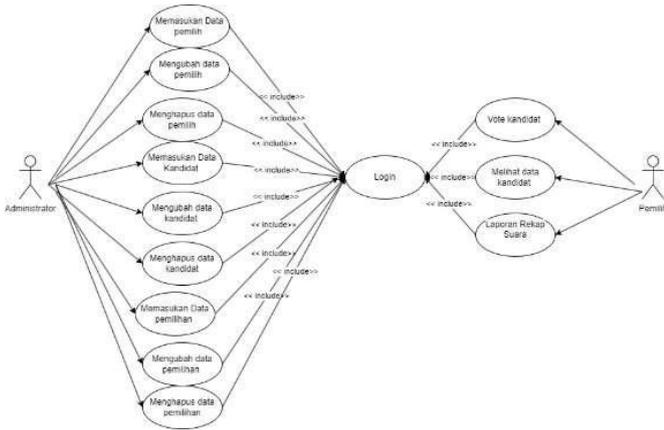
### 3.1 Analisis Kebutuhan (*Requirement Analysis*)

Berdasarkan hasil analisis kebutuhan untuk aplikasi *e-voting*, dapat disimpulkan bahwa implementasi *e-voting* menggunakan pendekatan metode *waterfall*, bahasa pemrograman *Python*, dan *framework Django* sangat relevan untuk meningkatkan efisiensi, keamanan, dan keandalan proses pemilihan di lingkungan kampus. Studi literatur menunjukkan bahwa *e-voting* dapat mengatasi kelemahan dari sistem tradisional, seperti efisiensi waktu dan biaya yang lebih rendah, sambil mempertahankan integritas dan keamanan data pemilih. Wawancara dengan narasumber menggarisbawahi keinginan akan fitur yang memperbaiki kerahasiaan, mengurangi kesalahan manusia, dan menghadirkan keamanan yang lebih baik, mencerminkan tantangan utama terkait kepercayaan dan keamanan dalam implementasi *e-voting* ini.

### 3.2 Perancangan (*Desain*)

Pada tahap hal yang dilakukan dengan pemodelan UML (*Unified Modeling Language*) yang diantaranya menggunakan *use case*, *class diagram*, *activity diagram*, dan *sequence diagram* untuk menggambarkan berbagai aspek dari sistem yang dikembangkan.

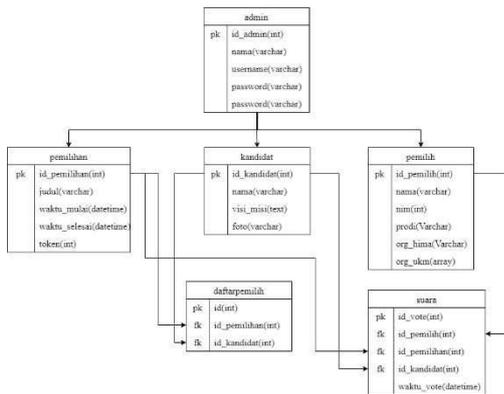
### 3.2.1 Use Case Diagram



Gambar 2. Use Case Diagram

Pada gambar 2 ini menggambarkan *use case* sistem *e-voting*. Terdapat dua aktor utama, yaitu Administrator dan Pemilih. Administrator mengelola berbagai aspek pemilihan, termasuk data pemilih, kandidat, dan hasil pemungutan suara. Pemilih dapat melakukan proses *voting*. Proses *login* menjadi pusat diagram, mengontrol akses ke fungsi-fungsi sistem.

### 3.2.2 Class Diagram

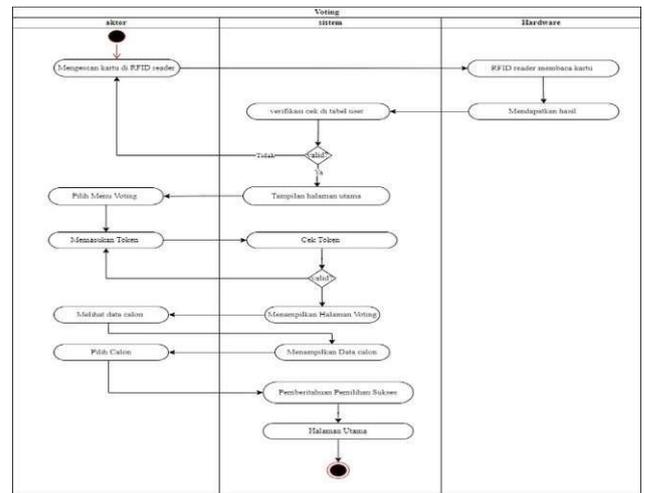


Gambar 3. Class Diagram

*Diagram Class* pada Gambar 3 ini menunjukkan struktur sistem *e-voting* dengan kelas-kelas utama, yakni Admin, Pemilihan Umum, Kandidat, Pemilih, Hasil Pemilihan, dan Suara Elektronik. Admin mengelola sistem, Pemilihan Umum mengatur proses pemilihan, Kandidat mewakili partisipan, Hasil Pemilihan menangani perhitungan, dan Suara Elektronik merepresentasikan suara digital. Setiap kelas memiliki atribut dan metode spesifik, dengan hubungan antar kelas yang menggambarkan interaksi dalam sistem. Struktur ini mendukung seluruh alur proses *e-voting* dari pendaftaran hingga penghitungan hasil.

