

Implementasi Website Pemilihan Raya Ketua BEM di STMIK Widya Utama Purwokerto

1st Nazar Yosa Wisnu Wardana
Program Studi Teknik Informatika
Universitas Telkom
Purwokerto, Indonesia

nazaryosa@student.telkomuniversity.ac.id

2nd Anggi Zafia, S.T., M.Eng
Program Studi Teknik
Informatika
Universitas Telkom
Purwokerto, Indonesia

anggiz@telkomuniversity.ac.id

3rd Trihastuti Yuniati, S.Kom., M.T
Program Studi Teknik Informatika
Universitas Telkom
Purwokerto, Indonesia

trihastutiy@telkomuniversity.ac.id

Abstrak — Pemilihan Ketua Badan Eksekutif Mahasiswa (BEM) di STMIK Widya Utama Purwokerto masih dilakukan secara manual, menyebabkan masalah seperti kesalahan pencatatan dan kurangnya transparansi. Untuk mengatasi hal ini, dikembangkanlah sistem informasi *e-voting* berbasis web menggunakan metode Waterfall. Sistem ini dirancang dengan React.js untuk antarmuka dan Supabase untuk basis data, memungkinkan mahasiswa untuk memberikan suara secara daring satu kali, mendokumentasikan proses dengan foto, dan melihat hasil secara *real-time*. Fitur pengelolaan data juga tersedia untuk admin. Hasil pengujian menunjukkan bahwa sistem ini berfungsi dengan baik dan layak menjadi solusi digital untuk pemilihan kampus yang lebih efisien dan transparan.

Kata kunci — *e-voting*, BEM, pemilihan mahasiswa, React.js, Supabase, sistem informasi, Waterfall

I. PENDAHULUAN

Pemilihan Ketua BEM merupakan bagian dari demokrasi di perguruan tinggi yang menuntut proses efektif dan transparan. Di STMIK Widya Utama Purwokerto, proses ini masih manual, yang sering menimbulkan masalah seperti kesalahan perhitungan, risiko kecurangan, dan kurangnya transparansi. Dengan jumlah mahasiswa yang mencapai lebih dari 1.900 orang, metode manual tidak lagi relevan. Kebutuhan akan sistem *e-voting* berbasis web menjadi mendesak untuk memverifikasi pemilih, menyederhanakan proses voting, dan menghitung suara secara otomatis. Sistem *e-voting* telah terbukti meningkatkan efisiensi dan transparansi di lingkungan pendidikan tinggi [1], [2].

Pengembangan sistem *e-voting* ini menggunakan metode Waterfall dengan tahapan yang sistematis [7]. Pengujian sistem dilakukan menggunakan metode *Black Box* untuk memastikan semua fitur berfungsi dengan baik sesuai fungsinya [8]. Sistem ini juga dapat mempercepat rekapitulasi, mencegah pemalsuan suara, dan meningkatkan kepercayaan pemilih terhadap hasil pemilihan [3], [4]. Sistem serupa telah dikembangkan untuk berbagai kasus lain, seperti pemilihan OSIS dan organisasi kampus [5], [6].

Dengan adanya sistem *e-voting* berbasis *web*, proses pemilihan diharapkan lebih adil, transparan, dan efisien. Sistem ini memudahkan mahasiswa memberikan suara tanpa harus hadir fisik, sehingga meningkatkan partisipasi. Selain itu, panitia terbantu dengan fitur validasi otomatis, manajemen akun, dan visualisasi hasil secara *real-time*. Penerapan sistem ini juga mendorong digitalisasi tata kelola organisasi mahasiswa dan mendukung budaya demokrasi yang sehat di kampus.

A. Website

Website merupakan sekumpulan halaman digital yang berisi informasi dalam bentuk teks, gambar, audio, dan video yang dapat diakses melalui internet [10]. Saat ini, *website* telah menjadi kebutuhan, termasuk bagi lembaga pendidikan untuk menyebarkan informasi, mendukung proses belajar, dan menyediakan layanan daring [10]. Selain itu, *website* juga bisa digunakan sebagai platform *e-voting* yang efisien, karena seluruh prosesnya terintegrasi di dalamnya [9].

