SKRIPSI

PERANCANGAN SISTEM PEMILIHAN GUBERNUR MAHASISWA DENGAN METODE *ELECTRONIC VOTING* BERBASIS WEBSITE

(Studi Kasus : Badan Eksekutif Mahasiswa Fakultas Teknik Universitas Muhammadiyah Magelang)

R. BIMA GOFIRULI MAULIDA

16.0504.0039

PROGRAM STUDI TEKNIK INFORMATIKA S1

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH MAGELANG

2021

BAB 1
PENDAHULUAN

1.1 Latar Belakang Permasalahan

Pemungutan suara (voting) adalah salah satu tahap pemilihan umum yang dilaksanakan pada setiap negara, hal ini termasuk pemungutan suara yang dilaksanakan di Universitas Muhammadiyah Magelang yang memiliki bermacam-macam jenis. Pemungutan suara pada pemilihan calon ketua Himpunan Mahasiswa Jurusan (HMJ), Badan Eksekutif Mahasiswa (BEM) dan Pemilihan Presiden Mahasiswa (Pemira), yang saat ini berjalan masih menggunakan sistem manual, seperti dengan menggunakan kertas, pencoblosan dan pencelupan jari ke tinta.

Pemrosesan data pemilih dan data hasil pemungutan suara saat ini masih menggunakan kertas sehingga memiliki beberapa kekurangan yaitu dalam hal penyimpanan dan pencarian data. Pada saat dilakukannya proses audit oleh kepengurusan yang lama, masih menggunakan cara yang manual sehingga mengakibatkan banyak data rekap yang hilang dan membutuhkan waktu yang cukup lama dalam proses audit (Lavarino, 2016). Rekap hasil pemungutan suara pada tahun sebelumnya belum terdokumentasi dengan baik.

Metode voting saat ini terdiri dari dua metode yaitu pemungutan suara offline (metode berbasis kertas, metode pemungutan suara mesin, pemungutan suara melalui pos) dan pemungutan suara online (*e-voting*). Pemungutan suara secara offline dinilai masih memiliki kelemahan dari aspek efisiensi waktu dan kurang efektif. *E-voting* merupakan sistem yang memanfaatkan perangkat elektronik dan mengolah informasi digital untuk membuat surat suara, memberikan suara, menghitung perolehan suara, menayangkan perolehan suara, dan penyimpanan berkas untuk jejak audit (Ridwan et al., 2016).

Pada penelitian sebelumnya yaitu Implementasi RESTful Web Service untuk Sistem Perhitungan Suara Secara Cepat pada Pilkada (Rofiq, 2017). Penelitian tersebut menjelaskan tentang pemungutan suara menggunakan metode Real-Quick Count dengan pemanfaatan web service didalamnya. Dalam penelitiannya terdapat beberapa kekurangan dalam validasi data pemungutan suara,

validasi yang belum tercantum yaitu validasi data serta validasi waktu. Hal tersebut sangat beresiko terhadap hasil pemilihan karena dapat terjadi manipulasi data.

Hasil dari pemungutan sebelumnya adalah proses perhitungan suara memakan waktu yang lama dan terdapat kesalahan data karena terjadinya kertas suara yang tidak sah. Berdasarkan penelitian yang telah dilakukan, maka akan dibuat aplikasi pemungutan suara dengan metode *e-voting* untuk pemilihan gubernur BEM Fakultas Teknik. Metode *e-voting* ini merupakan suatu sistem yang memanfaatkan perangkat elektronik dan mengolah informasi digital untuk membuat surat suara, memberikan suara, menghitung perolehan suara, menayangkan perolehan suara dan memelihara serta menghasilkan jejak audit. Dengan penerapan sistem *e-voting* ini akan membantu proses pemungutan suara di BEM Fakultas Teknik dengan penerapan bahasa pemrograman PHP dan sistem basis data MySQL.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan di atas, maka rumusan masalah yang dipilih yaitu bagaimana membangun sistem pemungutan suara pemilihan calon Gubernur BEM Fakultas Teknik menggunakan metode *e-voting* berbasis website.

1.3 Tujuan Penelitian

Adapun tujuan penulisan ini adalah membangun sistem pemungutan suara pemilihan calon Gubernur BEM Fakultas Teknik sebagai alat bantu pemungutan suara yang lebih efisien dan akurat dalam data yang dihasilkan dalam pemilihan menggunakan metode *e-voting* berbasis website.

1.4 Manfaat Penelitian

Dari penelitian ini, didapatkan beberapa manfaat antara lain :

1. Mempercepat dalam pemungutan dan menayangkan perhitungan hasil pilihan suara secara *real time* calon Gubernur BEM Fakultas Teknik.

2. Mahasiswa dapat melakukan pemilihan suara tanpa harus datang ke kampus dan dapat diakses secara online.
3. Mempermudah mahasiswa untuk mengetahui calon Gubernur BEM Fakultas Teknik.

BAB 2
TINJAUAN PUSTAKA

2.1 Penelitian yang relevan

1. Penelitian yang dilakukan (Adi Hermawansyah, 2019) dengan judul "Perancangan Sistem Informasi E-Voting Pemilihan Ketua RT Dengan Verifikasi Nomor Induk Keluarga Berbasis Web" menyatakan bahwa penelitian tentang pemilihan Ketua RT saat ini masih masih menggunakan sistem *voting* konvensional, serta pemungutan suara dan perhitungan masih manual. Pembuatan sistem ini bertujuan untuk memudahkan masyarakat dalam memungut suara dalam pemilihan Ketua RT, terutama banyak warga sekitar yang sibuk dengan pekerjaannya sehingga tidak ada waktu untuk ke pemilihan ketua RT. Dalam kasus tersebut, penelitian ini menggunakan verifikasi Nomor Induk Keluarga dalam membatasi hak suara agar pemilihan dilakukan sekali oleh satu nomor induk. Penelitian tersebut juga belum menyertakan konfirmasi pemungutan suara. Dalam hal itu membuktikan bahwa pemungutan suara dapat dilakukan secara online dengan pembuatan sistem baru yaitu *e-voting* dengan menggunakan verifikasi *account gmail* dari setiap mahasiswa yang terdaftar dalam peserta pemilihan calon Gubernur Mahasiswa, verifikasi tersebut juga berfungsi untuk pembatasan hak pilih dari setiap mahasiswa agar pemilihan hanya bisa dilakukan satu kali. Dan untuk konfirmasi pemilihan nanti akan mendapatkan *message* ke *e-mail* dari masing-masing pemilih.
2. Penelitian yang dilakukan (Rotikan et al., 2020) dengan judul "Penerapan PWA Untuk Aplikasi E-Voting Pemilihan Pengurus BEM dan Senat di Universitas Klabat" menyatakan bahwa pemungutan suara dalam pemilihan pengurus baru BEM maupun Senat masih bersifat konvensional. Kekurangan yang biasa terjadi yaitu penggunaan kertas suara dan perhitungan suara yang memakan waktu lama. Dalam kasus tersebut, maka dibuatlah sebuah sistem *e-voting* berbasis web dengan menggunakan *Progressive Web Application* (PWA) yang berfungsi

untuk mempermudah pengguna merasakan aplikasi *mobile* melalui *browser*. PWA tersebut berfungsi untuk menampilkan aplikasi mobile di browser kita sendiri sehingga tampilannya akan tetap sama, hanya mengikuti dari ukuran browser yang dipakai. Hal ini jelas mempermudah mahasiswa dalam melakukan pemungutan suara secara online dan tanpa mengunduh aplikasi di layanan aplikasi seperti *play store* maupun *apps store*. Hal ini membuktikan pentingnya pembuatan sistem *e-voting* dalam pemungutan suara agar lebih efisien dan akurat serta penghematan sumber daya seperti *paperless*.