### 3.2.3 Activity Diagram

Pada *activity diagram* menjelaskan tentang alur kerja pada sistem yang dilakukan pada perancangan sistem *e-voting* berbasis web.

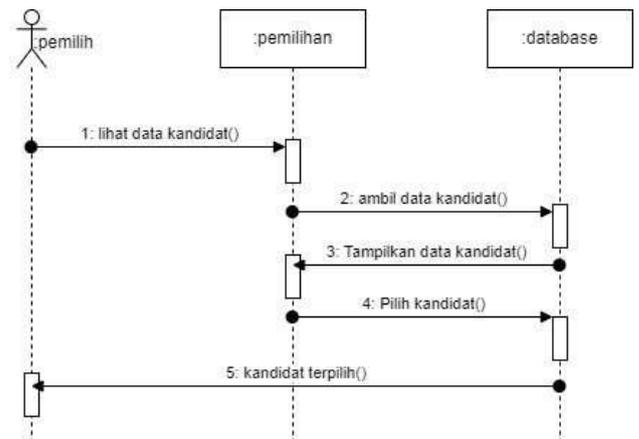


Gambar 4. Proses Voting

Diagram aktivitas seperti gambar 4 menggambarkan proses *voting* elektronik dengan tiga komponen utama yaitu pengguna, sistem, dan perangkat keras. Proses dimulai saat pengguna menggesekkan kartu RFID mereka. RFID reader membaca kartu dan mengirimkan hasilnya ke sistem. Sistem memverifikasi data pengguna dari tabel pengguna. Jika *valid*, sistem menampilkan halaman utama. Pengguna kemudian memilih menu *voting* dan memasukkan token. Sistem memeriksa validitas token. Jika *valid*, sistem menampilkan halaman *voting* dengan data calon. Pengguna memilih calon yang diinginkan, dan sistem menampilkan data calon tersebut. Setelah selesai, sistem memberikan pemberitahuan bahwa pemilihan berhasil dan mengembalikan pengguna ke halaman utama.

### 3.2.4 Sequence Diagram

*Sequence diagram* disusun berdasarkan urutan waktu dan digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu.



Gambar 5. Sequence Diagram Proses Voting

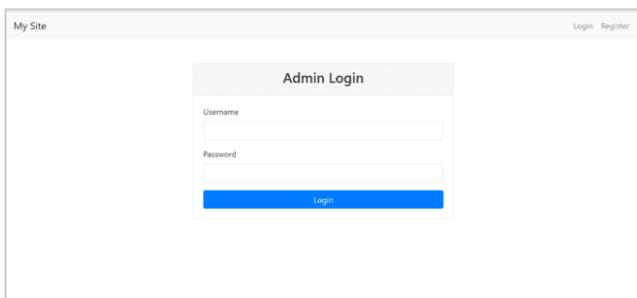
Gambar 5 menjelaskan alur proses *voting* dalam sistem pemilihan. Proses dimulai ketika pemilih meminta data kandidat. Sistem pemilihan mengakses *database* untuk mengambil data kandidat dan menampilkannya kepada

pemilih. Pemilih memilih kandidat dan mengirimkan pilihannya kembali ke sistem. Sistem mengkonfirmasi pilihan tersebut kepada pemilih, menyelesaikan proses voting.

### 3.3 Implementasi (*Implementation*)

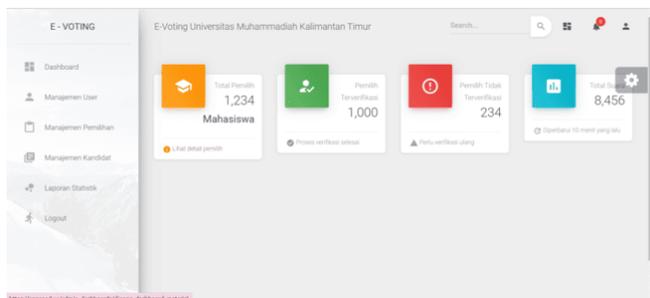
Setelah tahap desain sistem selesai, maka memasuki tahap implementasi, di mana desain tersebut diubah menjadi aplikasi *e-voting* berbasis web yang fungsional. Pada tahap ini, dilakukan pengembangan dan integrasi berbagai komponen sistem, khususnya fitur-fitur pada bagian admin dan pemilih.

#### 3.3.1 Tampilan Administrator



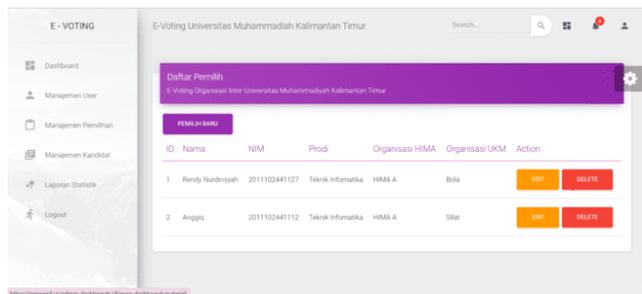
Gambar 6. Login Admin

Gambar 6 adalah tampilan *login* untuk administrator, yang dimana seorang admin memasukkan *username* dan *password* yang benar dan sesuai dengan *database*, setelah *username* dan *password* yang di masukan sesuai maka akan masuk ke dalam halaman *dashboard*.