B. Metode Waterfall

Metode Waterfall adalah model pengembangan perangkat lunak klasik yang memiliki alur sistematis dan terstruktur [11]. Dalam pendekatan ini, setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Tahapan tersebut meliputi analisis kebutuhan,

perancangan, pengembangan, pengujian, dan pemeliharaan. Setiap tahapan ini memastikan proses pengembangan berjalan secara berurutan dan terarah.

C. React JS

React JS adalah salah satu *library* JavaScript open-source yang dikembangkan oleh Facebook [12]. Meskipun bukan framework murni, React memiliki beberapa fitur unggulan. Fitur-fitur tersebut adalah JSX atau *extended* JavaScript yang memungkinkan modifikasi DOM menggunakan kode HTML, *single way data flow* yang memudahkan penanganan masalah, serta fitur *reusable* yang memungkinkan pembuatan komponen untuk digunakan kembali di bagian lain.

D. Supabase

Supabase adalah layanan *backend open-source* yang menyediakan layanan basis data relasional (SQL) dan penyimpanan berkas [13]. Layanan ini menyediakan tempat bagi pengembang untuk menyimpan dan mengelola berbagai jenis berkas, sehingga memudahkan dalam proses pengembangan aplikasi.

E. BlackBox Testing

Blackbox Testing adalah model pengujian yang berfokus pada fungsionalitas *software* tanpa melihat struktur internal kodennya [1]. Metode ini digunakan untuk menemukan kesalahan seperti kesalahan fungsi, tampilan, data yang digunakan, dan kesalahan deklarasi. Dalam penelitian ini, tingkat keberhasilan fungsional diukur dengan membandingkan jumlah kasus uji yang berhasil dengan total kasus uji.

F. User Acceptance Testing

User Acceptance Testing (UAT) adalah jenis pengujian yang dilakukan oleh pengguna akhir untuk memastikan sistem berjalan sesuai dengan kebutuhan dan tujuan mereka [16]. Pengujian ini menggunakan pendekatan *blackbox testing* untuk memverifikasi bahwa sistem benar-benar dapat digunakan dan memberikan manfaat. Pengukuran UAT dilakukan dengan menghitung persentase jawaban "Ya/Tidak" dan nilai rata-rata dari responden.

G. Unified Modeling Language

UML adalah alat bantu dalam pengembangan sistem berorientasi objek yang berfungsi sebagai standar untuk membuat blueprint *software* [14]. Pemodelan ini membantu menggambarkan sistem nyata dalam bentuk visual seperti *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram*, sehingga membuatnya lebih mudah dipahami dan diimplementasikan [15].

II. METODE

A. Studi Literatur

Studi literatur dilakukan dengan menelaah berbagai penelitian terdahulu terkait sistem informasi *e-voting*, pengembangan *web* modern menggunakan React.js dan Supabase, serta metode pengembangan perangkat lunak. Literatur ini digunakan sebagai dasar dalam merancang

solusi sistem informasi yang efektif dan relevan untuk konteks pemilihan raya mahasiswa.

B. Analisis Kebutuhan

Analisis kebutuhan dilakukan melalui observasi lapangan dan wawancara langsung dengan Ketua BEM serta mahasiswa aktif STMIK Widya Utama Purwokerto. Dari proses ini ditemukan bahwa sistem pemilihan Ketua BEM yang berjalan secara manual menimbulkan sejumlah kendala, seperti risiko kesalahan dalam pencatatan suara, keterlambatan rekapitulasi hasil, dan rendahnya transparansi proses pemungutan suara. Ketiadaan sistem verifikasi juga memungkinkan mahasiswa memilih lebih dari satu kali, serta menyulitkan proses dokumentasi yang akurat dan terdigitalisasi. Berdasarkan temuan tersebut, ditetapkan beberapa kebutuhan fungsional utama yang harus dipenuhi oleh sistem. Pertama, sistem harus dapat memverifikasi identitas mahasiswa berdasarkan NIM, dan membatasi hak pilih hanya satu kali. Kedua, sistem harus menyertakan proses dokumentasi berupa pengambilan foto sebagai bukti autentikasi saat *voting*. Ketiga, sistem perlu menyediakan fitur bagi admin untuk mengelola data kandidat dan pemilih secara fleksibel. Terakhir, sistem harus mampu menampilkan hasil perolehan suara secara *real-time* agar transparansi proses dapat terjamin.