3. Penelitian ini dilakukan (Wisnu, Dyah A. M. G., Aswin Suharsono, S.T., M.T., M.Kom., Denny S. R., S.Kom., 2012) dengan judul "Rancang Bangun Sistem E-Voting Dengan Menerapkan Hash dan Digital Signature Untuk Verifikasi Data Hasil Voting" menyatakan bahwa banyak kesalahan yang terjadi antara lain kesalahan dalam memberi tanda pada kertas suara sehingga suara menjadi tidak sah, pengumpulan kartu suara yang berjalan lambat karena perbedaan kecepatan pelaksanaan dari tiap daerah, aspek keamanan data yang belum ada. Dengan begitu, dapat dibuat sistem pemungutan suara menggunakan metode *e-voting* guna untuk meminimalisir beberapa kekurangan diatas dengan penghitungan suara yang lebih cepat, keamanan data yang mengimplementasikan token untuk setiap pemilih sehingga kerahasiaan hasil pilihan suara tetap terjaga serta penambahan fitur untuk penayangan hasil pilihan suara.
4. Penelitian yang dilakukan (Zulkiawan et al., 2020) dengan judul "Penerapan Metode *User Authentication* Pada Sistem Monitoring, *E-Voting*, dan Evaluasi PEMIRA" menyatakan bahwa sistem PEMIRA tersebut tidak melibatkan panitia untuk monitoring dan evaluasi hasil pemungutan suara sehingga dinilai tidak transparan. Sistem tersebut juga dapat diakses secara bebas dengan menggunakan NIM dan *password* namun hal ini masih dianggap rentan karena tidak adanya validasi bahwa pemilih yang melakukan vote adalah pemilik akun yang sebenarnya. Dengan begitu maka dapat dibuat sistem *e-voting* guna

untuk mengatasi masalah diatas dengan verifikasi *e-mail* untuk validasi *user account* agar pemilih yang *login* ke dalam sistem memang benar sebagai pemilik *account* yang sebenarnya.

5. Penelitian yang dilakukan (Nuryanto & Wintoro, 2017) dengan judul "Rancang Bangun Sistem E-Voting Pada Pemilihan Raya (PEMIRA) IIB Darmajaya Berbasis Web" menyatakan bahwa pemilihan suara masih bersifat konvensional sehingga menyebabkan beberapa kekurangan dalam pelaksanaannya seperti perhitungan yang lama, meminimalisir dana dan percetakan kertas suara. Dalam pengembangan sistem tersebut masih memiliki kekurangan dalam verifikasi *user* yang belum jelas, keamanan data yang belum dijelaskan secara menyeluruh. Dengan begitu dapat dibuat sistem menggunakan metode *e-voting* dengan penggunaan token ke semua *account* pemilih guna untuk keamanan data agar vote tidak diketahui oleh siapapun, verifikasi *e-mail* untuk memastikan pemilih yang melakukan vote adalah pemilik *account* yang sebenarnya.

Dari kelima penelitan diatas terdapat beberapa penelitan yang menerapkan voting secara konvensional sehingga menyebabkan beberapa kendala diantaranya penggunaan kertas dan penghitungan suara yang cukup lama. Selain itu ada beberapa penelitian yang telah terbukti bahwa voting dapat diimplementasikan secara online yaitu dengan menggunakan metode *e-voting.*

2.2 Landasan Teori

Sistem *e-voting* yang akan dibuat penulis dalam penelitian ini berdasarkan hasil analisa dari beberapa penelitian relevan yang dibahas diatas. Pembuatan sistem ini mengimplementasikan metode *e-voting* dengan memanfaatkan perangkat elektronik dan mengolah informasi digital untuk membuat surat suara, memberikan suara, menghitung perolehan suara, menayangkan perolehan suara dan memelihara serta menghasilkan jejak audit.

Pembuatan sistem *e-voting* ini dapat menggunakan bahasa pemrograman PHP dan media penyimpanan database MySQL. Berdasarkan analisa dari penelitian relevan diatas dan penjabaran variable-variabel yang berkaitan, dengan penelitian

ini dapat diambil kesimpulan bahwa pembuatan sistem ini dapat mengimplementasikan metode *e-voting*.

2.2.1 *E-Voting*

Menurut (Azwanti, 2017) dalam jurnalnya "Perancangan E-Voting Berbasis Web" menjelaskan bahwa sistem *e-voting* (pemungutan suara elektronik) adalah proses pemungutan suara yang memungkinkan pemilih untuk memberikan suara secara aman dan rahasia melalui internet. *E-voting* sendiri merupakan sebuah sistem yang memanfaatkan perangkat elektronik dan mengolah informasi digital untuk membuat surat suara, memberikan suara, menghitung perolehan suara, menayangkan perolehan suara dan memelihara serta menghasilkan jejak audit.

2.2.2 *Flowchart*

Menurut (Lavarino, 2016) flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Tujuan pembuatan flowchart yaitu :

a. Menggambarkan suatu tahapan penyelesaian masalah
b. Secara sederhana, terurai, rapi, dan jelas
c. Menggunakan simbol-simbol standar

SIMBOL-SIMBOL DIAGRAM ALIR

Simbol	Maksud	Simbol	Maksud
	Terminal (START, END)		Titik sambungan pada halaman yang sama
	Input/Output (READ, WRITE)		Titik konektor yang berada pada halaman lain
	Proses (menyatakan assignment statement)		Call (Memanggil subprogram)
	Decision (YES, NO)		Dokumen
	Display		Stored Data
	Alur proses		Preparation (Pemberian nilai awal suatu variabel)

Gambar 2. 1 Simbol *Flowchart*

2.2.3 PHP

Dalam jurnal (Lavarino, 2016) *Hypertext Preprocessor* (PHP) adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. PHP yang merupakan *server-side*

scripting maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya dikirim ke browser dalam format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman web lebih terjamin. Dalam penelitian ini menggunakan PHP karena memiliki beberapa kelebihan antara lain :

a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunanya.
b. PHP memiliki tingkat akses yang lebih cepat.
c. PHP memiliki tingkat *lifecycle* yang cepat sehingga selalu mengikuti perkembangan teknologi internet.
d. PHP juga mendukung akses ke beberapa database yang sudah ada baik yang bersifat *free*/gratis ataupun komersial. Database itu antara lain : MySQL, PostgresSQL, infomix, dan MicrosoftSQL Server. Web server yang mendukung PHP dapat ditemukan dimana-mana dari mulai Apache, IIS, AOserver, phttp, Fhttp, PWS, Lighttpd hingga Xitami dengan konfigurasi yang relative mudah.

2.2.4 MySQL

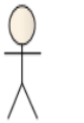
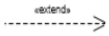
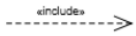
Menurut (Lavarino, 2016) MySQL (*My Structure Query Language*) adalah salah satu Basis Data Management System (DBMS) dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL, dan lainnya. MySQL berfungsi untuk mengolah Basis Data menggunakan Bahasa SQL. MySQL bersifat *open source* sehingga kita bisa menggunakannya secara gratis. Pemrograman PHP juga sangat mendukung atau mensupport dengan Basis Data MySQL.

2.2.5 UML (*Unified Modelling Language*)

Menurut (Azwanti, 2017) UML (*Unified Modelling Language*) adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi, dan mendokumentasikan *artifacts* dari sistem *software*. Adapun beberapa jenis diagram UML yang biasa digunakan untuk perancangan, diantaranya :

a. *Use Case Diagram*

Use Case adalah suatu pola atau gambaran yang menunjukkan kelakuan atau kebiasaan sistem, serta mempresentasikan sebuah interaksi antara aktor dengan sistem.

	*ACTOR* Orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal frase nama *actor*.
	*USE CASE* Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesar antar unit atau actor biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case.
	ASOSIASI/*ASSOCIATION* Komunikasi antara *actor* dan use case yang berpartisipasi pada use case atau use case memiliki interraksi dengan *actor*.
	EKSTENSI/*EXTEND* Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan memiliki nama depan yang sama dengan use case yang di tambahkan.
	GENERALISASI/*GENERALIZATION* Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
	MENGGUNAKAN/*INCLUDE* Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsional atau sebagai syarat dijalankan use case ini.

Gambar 2. 2 *Use Case Diagram*

b. Class Diagram

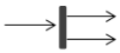
Class menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

GAMBAR	NAMA	KETERANGAN
	Generalization	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
	*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	Realization	Operasi yang benar-benar dilakukan oleh suatu objek.
	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya

Gambar 2. 3 *Class Diagram*

c. *Activity Diagram*

Activity menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

	STATUS AWAL/*INITIAL* Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah satutus awal.
	AKTIVITAS/ *ACTIVITY* Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	*PERCABANGAN / DECISION* Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	PENGGABUNGAN/ *JOIN* Asosiasi penggabungan dimana lebih dari satu aktivitas lebih dari satu.
	STATUS AKHIR/ *FINAL* Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status satu.
	*SWIMLINE* Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Gambar 2. 4 *Activity Diagram*

d. *Sequence Diagram*

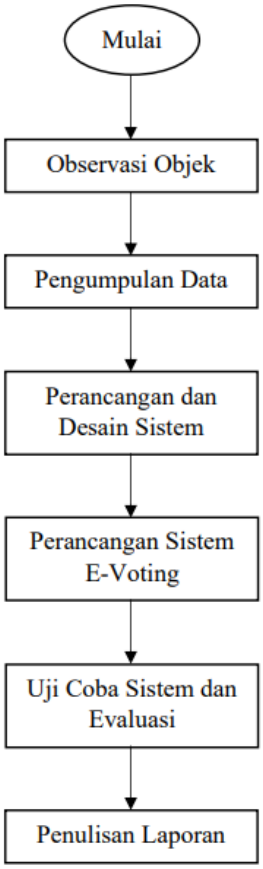
Sequence menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu.