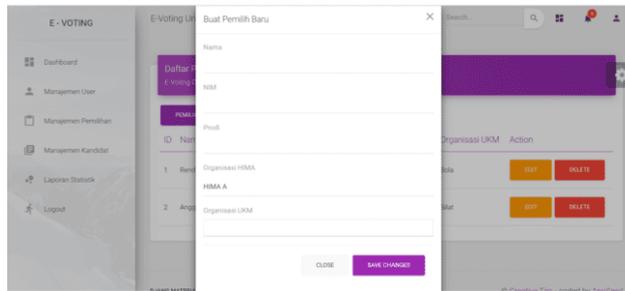
Gambar 7. Dashboard Admin

Pada gambar 7 adalah halaman yang muncul setelah admin melakukan *login*, di dalamnya terdapat menu serta data-data dari pemilihan yang ada dalam sistem.



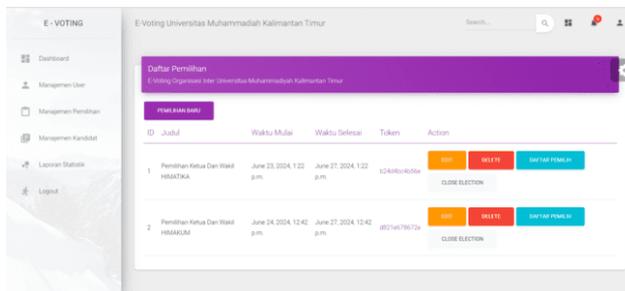
Gambar 8. Manajemen Pemilih

Pada gambar 8 Halaman Manajemen Pemilih yang mana terdapat daftar pemilih, pada halaman ini juga administrator bisa menambah pemilih baru, mengedit pemilih dan menghapus pemilih.



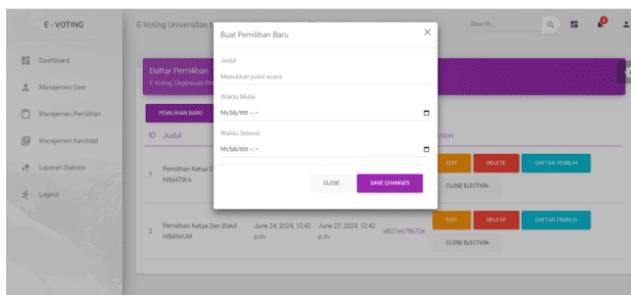
Gambar 9. Form Tambah Pemilih

Gambar 9 adalah bentuk *Form* untuk menambah data pemilih baru, yang di-*input* adalah nama, NIM, Prodi, Organisasi HIMA dan Organisasi UKM.



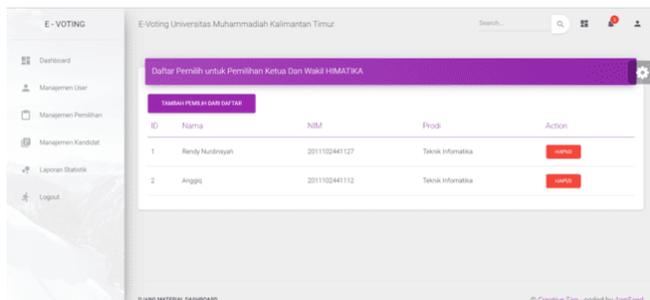
Gambar 10. Manajemen Pemilihan

Halaman Manajemen Pemilihan pada gambar 10 yang mana terdapat daftar pemilihan yang berlangsung, pada halaman ini administrator bisa menambah pemilihan baru, mengedit pemilihan, menghapus pemilihan yang ada, ada *button* untuk ke halaman daftar pemilih yang berhak memilih pada pemilihan tersebut dan ada *button* untuk menutup pemilihan serta membuka pemilihan sesuai dengan kondisi.



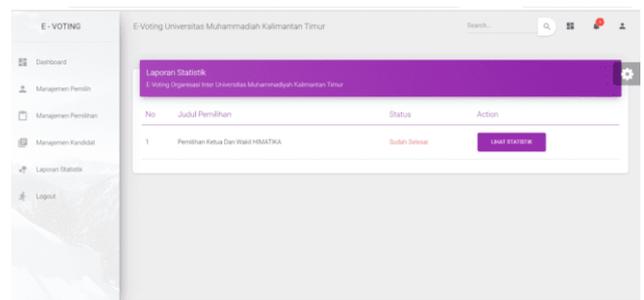
Gambar 11. Form Tambah Pemilihan

Gambar 11 menunjukkan *form* tambah pemilihan, yang mana bisa menambahkan pemilihan dengan yang di isi adalah judul pemilihan, waktu mulai dan waktu selesai, setelah pemilihan berhasil dibuat.