C. Perancangan Sistem

Perancangan sistem melibatkan pembuatan desain antarmuka pengguna (UI), struktur basis data, serta diagram pemodelan sistem menggunakan pendekatan UML. Beberapa diagram yang digunakan meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*. Desain antarmuka dibuat responsif menggunakan Tailwind CSS, sedangkan *frontend* dikembangkan dengan React.js dan *backend* menggunakan layanan Supabase untuk autentikasi serta pengelolaan database berbasis PostgreSQL.

Sistem dirancang agar memisahkan hak akses antara pengguna (mahasiswa) dan *administrator* (admin). Mahasiswa dapat *login*, memilih calon Ketua BEM, dan melihat hasil *voting*. Admin dapat mengelola data kandidat dan pemilih, serta mengawasi proses voting secara keseluruhan.

D. Implementasi

Implementasi dilakukan berdasarkan hasil perancangan sistem sebelumnya. Proses pengembangan menggunakan text editor Visual Studio Code dan browser Chrome untuk pengujian tampilan antarmuka. Sistem dibangun dengan membagi komponen berdasarkan fungsionalitas, seperti halaman *login*, halaman *voting*, halaman *real-count*, dan dashboard admin. Autentikasi pengguna disediakan oleh layanan Supabase Auth, dan data disimpan pada tabel-tabel PostgreSQL dengan struktur relasional. Setiap aksi voting akan mencatat waktu, pilihan kandidat, dan bukti berupa foto snapshot dari pengguna.

E. Pengujian

Pengujian sistem bertujuan untuk memastikan bahwa seluruh fungsionalitas berjalan sesuai kebutuhan yang telah dirumuskan. Pengujian dilakukan dengan dua pendekatan

utama *BlackBox Testing* dan *User Acceptance Testing (UAT)*.

BlackBox Testing digunakan untuk mengevaluasi fungsionalitas sistem tanpa melihat struktur internal kode. Pengujian ini dilakukan oleh seorang pengembang web profesional dengan pengalaman lebih dari lima tahun dalam pengembangan sistem berbasis web. Fokus pengujian mencakup fitur login, pembatasan *voting* satu kali, dokumentasi pemilih melalui foto, manajemen data oleh admin, dan visualisasi hasil *voting* secara *real-time*.

Pengujian dilakukan dengan menjabarkan skenario uji untuk setiap fitur sistem dan mencatat apakah fitur tersebut bekerja sesuai harapan. Hasil dari pengujian kemudian dihitung menggunakan rumus berikut:

$$\% \text{Berhasil} = \frac{\text{Jumlah Kasus Valid}}{\text{Jumlah Total Kasus Uji}} \times 100\%$$

Hasil pengujian menunjukkan bahwa seluruh fitur utama sistem berhasil dijalankan tanpa error, dengan persentase keberhasilan mencapai 100%, menandakan bahwa sistem telah memenuhi kebutuhan fungsional secara utuh.

Selain itu, dilakukan juga *User Acceptance Testing (UAT)* dengan melibatkan pengguna akhir dari kalangan mahasiswa untuk menilai apakah sistem layak digunakan dari sudut pandang pengguna. UAT menggunakan dua bentuk penilaian:

1. Pertanyaan dengan jawaban Ya/Tidak, dihitung dengan rumus:

$$\text{Persentase} = \frac{\text{Jumlah Jawaban}}{\text{Total Responden}} \times 100\%$$

2. Pertanyaan menggunakan skala likert 1-5, dihitung rata-ratanya dengan rumus:

$$\text{Rata - rata} = \frac{\text{Total Skor}}{\text{Total Responden}}$$

III. HASIL DAN PEMBAHASAN

Sistem e-voting berbasis web untuk pemilihan raya Ketua BEM di STMIK Widya Utama telah berhasil diimplementasikan menggunakan pendekatan pengembangan perangkat lunak model Waterfall. Implementasi dilakukan dalam bentuk aplikasi berbasis web yang terdiri dari dua jenis pengguna, yaitu mahasiswa sebagai pemilih dan admin sebagai pengelola sistem.