GAMBAR	NAMA	KETERANGAN
	*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
	*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
	*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Gambar 2. 5 *Sequence Diagram*

BAB 3
METODOLOGI PENELITIAN

3.1 Prosedur Penelitian



Gambar 3. 1 Prosedur Penelitian

3.1.1 Observasi Objek

Kegiatan pengamatan terhadap objek yang akan diteliti baik secara langsung maupun tidak langsung dengan tujuan mendapatkan data yang diperlukan selama penelitian. Penulis melakukan observasi secara langsung ke lokasi objek penelitian. Observasi langsung dilakukan pada BEM Fakultas Teknik Universitas Muhammadiyah Magelang dan memakan waktu sekitar dua hari untuk mengetahui SOP pemilihan calon Gubernur.

Pada observasi ini, peneliti melakukan pencatatan seluruh kegiatan voting sebelumnya yang masih menerapkan voting secara konvensional. Pencatatan dilakukan secara terurut dari awal pencalonan

Gubernur baru sampai tahap penghitungan suara setelah acara pemungutan suara.

3.1.2 Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan oleh sistem. Pengumpulan data dilakukan sebelum pembuatan sistem dengan beberapa metode yaitu metode wawancara, observasi dan studi pustaka.

1. Wawancara dan Observasi

   Metode ini dilakukan dengan tanya jawab langsung dengan Pengurus Badan Eksekutif Mahasiswa Fakultas Teknik. Dalam wawancara tersebut melibatkan ketua dan wakil DPM Fakultas Teknik selaku panita pengurus pemilihan dan dikumpulkan beberapa informasi yang berkaitan dengan pembuatan sistem *e-voting.* Sistem yang dahulu digunakan dalam BEM yaitu secara konvensional/manual dengan pengadaan kertas suara, perhitungan suara yang masih manual sehingga membutuhkan waktu yang lama dan pengarsipan yang masih bersifat *hardfile*.

2. Kuisioner

   Metode ini dilakukan dengan menyediakan kuisioner *google form* untuk ditujukan langsung ke mahasiswa guna untuk mendata mahasiswa Fakultas Teknik sejumlah 10 orang. Data tersebut yang nantinya akan digunakan untuk implementasi perancangan *e-voting*.

3. Studi Pustaka

   Metode ini dilakukan dengan mencari teori-teori yang mendasari masalah dan bidang yang akan diteliti. Dengan melakukan pencarian metode *e-voting* di internet seperti *google scholar*. Selain itu informasi juga didapat dari penelitian-penelitan sejenis yang berkaitan dengan penelitian yang dilakukan. Pustaka yang dijadikan acuan untuk menyusun penelitian ini dapat dilihat di Daftar Pustaka.

3.1.3 Perancangan dan Desain Sistem

Tahapan selanjutnya setelah mendapatkan data, maka dilakukan perancangan sistem dengan aplikasi yang mendukung perancangan sistem menggunakan metode UML (*Unified Modeling Language*) untuk memvisualisasi, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, *artifacts* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem yang akan dibuat. Adapun tahapan yang dilakukan dalam pemodelan UML meliputi :

a. *use case diagram*, suatu sistem memvisualisasi sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Aktor pada sistem *e-voting* ini adalah admin dan pengguna (mahasiswa). Interaksi yang terjadi yaitu admin memiliki wewenang dalam mengelola sistem tersebut antara lain menginputkan pasangan calon Gubernur yang baru, mengelola data yang masuk dan menghitung hasil pemilihan serta menampilkan secara *realtime*. Sedangkan pengguna (mahasiswa) memiliki wewenang untuk memilih pasangan calon, melihat visi misi pasangan calon dan melihat hasil pilihan.

b. *class diagram*, berisikan gambaran struktur sistem tertentu dengan memodelkan kelas, atribut, operasi serta hubungan antar objek.

c. *sequence diagram*, menggambarkan kolaborasi dinamis antara sejumlah objek yang berguna untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek.

d. *activity diagram*, berisikan alir aktifitas dalam program yang sedang dirancang, bagaimana proses alir berawal, keputusan yang mungkin terjadi dan bagaimana sistem berakhir.

3.1.4 Perancangan Sistem E-Voting

Pada tahap ini merupakan perancangan sistem menggunakan metode *e-voting* dengan tahapan analisis, desain, coding dan testing sistem.

3.1.5 Uji Coba Sistem dan Evaluasi

Setelah sistem selesai dibuat akan dilakukan pengujian kelayakan terhadap aplikasi tersebut. Pengujian dilakukan secara bertahap dan memungkinkan memakan waktu yang cukup lama dikarenakan beberapa kemungkinan seperti *error* koding dan lain-lain.

Pengujian dilakukan dengan pengecekan kebergunaan tombol-tombol pada *interface* sistem dan mengecek tujuan penelitian yang akan terpenuhi di sistem tersebut. Pengujian menggunakan *black box testing* yang bertujuan untuk mengetahui apakah fungsi aplikasi telah sesuai dengan ketentuan yang dibuat atau belum. Fokus utamanya adalah *end-user* aplikasi yaitu mahasiswa Fakultas Teknik sebagai pengguna/pemilih dan admin DPM (Dewan Perwakilan Mahasiswa) sebagai panitia penyelenggara. Metode yang dipakai dalam penentuan responden yaitu metode *simple random sampling.* Metode tersebut digunakan karena dalam pengujiannya nanti akan menyimpulkan statistik dari populasi itu sendiri. Statistik tersebut nantinya akan berupa *pie chart* dengan pengelompokkan dari beberapa populasi yang dipilih secara acak dengan masing-masing jumlah vote. Setelah uji coba sistem telah dilaksanakan maka akan dilakukan evaluasi terhadap sistem tersebut guna untuk mengetahui proses sistem telah sesuai dengan tujuan awal penelitian dan sebagai bahan pertimbangan untuk pengembangan sistem selanjutnya.

3.1.6 Penulisan Laporan

Setelah sistem dirasa sudah memenuhi tujuan awal penelitian maka akan dilakukan penulisan laporan oleh penulis. Semua hal yang berkaitan dengan proses penelitian akan dicantumkan dalam laporan.

3.2 Metode Pengolahan Data

Dalam metode ini berhasil dikumpulkan beberapa *sample* mahasiswa dengan total 10 orang yang mengisi *google form* guna untuk sebagai pemilih dalam pemungutan suara dalam implementasi *e-voting*. Pengolahan data nantinya akan menggunakan metode statistika deskriptif karena keluaran data berupa *pie chart*. Hal tersebut berfungsi sebagai keluaran hasil pemilihan suara yang dilakukan oleh sistem. Berikut data yang didapat :

Tabel 3. 1 Data Peserta Pemungutan Suara

Email	Nama Lengkap	NPM	Program Studi
khusaenifajar484@gmail.com	Khusaeni Fajar	1605040026	Teknik Informatika S1
irulwaritsin45@gmail.com	Khoirul Waritsin	1605040053	Teknik Informatika S1
muhammadridwansulistio@gmail.com	Muhammad Ridwan Sulistyono	1905040005	Teknik Informatika S1
kholidshalahuddin@gmail.com	Muhammad Kholid Shalahuddin	1705040086	Teknik Informatika S1
endinpratama1@gmail.com	Endin Pratama	1705040106	Teknik Informatika S1
irvanrezzapratama@gmail.com	Irvan Reza Pratama	1705040003	Teknik Informatika S1
yusrildisorder5@gmail.com	M Yusril Arrojak	1705040148	Teknik Informatika S1
irvanuciha06@gmail.com	Akhmad Irvan Sihabudin	1805040071	Teknik Informatika S1
afifbagasansori@gmail.com	Afif Bagas Ansori	1805030019	Teknik Otomotif D3
ormawaunimma@gmail.com	Pandu Prasetya Putra	1705040107	Teknik Informatika S1
soniansori28@gmail.com	Soni Ansori	1805040002	Teknik Informatika S1

BAB 5
HASIL DAN PEMBAHASAN

5.1 Hasil

Hasil dari pengujian yang dilakukan sesuai dengan rancangan yang diharapkan dan kebutuhan pengguna sistem yaitu pemilihan suara dapat dilakukan secara *online* dengan tujuan menciptakan pemilihan suara yang yang lebih efisien dalam waktu dan mempermudah pengguna dalam melakukan pemilihan suara. Pengurus dari kegiatan pemungutan suara ini yaitu DPM (Dewan Perwakilan Mahasiswa) juga dapat langsung mendapatkan *report*/rekap data dari sistem ini.