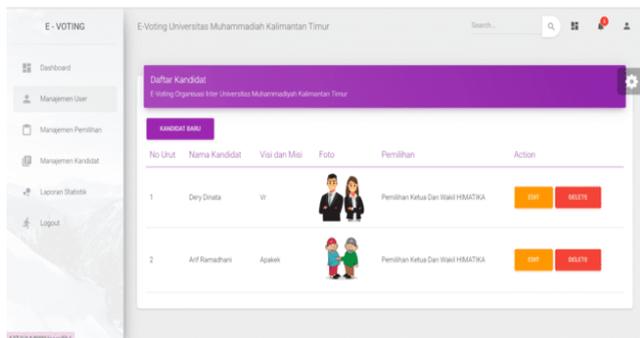
Gambar 12. Daftar Pemilih Terpilih

Pada gambar 12 adalah halaman daftar pemilih yang dimana terdapat pemilih yang berhak melakukan *voting* pada pemilihan tertentu, pada halaman ini bisa menambahkan pemilih yang diinginkan.



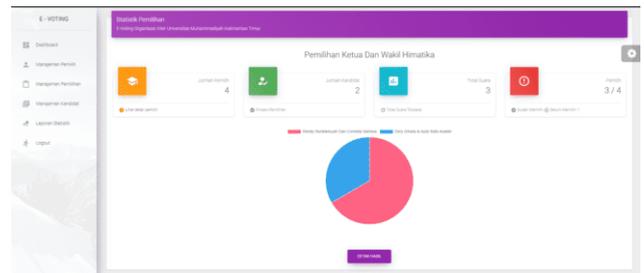
Gambar 15. Laporan Statistik

Gambar 15 di atas adalah daftar pemilihan dan kita bisa melihat statistik pemilihannya, akan tetapi untuk bisa melihat statistik pemilihan harus sudah selesai terlebih dahulu.



Gambar 13. Manajemen Kandidat

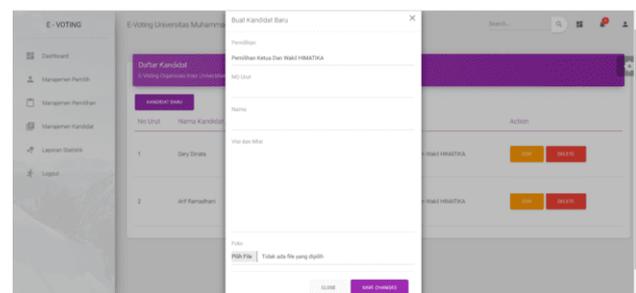
Gambar 13 adalah halaman untuk manajemen data kandidat yang dimana terdapat daftar kandidat, serta administrator bisa menambahkan kandidat, mengubah data kandidat yang ada serta menghapus kandidat yang sudah ada.



Gambar 16. Statistik Pemilihan

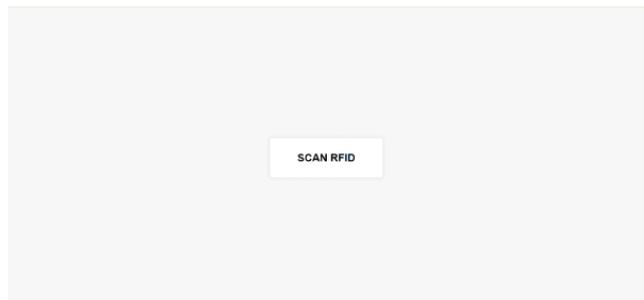
Gambar 16 adalah tampilan dari statistik suatu pemilih, yang dimana di dalamnya terdapat data dari pemilihan tersebut seperti jumlah suara, perolehan suara dari tiap kandidat dan data pemilihan lainnya serta bisa mencetak laporan statistik pemilihannya.

### 3.3.2 Tampilan Pemilih



Gambar 14. Form Tambah Kandidat

Gambar 14 menunjukkan *form* tambah pemilihan, yang dimana bisa menambahkan kandidat dengan yang di isi adalah pemilihan yang diikuti, nomor urut, nama kandidat, visi dan misi dan foto dari kandidat.



Gambar 17. Login Pemilih

Pada Gambar 17 adalah halaman *login* untuk pemilih yang dimana pada halaman ini pemilih diminta untuk *scan* kartu akses mereka pada alat RFID Reader.



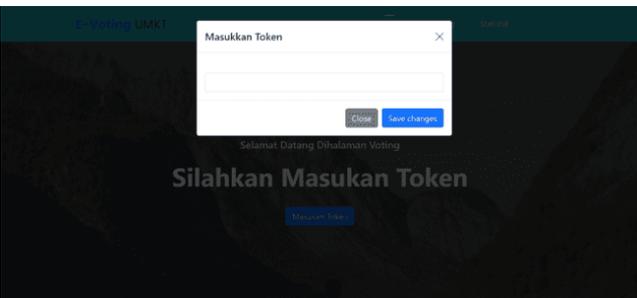
Gambar 18. Home

Gambar 18 adalah halaman *home* ketika pemilih mengunjungi *website*, pada halaman *home* ini terdapat 3 menu utama, yaitu *home*, *voting*, dan statistik.