Pada sisi mahasiswa, pengguna dapat melakukan login menggunakan NIM, melakukan proses *voting* satu kali, serta sistem secara otomatis akan menangkap gambar sebagai dokumentasi saat *voting* dilakukan. Proses ini bertujuan untuk meningkatkan keabsahan data dan mengurangi potensi kecurangan. Setelah proses *voting* selesai, mahasiswa dapat langsung melihat hasil perolehan suara dalam bentuk visualisasi diagram batang, yang menampilkan jumlah suara masing-masing kandidat secara *real-time*.

Pada sisi admin, pengguna memiliki akses untuk mengelola data pemilih, data kandidat, serta melakukan *reset* hasil *voting* jika dibutuhkan. Admin juga dapat

memantau aktivitas *voting* dan memverifikasi daftar pemilih sebelum proses dimulai.

A. Skenario 1 – Validasi *Voting* disisi *Frontend*

Pada skenario ini, mahasiswa *login* menggunakan akun yang telah terdaftar, kemudian melakukan *voting* melalui halaman pemilihan. Setelah *voting* berhasil dilakukan, tombol *voting* pada antarmuka disembunyikan agar tidak bisa digunakan lagi. Validasi yang diterapkan dalam pendekatan ini hanya berada di sisi tampilan, sehingga jika pengguna mencoba memanipulasi elemen HTML atau menggunakan *inspect element*, kemungkinan besar tombol *voting* dapat dimunculkan kembali.

Skenario ini dilakukan untuk menguji apakah sistem masih dapat dibobol jika hanya bergantung pada penguncian antarmuka. Mahasiswa yang telah *voting* mencoba kembali mengakses halaman *voting*, bahkan memaksa munculnya tombol *voting* kedua. Dengan validasi yang hanya bergantung pada tampilan, terbuka kemungkinan bahwa *request* kedua dapat dikirim ke server tanpa ada pengecekan logis yang sah.

B. Skenario 2 – Validasi *Voting* disisi *Backend*

Berbeda dengan skenario sebelumnya, pendekatan pada skenario ini adalah dengan menambahkan validasi langsung pada server. Ketika mahasiswa mencoba melakukan *voting*, sistem terlebih dahulu memeriksa apakah *user_id* yang bersangkutan sudah tercatat dalam tabel *vote_logs*. Jika sudah ada entri dengan ID tersebut, maka *request voting* akan langsung ditolak meskipun tombol *voting* tetap bisa dimunculkan secara paksa melalui manipulasi tampilan.

Skenario ini dilakukan dengan langkah yang sama seperti sebelumnya, yaitu *login*, *voting*, lalu mencoba *voting* ulang baik melalui UI maupun manipulasi jaringan. Dengan adanya pengecekan di sisi server, *voting* kedua benar-benar ditolak sebelum tersimpan ke dalam database. Hasil ini menjadi acuan apakah sistem telah sepenuhnya aman terhadap tindakan eksloitasi dari sisi pengguna.

C. Analisis

Hasil implementasi dan pengujian sistem *e-voting* menunjukkan bahwa sistem mampu mengatasi berbagai permasalahan yang sering terjadi dalam proses pemilihan manual, seperti duplikasi suara, keterlambatan rekapitulasi, serta minimnya transparansi. Dengan penggunaan autentikasi berbasis NIM dan fitur dokumentasi berupa pengambilan foto saat *voting*, sistem berhasil meningkatkan keandalan proses pemungutan suara. Fitur pembatasan *voting* satu kali juga bekerja dengan baik, mencegah adanya pemilih yang memberikan suara lebih dari sekali.