5.1.1 Perangkat Keras dan Perangkat Lunak

Spesifikasi perangkat keras yang digunakan dalam implementasi sistem adalah :

a. Laptop ASUS Vivobook
b. RAM 4 GB
c. Hardisk 1000 GB

Spesifikasi perangkat lunak yang digunakan dalam implementasi sistem adalah :

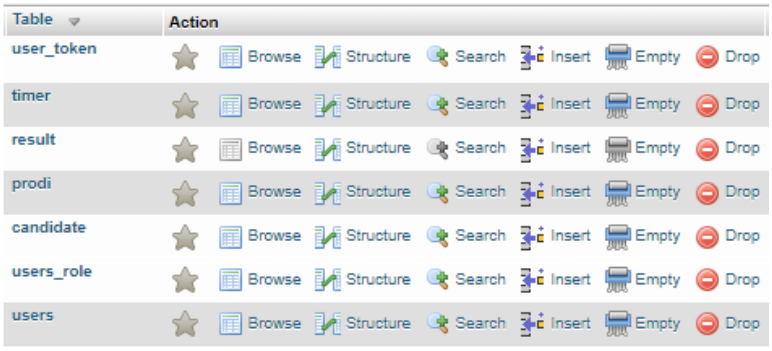
a. Sistem operasi windows 10
b. XAMPP untuk windows
c. Web browser

5.1.2 Implementasi Database

Pada penelitian ini, database yang digunakan adalam MySQL dan proses implementasi database sesuai rancangan yang telah dijabarkan diatas dapat dilihat pada uraian dibawah.

a. Pembuatan Database

Berikut merupakan database yang digunakan sistem *e-voting*. Data tersebut yang nantinya akan digunakan dalam menampung beberapa data yang dipakai dalam sistem.



Gambar 5. 1 Struktur Database E-Voting

b. Tabel Users

Dalam tabel *users* terdapat beberapa atribut yang akan menampung data dari setiap *user*. Tabel ini juga memiliki atribut npm dan *password* untuk akses login dari setiap *user*.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	npm	int(11)			No	None		
3	name	varchar(255)	utf8_general_ci		No	None		
4	prodi_id	int(11)			No	None		
5	email	varchar(255)	utf8_general_ci		No	None		
6	image	varchar(128)	utf8_general_ci		No	None		
7	password	varchar(255)	utf8_general_ci		No	None		
8	role_id	int(11)			No	None		
9	is_active	int(1)			No	None		
10	date_created	int(11)			No	None		

Gambar 5. 2 Tabel Users

c. Tabel Candidate

Dalam tabel candidate terdapat beberapa atribut guna untuk menampung data calon gubernur yang nantinya data akan diambil berdasarkan id dari setiap *user*.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	no_urut	int(11)			No	None		
3	id_ketua	int(11)			No	None		
4	id_wakil	int(11)			No	None		
5	visimisi	longtext	utf8_general_ci		No	None		
6	image	text	utf8_general_ci		No	None		

Gambar 5. 3 Tabel Candidate

d. Tabel Prodi

Dalam Tabel 5.4 ini terdapat beberapa atribut yang nantinya akan diisi data program studi yang ada di Universitas Muhammadiyah Magelang. Data prodi nantinya akan berhubungan dengan tabel users sebagai identitas prodi tiap mahasiswa.

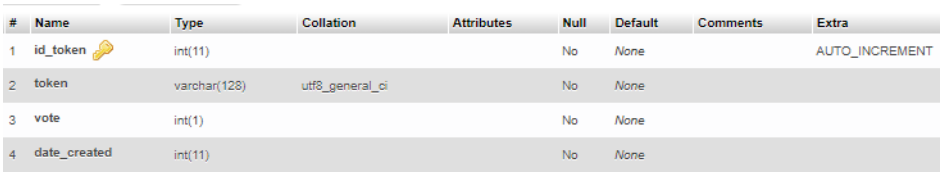


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	name_prodi	varchar(128)	utf8_general_ci		No	None		

Gambar 5. 4 Tabel Prodi

e. Tabel User Token

Dalam Tabel 5.5 ini terdapat beberapa atribut yang nantinya akan menyimpan token dari setiap *user* saat *user* melakukan aktifasi melalui *email.* Token tersebut digunakan sebagai penanda pemilih melakukan vote.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_token	int(11)			No	None		AUTO_INCREMENT
2	token	varchar(128)	utf8_general_ci		No	None		
3	vote	int(1)			No	None		
4	date_created	int(11)			No	None		

Gambar 5. 5 Tabel User token

f. Tabel Result

Dalam Tabel 5.6 ini memiliki beberapa atribut yang menyimpan data calon gubernur dengan jumlah suara yang didapatkan. Tabel tersebut akan berisi total suara dari setiap calon kandidat yang telah terdaftar di sistem.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	id_ketua	int(11)			No	None		
3	id_wakil	int(11)			No	None		
4	vote	int(25)			No	None		

Gambar 5. 6 Tabel Result

g. Tabel Timer

Dalam Tabel 5.7, timer ini nantinya akan menyimpan waktu pemilihan suara yang akan diinputkan oleh admin. Waktu tersebut yang akan menjadi acuan pemilih dalam melakukan pemilihan suara.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id_timer	int(11)			No	None		AUTO_INCREMENT
2	time	text	utf8_general_ci		No	None		

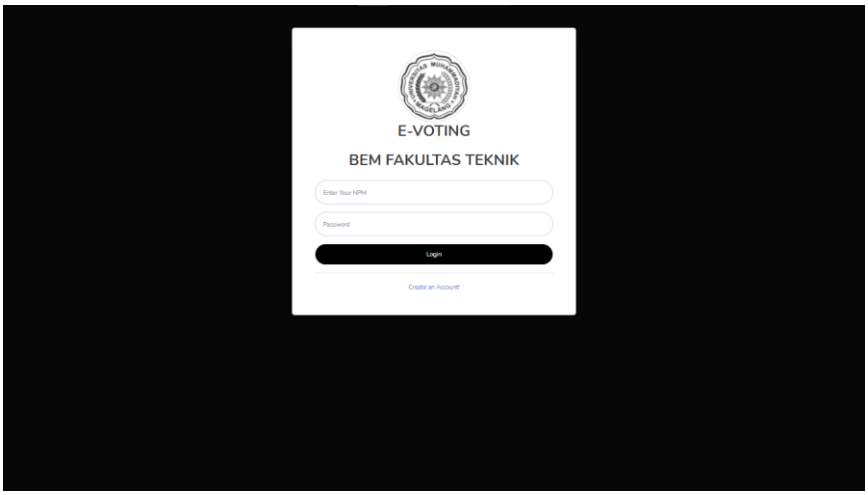
Gambar 5. 7 Tabel Timer

5.1.3 Implentasi *Interface*

Implementasi *Interface* atau layar antarmuka merupakan hasil rancangan i*nterface* yang telah dirancang di bab sebelumnya. Proses ini akan menampilkan layar *login* untuk setiap *user* maupun admin. Nama aplikasi ini adalah E-Voting BEM-FT.

1. *Interface* untuk pengguna/*user*

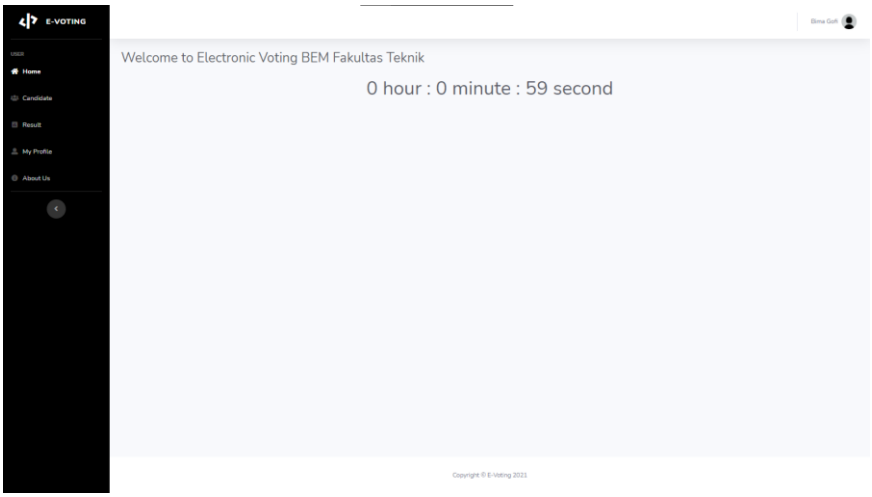
a. Halaman Login



Gambar 5. 8 Halaman Login

Halaman *login* ini merupakan tampilan awal saat *user* akan memulai menggunakan sistem dan *user* telah mendapatkan akun untuk mengakses sistem ini.