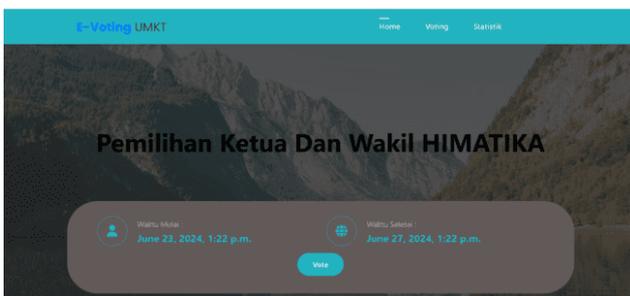
Gambar 19. Halaman Voting

Halaman *voting* yang terlihat pada gambar 19 dapat di akses ketika pemilih berhasil *login*, pada halaman ini terdapat *form* untuk masuk ke halaman pemilihan.



Gambar 20. Form Masukan Token

Gambar 20 adalah *form* untuk memasukkan token yang berguna sebagai akses masuk ke dalam halaman pemilihan, token tersebut akan diberikan dari administrator kepada pemilih yang terdaftar di halaman pemilihan seperti gambar 8.



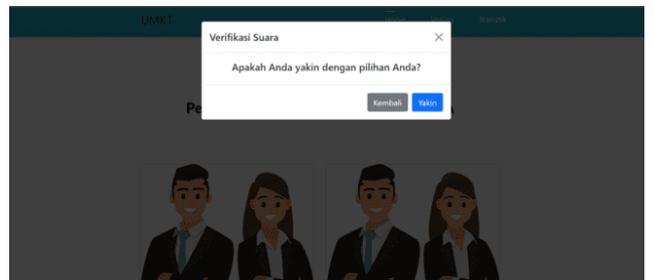
Gambar 21. Halaman Vote

Gambar 21 adalah halaman pemilihan, dimana halaman ini terdapat data pemilihan yang sedang berlangsung seperti judul pemilihan, waktu mulai dan waktu selesai.



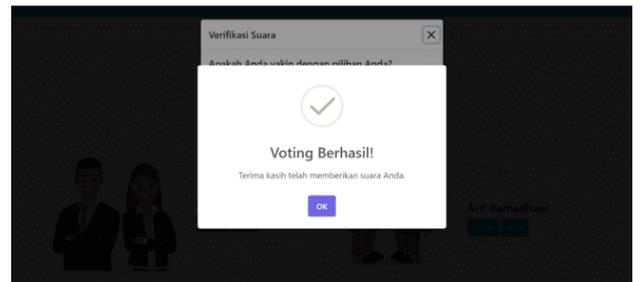
Gambar 22. Pemilihan Kandidat

Gambar 22 adalah halaman daftar kandidat yang berpartisipasi dalam pemilihan, pada halaman ini pemilih bisa melakukan *vote* dan detail kandidat.



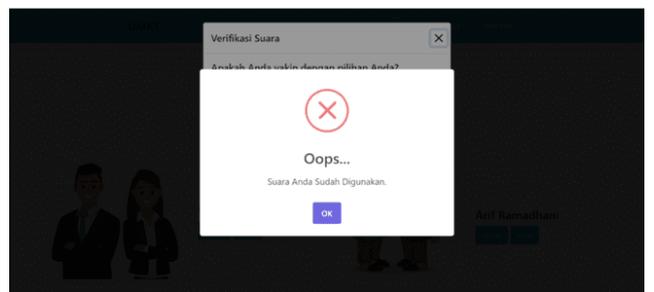
Gambar 23. Validasi Voting

Gambar 23 adalah *pop up* untuk verifikasi keyakinan terhadap pilihan dari pemilih.



Gambar 24. Voting Berhasil

Gambar 24 adalah pemberitahuan yang muncul ketika berhasil melakukan voting dan ketika pemilih menekan tombol OK maka akan otomatis kembali ke halaman *login*.



Gambar 25. Voting Gagal

Gambar 25 adalah pemberitahuan yang muncul ketika tidak berhasil melakukan voting serta alasan kenapa tidak berhasil dan ketika pemilih menekan tombol OK maka akan otomatis kembali ke halaman *login*.

### 3.4 Pengujian (Testing)

Setelah aplikasi *e-voting* berbasis *website* berhasil dibuat maka akan dilakukan pengujian untuk memastikan bahwa fitur-fitur yang ada dalam aplikasi berjalan dengan baik, pengujian pada aplikasi *e-voting* ini menggunakan metode pengujian *black box* dan pengujian model TAM (*Technology Acceptance Model*). Berikut ini adalah rincian dari pengujiannya.

#### 3.4.1 Pengujian Administrator

Pada pengujian Administrator ini yang di uji adalah mulai dari *login* untuk admin, pengelolaan data pemilih, kandidat, dan pemilihan, hasil pengujian dapat dilihat pada Tabel 1 berikut ini.

Tabel 1 . Pengujian Administrator

No.	Deskripsi Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
1	<i>Login</i>	Masukkan nama pengguna dan kata sandi yang benar.	Pengguna berhasil masuk.	Sukses
2	Pengelolaan Data Pemilih	Admin mencoba mengelola seperti Tambah, edit dan <i>delete</i>	Data Pemilih Berhasil Dikelola	Sukses
3	Pengelolaan Data Kandidat	Admin mencoba mengelola seperti Tambah, edit dan <i>delete</i>	Data Kandidat Berhasil Dikelola	Sukses
4	Pengelolaan Data Pemilihan	Admin mencoba mengelola seperti Tambah, edit dan <i>delete</i>	Data Pemilihan berhasil Dikelola	Sukses

#### 3.4.2 Pengujian Pemilih

Pada pengujian bagian halaman pemilih yang di uji seperti *login* pemilih sampai proses *voting*. Untuk Hasil dari pengujian pemilih dapat dilihat pada Tabel 2 di bawah ini.