Dari sisi efisiensi, sistem memberikan pengalaman pengguna yang cepat dan praktis. Proses *login*, pemilihan kandidat, dan pengiriman suara dapat diselesaikan dalam waktu kurang dari satu menit. Hal ini menunjukkan bahwa

sistem layak digunakan bahkan dalam situasi pemilihan dengan jumlah pemilih besar. Visualisasi hasil suara secara langsung juga memberikan nilai tambah dalam hal keterbukaan hasil pemilihan kepada semua pihak yang berkepentingan.

Secara teknis, penggunaan *framework* modern seperti React.js dan layanan Supabase terbukti mampu menyediakan performa yang ringan, stabil, dan mudah dikembangkan lebih lanjut. Fitur real-time yang dihadirkan melalui Supabase *Realtime* juga meningkatkan kepercayaan terhadap akurasi data suara karena hasil dapat dilihat tanpa jeda waktu.

Dari hasil UAT, pengguna menyatakan bahwa antarmuka sistem mudah dipahami dan digunakan. Ini menunjukkan bahwa sistem telah berhasil memenuhi prinsip dasar *user experience* (UX), yaitu intuitif, efisien, dan tidak membingungkan. Namun, dari sisi keterbatasan, sistem masih belum terhubung langsung dengan sistem informasi akademik kampus, sehingga proses validasi data pemilih masih dilakukan secara manual oleh admin. Selain itu, fitur dokumentasi melalui kamera sangat bergantung pada izin akses perangkat pengguna, yang berpotensi menimbulkan kendala teknis pada perangkat tertentu.

Secara keseluruhan, sistem *e-voting* ini telah memenuhi tujuan awal pembangunan, yaitu menyediakan sarana pemilihan digital yang aman, efisien, dan transparan. Sistem ini juga memiliki potensi besar untuk diimplementasikan dalam skala yang lebih luas, misalnya dalam pemilihan organisasi mahasiswa tingkat fakultas maupun universitas, dengan catatan perlu dilakukan integrasi lebih lanjut dan peningkatan pada sisi keamanan data.

IV KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem *e-voting* berbasis web sebagai solusi terhadap permasalahan pemilihan Ketua BEM di STMIK Widya Utama Purwokerto yang sebelumnya dilakukan secara manual. Sistem yang dibangun memungkinkan proses pemungutan suara dilakukan secara daring, dengan fitur-fitur seperti autentikasi berbasis NIM, pembatasan satu kali *voting*, dokumentasi visual saat pemilihan, serta visualisasi hasil suara secara *real-time*. Penggunaan metode pengembangan Waterfall memberikan struktur yang sistematis dan mendukung proses pembangunan sistem mulai dari analisis kebutuhan hingga pengujian.

Hasil pengujian fungsional melalui BlackBox Testing yang dilakukan oleh *expert* menunjukkan bahwa seluruh fitur utama sistem berjalan dengan baik dan memenuhi kebutuhan fungsional secara utuh. Sementara itu, hasil UAT dari pengguna akhir menunjukkan tingkat penerimaan yang tinggi terhadap sistem, baik dari segi antarmuka maupun kemudahan penggunaan. Hal ini

menandakan bahwa sistem tidak hanya layak dari sisi teknis, tetapi juga secara praktis dapat diterima oleh pengguna kampus.

Secara keseluruhan, sistem informasi pemilihan raya ini dinilai berhasil mencapai tujuan awal penelitian, yaitu meningkatkan efisiensi, akurasi, dan transparansi dalam proses pemilihan Ketua BEM. Sistem ini juga memberikan kontribusi nyata dalam mendukung digitalisasi tata kelola organisasi mahasiswa dan dapat dijadikan model awal untuk pengembangan sistem pemilu kampus yang lebih modern dan adaptif.