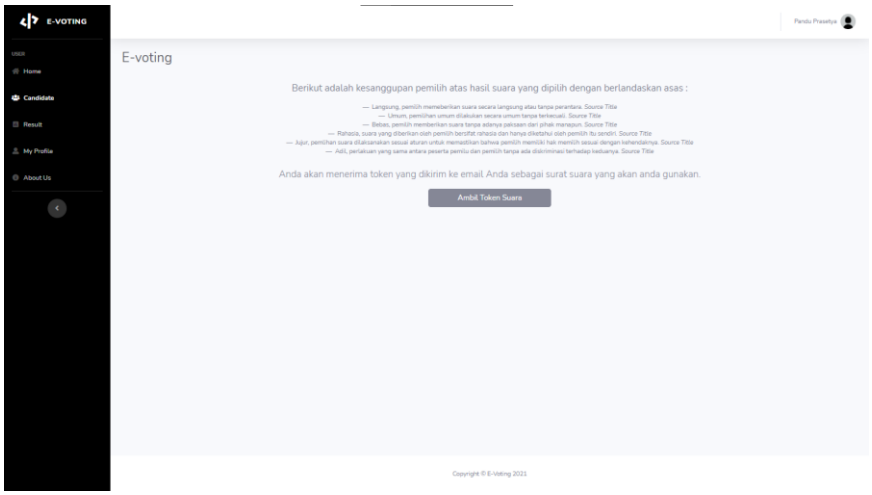
b. Halaman Home



Gambar 5. 9 Halaman Home

Halaman home berisi tentang *count down* dari pemilihan gubernur saat itu. Hal itu difungsikan sebagai penanda terhadap hasil yang nantinya akan langsung dikeluarkan oleh sistem dan *user* dapat melihat langsung melalui aplikasi. Itu juga sebagai hasil bahwa sistem *e-voting* ini lebih efisien dalam waktu, karena hasil pemilihan langsung dapat dilihat *user.*

c. Halaman Token

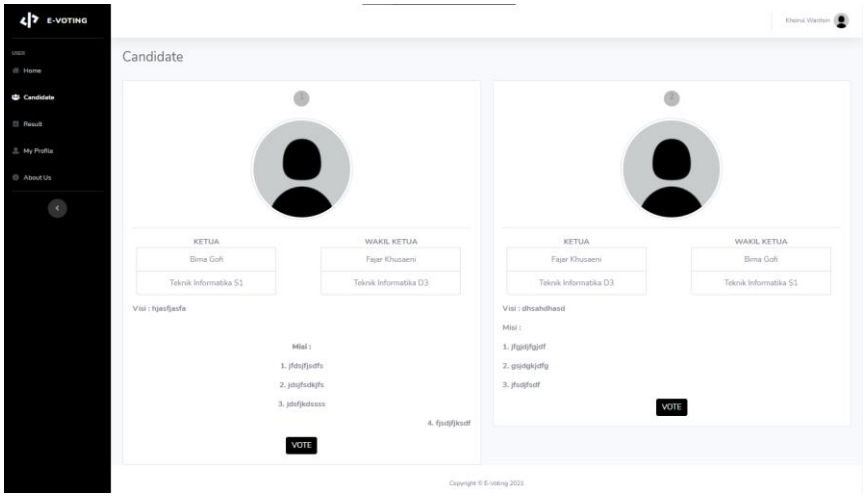


Gambar 5. 10 Halaman Token

Halaman token ini hanya berisi tentang penjelasan *luberjurdil* untuk asas pemilihan suara dan tombol ambil token. Setelah tombol diklik, sistem akan mengirim pesan ke *email* yang sedang login untuk melakukan aktivasi

atau bisa disebut pengambilan surat suara. Token ini juga berfungsi sebagai implementasi dari keakuratan data karena dengan penggunaan *email*, dapat dilakukan pembatasan untuk setiap *user* dalam memilih. *Email* yang diambil adalah *email* yang telah terdaftar di Web Fakultas Teknik.

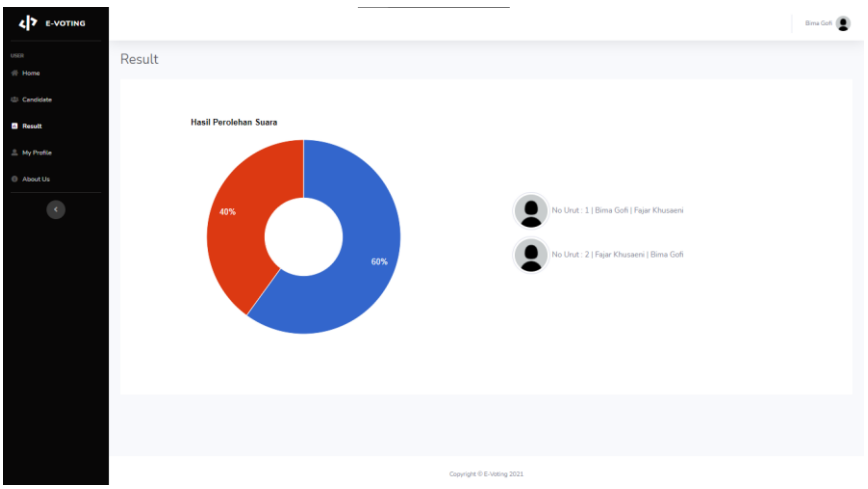
d. Halaman Candidate



Gambar 5. 11 Halaman Candidate

Halaman kandidat berisi tentang calon pasangan gubernur yang dicalonkan pada saat itu setelah *user* aktifasi melalui *email*. Pada halaman tersebut disediakan tombol vote untuk *user* agar langsung bisa memilih pasangan calon.

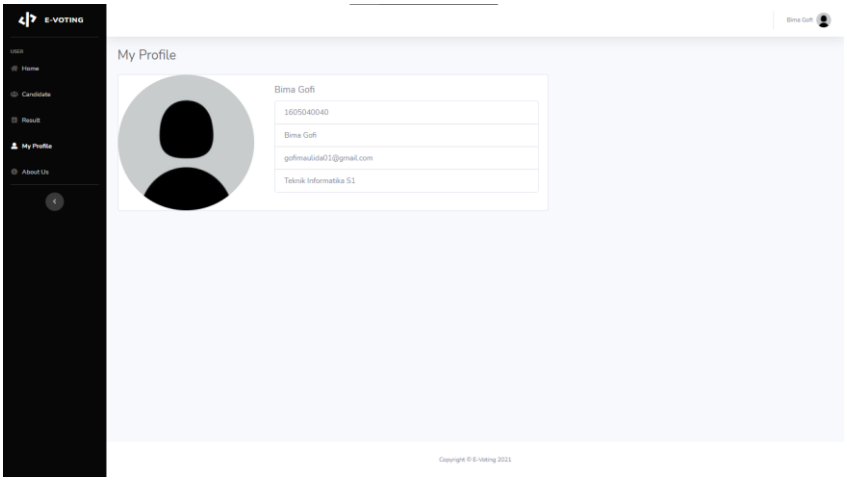
e. Halaman Result



Gambar 5. 12 Halaman Result

Halaman result/hasil ini nantinya akan memuat grafik dari hasil pemilihan suara dengan presntase persen. Hal itu akan berguna untuk informasi ke pemilih terkait calon gubernur dengan suara terbanyak. Halaman itu akan muncul saat *e-voting* telah selesai dilakukan atau mencapai batas waktu. Pada tampilan result ini menghasilkan pemilihan suara yang lebih efisien dalam waktu karena hasil pemilihan langsung dapat dilihat oleh *user*/pemilih saat pemilihan telah berakhir.

f. Halaman Profile

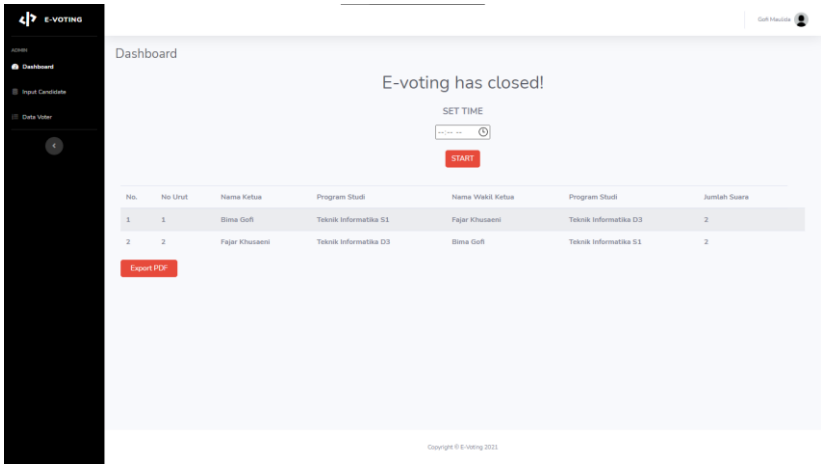


Gambar 5. 13 Halaman Profile

Halaman profile ini hanya berisi tentang informasi *user* yaitu nama, npm, email serta program studi. Dalam halaman ini, *user* tidak bisa mengubah informasi miliknya karena nantinya setiap *account user* akan didapatkan dari web Fakultas Teknik.