Tabel 2. Pengujian Pemilih

No	Deskripsi Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
1	Pengajuan suara dengan kandidat dipilih	Masuk sebagai pengguna terdaftar dan berikan suara untuk kandidat.	Suara berhasil diajukan.	Sukses

No	Deskripsi Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
2	Pengajuan suara dua kali untuk kandidat yang sama	Coba berikan suara lebih dari sekali untuk kandidat yang sama.	Pesan kesalahan ditampilkan.	Sukses
3	Verifikasi suara setelah batas waktu pemungutan	Periksa hasil penghitungan setelah batas waktu pemungutan suara berakhir.	Hasil penghitungan hanya tersedia setelah pemungutan suara berakhir.	Sukses
4	Uji integritas data suara	Periksa bahwa suara yang telah diajukan tidak dapat diubah atau dihapus.	Data suara yang telah diajukan tidak dapat diubah atau dihapus setelah diajukan.	Sukses

#### 3.4.3 Pengujian TAM

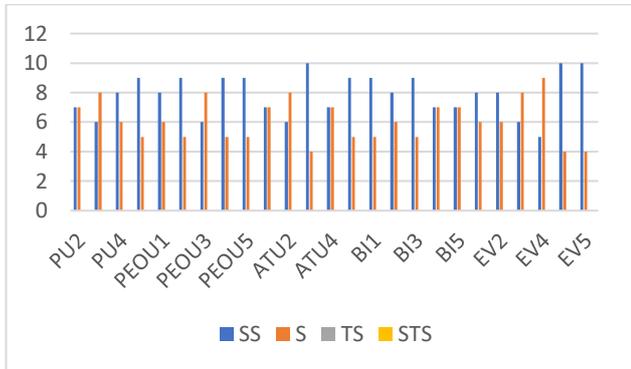
Pada pengujian ini bertujuan untuk mendapatkan respons yang berfokus pada 5 variabel penting, yaitu *Perceiver Usefulness* (PU), *Perceived Ease of Use* (PEOU), *Attitude Toward Using* (ATU), *Behavioral Intention to Use* (BI), dan *External Variables* (EV). Tabel 3 berikut ini adalah rincian pertanyaan dari tiap variabel.

Tabel 3. Pertanyaan Pengujian TAM

Variabel	Pertanyaan	Simbol
PU	Apakah Anda merasa setuju bahwa menggunakan sistem <i>e-voting</i> akan meningkatkan efisiensi pemilihan?	PU1
	Apakah Anda setuju bahwa sistem <i>e-voting</i> akan membuat proses pemilihan menjadi lebih cepat?	PU2
	Seberapa yakin Anda bahwa sistem <i>e-voting</i> akan meningkatkan akurasi hasil pemilihan?	PU3
	Menurut Anda, apakah Anda setuju sistem <i>e-voting</i> akan mempermudah proses pemilihan dibandingkan dengan metode tradisional?	PU4
	Apakah Anda merasa setuju bahwa sistem <i>e-voting</i> akan mengurangi kesalahan dalam penghitungan suara?	PU5
PEOU	Apakah Anda merasa setuju bahwa sistem <i>e-voting</i> mudah untuk dipelajari?	PEOU1
	Apakah Anda merasa setuju menggunakan sistem <i>e-voting</i> tanpa bantuan orang lain?	PEOU2
	Apakah Anda merasa setuju bahwa antarmuka sistem <i>e-voting</i> intuitif dan mudah dimengerti?	PEOU3
	Apakah Anda merasa setuju bahwa mengoperasikan sistem <i>e-voting</i> tidak memerlukan banyak usaha?	PEOU4
	Seberapa puas Anda menemukan informasi yang Anda butuhkan di dalam sistem <i>e-voting</i> ?	PEOU5
ATU	Bagaimana perasaan Anda tentang menggunakan sistem <i>e-voting</i> dalam pemilihan?	ATU1
	Apakah Anda merasa Setuju menggunakan sistem <i>e-voting</i> untuk memberikan suara?	ATU2

Variabel	Pertanyaan	Simbol
BI	Seberapa besar Anda merasa setuju menggunakan sistem <i>e-voting</i> ?	ATU3
	Apakah Anda merasa setuju dengan ide menggunakan sistem <i>e-voting</i> ?	ATU4
	Bagaimana kepuasan Anda secara keseluruhan terhadap sistem <i>e-voting</i> ?	ATU5
	Seberapa puas kemungkinan Anda akan menggunakan sistem <i>e-voting</i> di masa mendatang?	BI1
	Apakah Anda berniat untuk menggunakan sistem <i>e-voting</i> dalam pemilihan berikutnya?	BI2
EV	Seberapa puas keinginan Anda untuk merekomendasikan sistem <i>e-voting</i> kepada orang lain?	BI3
	Apakah Anda puas memilih sistem <i>e-voting</i> jika tersedia sebagai pilihan?	BI4
	Apakah Anda berencana untuk menggunakan sistem <i>e-voting</i> setiap kali ada pemilihan?	BI5
	Seberapa puas dukungan teknis yang Anda rasakan untuk menggunakan sistem <i>e-voting</i> ?	EV1
	Apakah Anda merasa puas mendapatkan pelatihan yang cukup untuk menggunakan sistem <i>e-voting</i> ?	EV2
	Seberapa puas infrastruktur teknologi yang tersedia untuk mendukung sistem <i>e-voting</i> ?	EV3
	Apakah Anda merasa puas bahwa lingkungan sosial (misalnya teman, keluarga, kolega) mendukung penggunaan sistem <i>e-voting</i> ?	EV4
	Apakah Anda puas bahwa data pribadi Anda aman saat menggunakan sistem <i>e-voting</i> ?	EV5