REFERENSI

- [1] L. Penelitian, P. Pembelajaran, P. Kepada Masyarakat, E. Prasetya Nuryanto, P. Budi Wintoro, and B. Lampung, “Semnas Iib Darmajaya Rancang Bangun Sistem *E-voting* Pada Pemilihan Raya (Pemira) Iib Darmajaya Berbasis Web,” 2017.
- [2] L. P. Gelu, D. Nababan, and Y. P.K Kelen, “*E-voting* Pemilihan Ketua Bem Universitas Timor Berbasis Mobile Menggunakan Metode Waterfall,” *Journal of Information and Technology*, vol. 2, no. 2, pp. 58–64, Sep. 2022, doi: 10.32938/jitu.v2i2.3192.
- [3] V. Primadyanti, J. Waleng, G. Putu, K. Juliharta,) Ketut, and Q. Fredlina, “Implementasi *E-voting* Pemilu Raya Mahasiswa Universitas Primakara Berbasis Web (Studi Kasus Pada Universitas Primakara).”
- [4] M. Magfur and N. Anwar, “Perancangan Aplikasi Mobile *E-voting* Dengan Framework Flutter Untuk Pemilihan Ketua BEM Universitas Esa Unggul Kampus Tangerang.”
- [5] A. Nabilah and Y. Amrozi, “Jurnal Teknologi Sistem Informasi dan Aplikasi Rancang Bangun *E-voting* Berbasis Web pada Organisasi Karang Taruna Kelurahan Kedurus,” vol. 2, no. 3, pp. 2654–4229, 2019, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/JTSI>
- [6] M. N. Syamsudin, Y. T. Wiranti, M. Ihsan, and A. Putera, “Pembangunan Sistem Informasi *E-voting* Pada Universitas XYZ.”
- [7] E. Prasetyo and A. Putra, “Implementasi Waterfall Model Dalam Pengembangan Sistem Informasi Eksekutif Penduduk,” *Journal of Information Systems and*

- Informatics*, vol. 3, no. 1, 2021, [Online]. Available: <http://journal-isi.org/index.php/isi>
- [8] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and D. Yulianti, “Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis,” vol. 4, no. 4, pp. 2622–4615, 2019, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/informatika>
- [9] A. Pardede, A. Sidik, Md. Diterbitkan Pertama Kali Oleh, U. Islam Kalimantan Muhammad Arsyad Al Banjari Jl Adhyaksa No, K. Banjarmasin, and K. Selatan Indonesia, *Desain Cover dan Layout*. 2019.
- [10] I. Tangkudung, R. Deddy, R. Dako, and A. Y. Dako, *Evaluasi Website Menggunakan Metode Iso/Iec 25010*. 2019. [Online]. Available: <http://www.ung.ac.id>
- [11] S. Supiyandi, M. Zen, C. Rizal, and M. Eka, “Perancangan Sistem Informasi Desa Tomuan Holbung Menggunakan Metode Waterfall,” *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 2, p. 274, Apr. 2022, doi: 10.30865/jurikom.v9i2.3986.
- [12] S. Mufti Prasetyo, M. Ivan Prayogi Nugroho, R. Lima Putri, and O. Fauzi, “BULLET : Jurnal Multidisiplin Ilmu Pembahasan Mengenai Front-End Web Developer dalam Ruang Lingkup Web Development”, [Online]. Available: <https://journal.mediapublikasi.id/index.php/bullet>
- [13] H. Ulum Fajar, A. Putra Kharisma, and A. Bhawiyuga, “Pengembangan Aplikasi Layanan Kesehatan berbasis Web untuk Skrining Pendengaran menggunakan Arsitektur Clean (Studi Kasus: Fakultas Kedokteran Universitas Brawijaya),” 2022. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [14] V. Yoga and P. Ardhana, “Perancangan Sistem Informasi Apotek Qamarul Huda Menggunakan Unified Modeling Language (UML),” 2021.
- [15] P. Aplikasi Hauling Trip Di Industri Tambang Batubara Ronal, “Desain Unified Modeling Language (UML) Dalam,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 9, no. 4, 2022, [Online]. Available: <http://jurnal.mdp.ac.id>
- [16] “View of Analisis Blackbox Testing Dan User Acceptance Testing Terhadap Sistem Informasi SolusimedsoSKU,” Kodepena.org, 2025. <http://jtk.kodepena.org/index.php/jtk/article/view/54/36> (accessed Jul. 06, 2025).