2. *Interface* untuk admin

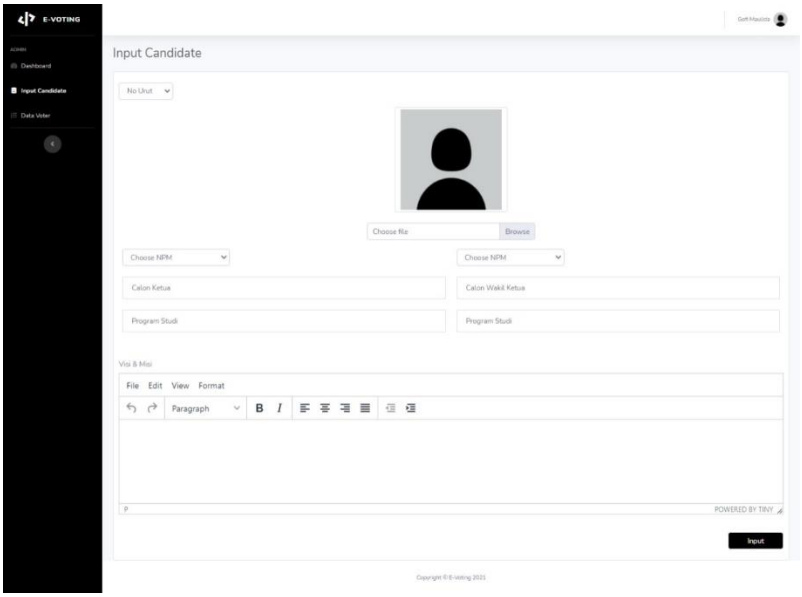
a. Halaman Dashboard



Gambar 5. 14 Halaman Dashboard

Halaman dashboard milik admin ini nantinya akan berisi data calon gubernur dengan jumlah suara dari tiap pasangan. Itu akan digunakan untuk membuat report data pemilihan. Dengan fitur *download pdf* untuk laporan pemilihan dapat dihasilkan bahwa data yang diambil langsung dari sistem yang berjalan, dengan begitu *e-voting* ini menggunakan data yang lebih akurat.

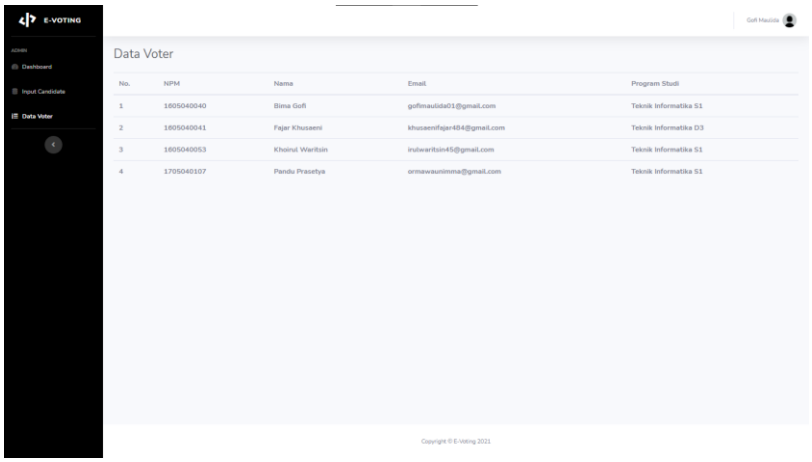
b. Halaman Input Candidate



Gambar 5. 15 Halaman Input Candidate

Halaman input kandidat ini berisi halaman admin yang berfungsi sebagai input data calon pasangan untuk pemilihan gubernur. Inputannya meliputi no urut, foto pasangan, npm, nama, program studi, visi, dan misi.

c. Halaman Data Voter



Gambar 5. 16 Halaman Data Voter

Halaman data voter ini nantinya akan berisi data mahasiswa yang terdaftar sebagai pemilih dari pemungutan suara. Data tersebut didapat dari kuisioner *google form* guna untuk keperluan pengerjaan penelitian ini.

5.1.4 Implementasi Program

a. Penginputan data kandidat

```
public function save_candidat( $data ) {

        $no_urut    = $data['no_urut'];
        $id_ketua   = $data['id_ketua'];
        $id_wakil   = $data['id_wakil'];
        $visimisi   = $data['visimisi'];

        $upload_image = $_FILES['image']['name'];

        $data       = array(
            'no_urut'   => $no_urut,
            'id_ketua'  => $id_ketua,
            'image'     => $upload_image,
            'id_wakil'  => $id_wakil,
            'visimisi'  => $visimisi
        );

        if($upload_image) {
            $config['upload_path']= './assets/img/candidate/';
                $config['allowed_types']='jpg|png';
                $this->upload->initialize($config);

            if(!$this->upload->do_upload('image')) {
                echo $this->upload->display_errors();
            } else {
                $this->upload->data();
            }
        }
        var_dump($config['upload_path']);

        $result     = $this->M_input->save_candidat( $data );

        if ( $result ) {
            redirect( 'admin/input' );
        }
    }
```

Gambar 5. 17 Pengkodean *Controller* Save Kandidat

Pada Gambar 5.17 adalah sebuah *function* untuk pencocokan data ke tabel candidate dengan beberapa atribut. Function tersebut akan menyimpan data inputan admin ke dalam database dengan bantuan model sebagai *query insert.*

```
public function save_candidat( $data ) {

    return $this->db->insert( 'candidate', $data );

}
```

Gambar 5. 18 Pengkodean Model Save Kandidat

Pada Gambar 5.18 menerangkan *query insert* ke database menggunakan bantuan model save_candidat.

```php
public function input()
    {
        if ( ! empty( $this->input->post() ) ) {
            $this->save_candidat( $this->input->post() );
        } else {
            $this->view_input();
        }
    }
```

Gambar 5. 19 Pengkodean Input Data

Pada Gambar 5.19 menjelaskan *controller* dari input data yang mengambil dari *function* save_candidate dan *query insert* ke database dengan model save_candidat.

b. Aktifasi *Email* ke *User*

```php
private function mayabeSendMail($token, $type) {

        $config = [
            'protocol'  => 'smtp',
            'smtp_host' => 'ssl://smtp.googlemail.com',
            'smtp_user' => 'evoting.21@gmail.com',
            'smtp_pass' => 'evote.21',
            'smtp_port' => 465,
            'mailtype'  => 'html',
            'charset'   => 'utf-8',
            'newline'   => "\r\n",
        ];

        $user_data = get_user_data();
        $email = $user_data['email'];

        $this->email->initialize($config);

        $this->email->from('evoting.21@gmail.com', 'Electronic Voting');
        $data = array(
            'message' => $this->input->post('message'),
            'token' => $token,
            'email' => $email
        );

        $this->email->to($email);

        $body = $this->load->view('user/mail', $data, true);

        if($type == 'verify') {
            $this->email->subject('Account Activation');
            $this->email->message($body);
            return $this->email->send();
        }
    }
```

Gambar 5. 20 Pengkodean *Controller Rules Email*

Pada Gambar 5.20 menerangkan *function* untuk penggunaan *email* dari codeigniter 3 untuk dapat melakukan pengiriman *email* dengan custom konfigurasi. Terdapat beberapa *set* untuk dapat menggunakan fitur *email*

tersebut dan memasukkan *email default* yang berfungsi sebagai pengirim pesan ke setiap *email user.*

```php
public function verify()
    {
        $email = $_GET['email'];
        $token = $_GET['token'];

        $user = $this->db->get_where('users', ['email' => $email])->row_array();

        if($user) {
            if($token) {
                    $this->db->set('is_active', 1);
                    $this->db->where('email', $email);
                    $this->db->update('users');

                    $user_token = [
                        'token' => $token,
                        'date_created' => time()
                    ];

                    $data_token = array(
                        'uid'   => $user['id'],
                        'token' => $token
                    );

                    $this->session->set_userdata( 'user_token', $data_token );

                    $exist = $this->db->get_where('user_token',array('token'=>$token))->num_rows();

                    if ( $exist == 0 ) {
                        $this->db->insert('user_token', $user_token);
                    }
                    redirect('user/candidate');
            } else {
                $this->session->set_flashdata('message', '<div class="alert alert-danger" role="alert">
                Wrong token!</div>');
                redirect('auth');
            }
        } else {
            $this->session->set_flashdata('message', '<div class="alert alert-danger" role="alert">
            Activation is failed!</div>');
            redirect('auth');
        }
    }
```