Dari pertanyaan-pertanyaan masing-masing variabel setelah dilakukan pengisian kuesioner dari 15 responden didapatkan hasil sebagai berikut.



Gambar 26. Grafik Hasil Kuesioner Pengujian TAM

Berdasarkan hasil grafik gambar 26 yang terdiri dari kuesioner 15 responden, dapat disimpulkan bahwa, sesuai dengan jawaban yang diterima dan menggunakan *Skala Likert*, responden memilih dari empat jawaban yang tersedia dengan bobot masing-masing: Sangat Setuju (ST) = 4, Setuju (S) = 3, Tidak Setuju (TS) = 2, dan Sangat Tidak Setuju (STS) = 1. Untuk Hasil Perhitungan Kuesioner TAM dapat dilihat pada Tabel 4 di bawah ini.

Tabel 4. Hasil Perhitungan Kuesioner TAM

No	Pertanyaan	Frekuensi Jawaban				Jumlah Skor				Total
		S	S	T	ST	S	S	T	S	
1	PU1	9	5	0	0	36	15	0	0	51
2	PU2	7	7	0	0	28	21	0	0	49
3	PU3	6	8	0	0	24	24	0	0	48
4	PU4	8	6	0	0	32	18	0	0	50
5	PU5	9	5	0	0	36	15	0	0	51
6	PEOU1	8	6	0	0	32	18	0	0	50
7	PEOU2	9	5	0	0	36	15	0	0	51
8	PEOU3	6	8	0	0	24	24	0	0	48
9	PEOU4	9	5	0	0	36	15	0	0	51
10	PEOU5	9	5	0	0	36	15	0	0	51
11	ATU1	7	7	0	0	28	21	0	0	49
12	ATU2	6	8	0	0	24	24	0	0	48
13	ATU3	10	4	0	0	40	12	0	0	52
14	ATU4	7	7	0	0	28	21	0	0	49
15	ATU5	9	5	0	0	36	15	0	0	51
16	BI1	9	5	0	0	36	15	0	0	51
17	BI2	8	6	0	0	32	18	0	0	50
18	BI3	9	5	0	0	36	15	0	0	51
19	BI4	7	7	0	0	28	21	0	0	49
20	BI5	7	7	0	0	28	21	0	0	49
21	EV1	8	6	0	0	32	18	0	0	50
22	EV2	8	6	0	0	32	18	0	0	50
23	EV3	6	8	0	0	24	24	0	0	48
24	EV4	5	9	0	0	20	27	0	0	47
25	EV5	10	4	0	0	40	12	0	0	52
Total Akhir Skor										1276
Total Skor Tertinggi (Skor Skala Tertinggi x Jumlah Responden x Jumlah Soal)										1400
Persentase Rata-rata (Total Akhir Skor / Total Skor Tertinggi x 100)										91%

Dari hasil persentase Tabel 4 Sistem Aplikasi *E-Voting* Berbasis Web memiliki nilai persentase rata-rata sebesar 91%, yang berarti bahwa sebagian besar responden setuju atau sangat setuju atas Sistem *Electronic Voting (E-Voting)* Berbasis Web.

### 3.5 Pemeliharaan (Maintenance)

Dalam pengembangan aplikasi *e-voting*, pemeliharaan mencakup semua aktivitas yang diperlukan untuk menjaga aplikasi tetap beroperasi dengan baik setelah tahap implementasi awal. Ini umumnya meliputi pembaruan perangkat lunak, perbaikan *bug*, penambahan fitur baru, dan penyesuaian terhadap perubahan lingkungan operasional atau kebutuhan pengguna.

#### 4. KESIMPULAN

Penelitian ini menyimpulkan bahwa pengembangan sistem aplikasi *e-voting* berbasis web dengan metode *waterfall* terbukti efektif. Proses pengembangan terdiri dari lima tahap: analisis kebutuhan melalui wawancara dan studi literatur, perancangan sistem dan antarmuka, implementasi menggunakan bahasa *Python* dengan *framework Django*, pengujian dengan metode *black box testing* dan TAM yang menghasilkan nilai 91%, dan pemeliharaan. Dengan demikian, metode *waterfall* telah terbukti efektif dalam menghasilkan solusi teknologi yang dapat mengatasi masalah pemungutan suara tradisional.

#### DAFTAR PUSTAKA

- [1] N. Chafid and H. Soffiana, "Impelementasi Algoritma Kriptografi Klasik Caesar Untuk Rancang Bangun Aplikasi E-Voting Berbasis Web (Studi Kasus: Sman 10 Tangerang)," *J. Ilm. Sains dan Teknol.*, vol. 6, no. 2, pp. 133–145, 2022, doi: 10.47080/saintek.v6i2.2249.
- [2] A. W. Krismanto, R. Adhitama, and N. A. Prasetyo, "Rancang Bangun Aplikasi E-Voting Pemilihan Ketua Pondok Pesantren Berbasis Web Menggunakan Metode Waterfall," *J. Inform. J. Pengemb. IT*, vol. 7, no. 2, pp. 104–108, 2022, doi: 10.30591/jpit.v7i2.3415.
- [3] S. Srimadona and S. Caniago, "Titik Temu Antara Voting Dan Syura Dalam Pemilihan Pemimpin Di Indonesia Perspektif Hukum Tata Negara Islam," *JISRAH J. Integr. Ilmu Syariah*, vol. 3, no. 2, p. 237, 2022, doi: 10.31958/jisrah.v3i2.6821.
- [4] K. Karmanis, "ELECTRONIC-VOTING (E-VOTING) DAN PEMILIHAN UMUM (Studi Komparasi di Indonesia, Brazil, India, Swiss dan Australia)," *Mimb. Adm. FISIP UNTAG Semarang*, vol. 18, no. 2, p. 11, 2021, doi: 10.56444/mia.v18i2.2526.
- [5] A. Kurniawan, "Perancangan Aplikasi E-Voting pada Pemilihan Ketua Osis Berbasis Mobile," *J. Ilm. Inform. dan Ilmu Komput.*, vol. 2, no. 1, pp. 26–31, 2023, doi: 10.58602/jima-ilkom.v2i1.15.
- [6] V. Ristiani, E. H. Hermaliani, and D. Y. Utami, "Sistem Informasi E-Voting Untuk Pemilihan Ketua OSIS Di SMK Strada II Jakarta," *Inf. Manag. Educ. Proffesionals*, vol. 4, no. 1, pp. 1–10, 2019, [Online]. Available: <https://ejournal-binainsani.ac.id/index.php/IMBI/article/view/1152>
- [7] M. Irvai, N. Efranda, and D. Mahdalena, "Perancangan Aplikasi Monitoring Calon Legislatif Berbasis Web Framework," *JUTIM (Jurnal Tek. Inform. Musirawas)*, vol. 8, no. 2, pp. 114–123, 2024, doi: 10.32767/jutim.v8i2.2157.
- [8] H. Willyandro, J. Setiawan, and A. Sulaiman, "Designing a Blockchain-Based Pemilu E-Voting Information System," *IJNMT (Internasional J. New Media Technol.)*, vol. 8, no. 1, pp. 42–49, 2021, doi: 10.31937/ijnmt.v8i1.1865.
- [9] R. S. Haq, K. Kaelan, and A. Armawi, "Implementasi Kebijakan E-Voting Dalam Pemilihan Kepala Desa (Pilkades) Di Kabupaten Boyolali Tahun 2019 Dan Implikasinya Terhadap Ketahanan Politik Wilayah," *J. Ketahanan Nas.*, vol. 26, no. 3, p. 399, 2020, doi: 10.22146/jkn.62262.
- [10] M. R. Anwar, R. Panjaitan, and R. Supriati, "Implementation Of Database Auditing By Synchronization DBMS," *Int. J. Cyber IT Serv. Manag.*, vol. 1, no. 2, pp. 197–205, 2021, doi: 10.34306/ijcitsm.v1i2.53.
- [11] D. Viduka, V. Kraguljac, and B. Licina, "A Comparative Analysis of the Benefits of Python and Java for Beginners," pp. 318–327, 2021.
- [12] T. S. Jalolov, "Advantage of Django Femworker," *Int. Multidiscip. J. Res. Dev.*, vol. 10, no. 12, p. 320, 2023, [Online]. Available: <https://www.ijmrd.in/index.php/imjrd>
- [13] S. Pargaonkar, "A Comprehensive Research Analysis of Software Development Life Cycle (SDLC) Agile & Waterfall Model Advantages, Disadvantages, and Application Suitability in Software Quality Engineering," *Int. J. Sci. Res. Publ.*, vol. 13, no. 8, pp. 120–124, 2023, doi: 10.29322/ijsrp.13.08.2023.p14015.
- [14] Mukrodin and Sugiamto, "Implementasi Metode Waterfall Dalam Membangun Tracer Study Dan Penerimaan Peserta Didik Baru," *J. Din. E-ISSN 2623-1786 / P-ISSN 0854-9524*, vol. 25, no. 1, pp. 39–50, 2020.
- [15] F. Aldi, "Web-Based New Student Admission Information System Using Waterfall Method," *Sinkron*, vol. 7, no. 1, pp. 111–119, 2022, doi: 10.33395/sinkron.v7i1.11242.
- [16] A. P. Putra, F. Andriyanto, K. Karisman, T. D. M. Harti, and W. P. Sari, "Pengujian Aplikasi Point of Sale Menggunakan Blackbox Testing," *J. Bina Komput.*, vol. 2, no. 1, pp. 74–78, 2020, doi: 10.33557/binakomputer.v2i1.757.