Gambar 5. 21 Pengkodean *Controller Verify Email*

Pada Gambar 5.21 menerangkan *function* send *email* untuk update data *user* jika belum aktif lalu dapat dikirimkan *email*. Pengiriman token akan dilakukan pada *variable* $token dan langsung terinput ke tabel user_token.

```php
public function sendEmail()
    {
        $is_active = false;
        $user_data = get_user_data();
        $is_active = $user_data['is_active'];

        if ( ! $is_active ) {
            $token = urlencode(uniqid());
            $result = $this->mayabeSendMail($token, 'verify');
        } else {
        }

        if ( $result ) {
            $this->db->where( array( 'npm' => $user_data['npm'] ) );
            redirect( 'user/candidate' );
        } else {
        }
    }
```

Gambar 5. 22 Pengkodean *Controller Send Email*

Pada Gambar 5.22 menerangkan *controller* send *email* untuk *check account* pengiriman *email. Function* diatas akan melakukan pengecekan apakah *user* berstatus aktif atau tidak dari *helper class* get_user_data. Jika belum aktif maka akan dikirim *email activation* untuk mendapatkan surat suara dan menerima token yang nantinya akan langsung terkirim ke database.

```php
function get_user_data()
{
    $ci     = get_instance();
    $npm    = $ci->session->userdata('npm');

    $data   =  $ci->db->get_where( 'users',
        ['npm' => $npm]
    )->row_array();

    return $data;
}
```

Gambar 5. 23 Pengkodean Helper

Pada Gambar 5.23 menerangkan *helper* untuk mendapatkan data user dari database (tabel users). *Helper* tersebut nantinya akan dipanggil ke dalam *controller* untuk *checking* data aktif atau tidak.

c. Pengiriman hasil suara/*vote user*

```php
public function save_vote($vote,$token)
    {
        $cek = $this->db->get_where('user_token',array('token'=>$token))->num_rows();
        var_dump($cek);
        if($cek>=1){
            $data = array(
                'vote' => $vote
            );
            $this->db->where('token', $token);
            $this->db->update('user_token', $data);
            var_dump($data);
            return "TRUE";
        }else{
            return "FALSE";
        }
    }
```

Gambar 5. 24 Pengkodean Model Save Vote

Pada Gambar 5.24 menjelaskan tentang model save_vote untuk melakukan update ke tabel user_token terhadap pilihan *user*.

```php
public function save_vote()
    {
        $vote = $this->input->post('votenum');
        $token = $this->input->post('set_token');
        $save = $this->M_input->save_vote($vote,$token);
        if($save == TRUE){
            header("location:../");
        }else{
            header("location:candidate");
        }
}
```

Gambar 5. 25 Pengkodean *Controller* Save Vote

Pada Gambar 5.25 menjelaskan *controller* save_vote yang nantinya akan memanggil model save_vote untuk mengupdate data di user_token.

```php
public function cek_vote()
    {
        $token=($this->session->userdata('user_token'))?$this->session->userdata('user_token')
        ['token']:'';

        $check = $this->db->get_where('user_token',array('token'=>$token))->row_array();
        if($check){
            $is_vote = $check['vote'];
            if($is_vote ==! 0 ){
                $on_vote = 'true';
            }else{
                $on_vote = 'false';
            }
            return $on_vote;
        }
    }
```

Gambar 5. 26 Pengkodean Model Save Vote

Pada Gambar 5.26 menjelaskan model save_vote untuk melakukan pengecekan ke database (tabel user_token) apakah *user* dengan token tersebut telah melakukan vote atau tidak. Token akan dibawa oleh *user* saat *login* dengan menggunakan *session* sebagai penanda *user* untuk melakukan vote.

d. Menampilkan hasil pemilihan suara

```
public function result()
    {
        set_title('Result');
        $data = $this->M_users->get_win();


        foreach ($data as $key => $value) {
            $final_data[] = array(
                'nama'  => sprintf( 'No Urut : %s
                %s | %s', $value['no_urut'], $value['nama_ketua'], $value['nama_wakil'] ),
                'total_vote' => $value['total_vote']
            );
        }

        $output = array(
            'data' => $final_data,
        );

        $this->load->view('user/result', $output);
    }
```

Gambar 5. 27 Pengkodean Controller Hasil Vote

Pada Gambar 5.27 menjelaskan *controller* result untuk menampilkan hasil pemilihan yang nantinya akan muncul dengan grafik. Data hasil pemilihan diambil menggunakan model get_win dengan *query select.*

```
public function get_win()
    {
        return $this->db->query(
            "SELECT count(`vote`) as `total_vote`,
                `c`.`image` as `image`,
                `c`.`no_urut` as `no_urut`,
                `k`.`name` as `nama_ketua`,
                `w`.`name` as `nama_wakil`,
                `pk`.`name_prodi` as `prodi_ketua`,
                `pw`.`name_prodi` as `prodi_wakil`
            FROM
                `candidate` as c
            LEFT JOIN
                `user_token` `n` on `n`.`vote` = `c`.`no_urut`
            LEFT JOIN
                `users` `k` ON `k`.`id` = `c`.`id_ketua`
            LEFT JOIN
                `users` `w` ON `w`.`id` = `c`.`id_wakil`
            LEFT JOIN
                `prodi` `pk` ON `pk`.`id` = `k`.`prodi_id`
            LEFT JOIN
                `prodi` `pw` ON `pw`.`id` = `w`.`prodi_id`
            GROUP BY
                `vote`
            ORDER BY
                `c` .`no_urut`
            ASC"
        )->result_array();
    }
```

Gambar 5. 28 Pengkodean Model Hasil Vote

Pada Gambar 5.28 menjelaskan model get_win dengan *query select* menggunakan COUNT untuk perhitungan jumlah suara dan GROUP BY untuk mengelompokan untuk setiap suara dari calon kandidat. *Query* tersebut menggunakan LEFT JOIN karena data yang diambil dari tabel users, prodi.

```
public function get_candidate() {

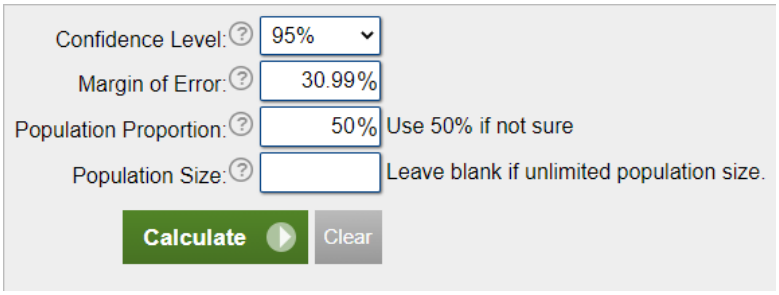
        return $this->db->query(
            "SELECT
                no_urut,
                c.image,
                k.name as nama_ketua,
                w.name as nama_wakil,
                c.visimisi as visi_misi,
                pk.name_prodi as prodi_ketua,
                pw.name_prodi as prodi_wakil
            FROM
                candidate as c
            LEFT JOIN
                users k ON k.id = c.id_ketua
            LEFT JOIN
                users w ON w.id = c.id_wakil
            LEFT JOIN
                prodi pk ON pk.id = k.prodi_id
            LEFT JOIN
                prodi pw ON pw.id = w.prodi_id"
        )->result_array();
    }
```

Gambar 5. 29 Pengkodean Model Kandidat

Pada Gambar 5.29 menjelaskan model get_candidate dengan *query select* untuk mendapatkan data yang telah diinput oleh admin di tabel candidate. *Query* tersebut juga menggunakan LEFT JOIN karena data yang dinput admin hanya berupa npm yang diambil dari tabel users.

5.1.5 Implementasi Pengujian Beta *Testing*

Pengujian yang melibatkan mahasiswa yang termasuk ke dalam data *kuisioner google form* berfungsi untuk mengetahui tingkat kesesuaian sistem dengan fungsi kemudahan yang ditawarkan sistem. Pengujian dilakukan dengan responden sejumlah 10 orang yang bertindak sebagai pemilih. Total responden didapat menggunakan *simple random sampling* dengan melakukan perhitungan dengan *sample size calculator* dengan menentukan tingkat kepercayaan *(confidence level)*, interval error *(margin of error)*, proporsi populasi *(population proportion)*. Data tersebut sebagai berikut.



Instrumen penelitian untuk pengujian *user* adalah sebagai berikut :

1. Sangat Kurang
2. Kurang
3. Cukup
4. Baik
5. Sangat Baik

Tabel 5. 1 Tabel Beta Testing User

No	Aktifitas	Kegiatan yang dikehendaki	Ketercapaian				
			1	2	3	4	5
1.	Login	Login berhasil masuk ke home sesuai user					
2.	Logout	Logout berhasil					

3.	Melihat data kandidat	Melihat pasangan calon gubernur di halaman Candidate dengan data no urut, foto calon, npm, nama calon ketua dan wakil serta visi dan misi					
4.	Melakukan vote	Melakukan vote pasangan calon gubernur					
5.	Melihat result/hasil	Melihat data hasil pemilihan suara berupa grafik					
6.	Melihat profile	Melihat biodata user sendiri					

Tabel 5. 2 Tabel Beta Testing Admin/DPM

No	**Aktifitas**	**Kegiatan yang dikehendaki**	**Ketercapaian**				
			1	**2**	**3**	**4**	**5**
1.	Login	Login berhasil masuk ke Dashboard Admin					
2.	Logout	Logout berhasil					
3.	Melihat data hasil pemilihan suara	Melihat hasil pemilihan suara berupa tabel dengan jumlah suara dari setiap pasanga calon dan dapat melakukan set waktu pemilihan					
4.	Export data hasil pemilihan	Dapat mendownload hasil pemilihan suara berupa file PDF					
5.	Input data kandidat	Melakukan input data kandidat di form Candidate dengan inputan sesuai kebutuhan admin					
6.	Melihat data mahasiswa	Melihat data mahasiswa/peserta pemilihan suara yang telah terdaftar dengan komponen npm, nama, email, dan program studi					

5.1.6 Hasil Pengujian Sistem

Hasil pengujian sistem dilakukan dengan pengujian *Black Box* yaitu dengan memperhatikan setiap fungsi pada sistem apakah dapat berjalan sesuai dengan yang diharapkan. Dengan menggunakan tabel pengujian dapat disimpulkan bahwa sistem *e-voting* yang dibuat dapat berfungsi sesuai dengan yang diharapkan dan dapat memenuhi kebutuhan seperti pada hasil analisa pada Bab 1 sampai Bab 4.

Data pengujian *Black Box* didapatkan setelah dilakukan pengujian langsung ke *user*/pemilih guna untuk mendapat respon langsung dari mahasiswa yang nantinya akan terlibat sebagai pengguna sistem *e-voting* BEMFT UNIMMA.

Tabel 5. 3 Tabel *Black Box* Pengujian Sistem

No. (1)	Rancangan Proses (2)	Hasil Yang Diharapkan (3)	Pengamatan (4)	Kesimpulan (5)
1.	Mulai jalankan program	Masuk halaman login	Ketika program dijalankan langsung masuk ke halaman login	Sesuai
2.	Klik menu Candidate	Masuk halaman Candidate dan muncul tombol ambil token	Ketika klik Candidate, maka akan muncul halaman ambil token	Sesuai
3.	Aktivasi email dengan membuka email masing-masing user	Muncul pesan untuk aktivasi akun	Muncul pesan ke email user dan menekan link ACTIVATE	Sesuai
4.	Melakukan vote terhadap pasangan calon gubernur	Masuk ke halaman Candidate dan muncul tombol VOTE	Setelah klik ACTIVATE maka akan dialihkan ke halaman Candidate	Sesuai
5.	Klik menu Result	Masuk ke halaman Result untuk melihat	Setelah klik menu Result maka akan	Sesuai

		hasil pemilihan suara	muncul grafik dari perhitungan suara	
6.	Klik menu My Profile	Masuk ke halaman My Profile	Setelah klik menu My Profile maka akan muncul data diri dari masing-masing user yang sedang login	Sesuai

Pada Tabel 5.2 dilakukan dengan melakukan skenario pada fitur yang disediakan sistem dengan menggunakan cara pengujian *Black Box* diatas.

Hasil pengujian berdasarkan form yang dibagikan dilakukan oleh 10 orang mahasiswa, dengan cara melakukan pengoperasian sistem dan menggunakan semua fitur yang ada. Setelah data pengujian, selanjutnya dihitung presentase jawaban responden dengan menggunakan rumus :

Persentase = jumlah skor yang diperoleh/jumlah skor tertinggi x 100%

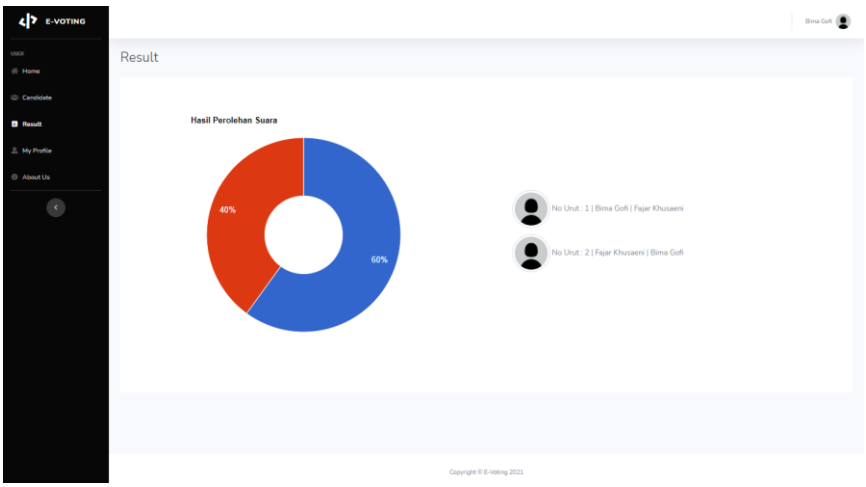
Tabel 5. 4 Hasil Beta Testing User/Mahasiswa

No	Aktifitas	Kegiatan yang dikehendaki	Skor	Persentasi
1.	Login	Proses login berhasil dilakukan oleh user	45	90%
2.	Logout	Proses logout berhasil dilakukan oleh user	46	92%
3.	Melihat data kandidat	Melihat pasangan calon gubernur di halaman Candidate dengan data no urut, foto calon, npm, nama calon ketua dan wakil serta visi dan misi	39	78%
4.	Melakukan vote	Melakukan vote pasangan calon gubernur	45	90%
5.	Melihat result/hasil	Melihat data hasil pemilihan suara berupa grafik	44	88%
6.	Melihat profile	Melihat biodata user sendiri	44	88%
Rata - Rata			43.83	87.67%

Tabel 5. 5 Hasil Beta Testing Admin/DPM

No	Aktifitas	Kegiatan yang dikehendaki	Skor	Persentasi
1.	Login	Login berhasil masuk ke Dashboard Admin	5	100%
2.	Logout	Logout berhasil	5	100%
3.	Melihat data hasil pemilihan suara	Melihat hasil pemilihan suara berupa tabel dengan jumlah suara dari setiap pasanga calon dan dapat melakukan set waktu pemilihan	4	80%
4.	Export data hasil pemilihan	Dapat mendownload hasil pemilihan suara berupa file PDF	4	80%
5.	Input data kandidat	Melakukan input data kandidat di form Candidate dengan inputan sesuai kebutuhan admin	4	80%
6.	Melihat profile	Melihat data mahasiswa/peserta pemilihan suara yang telah terdaftar dengan komponen npm, nama, email, dan program studi	4	80%
Rata - Rata			4.3	86.67%

a. Keluaran Hasil Perhitungan Suara



Gambar 5. 30 Hasil Perhitungan Suara

Dari hasil pengujian pada form result ini mampu menampilkan hasil perhitungan suara berupa grafik persentase dari setiap pasangan calon gubernur. Pada form ini, hasil diambil dari total vote pemilih kepada setiap calon berdasarkan database yang tersimpan di tabel user_token dengan melakukan pengelompokke dari setiap pilihan suara. Hal ini membuktikan bahwa sistem *e-voting* ini lebih efisien dalam segi waktu untuk pemilihan suara karena hasil dapat langsung dilihat oleh peserta pemilihan suara.

b. Keluaran Aktivasi Email



Gambar 5. 31 *Notifikasi Aktivasi Email*

Pada Gambar 5.31 menunjukkan notifikasi *email* saat *user* mengambil surat suara/token. Email akan terkirim sesuai ke *user* yang sedang melakukan *login*. Penggunaan aktivasi *email* ini nantinya akan mengambil data *email* yang telah terdaftar di Fakultas Teknik, dengan begitu data yang digunakan lebih akurat karena langsung mengambil data dari fakultas terkait.

c. Keluaran Laporan Hasil Pemilihan Suara

LAPORAN PEMILIHAN UMUM

BEM-FT UNIMMA

No Urut	Nama Ketua	Program Studi	Nama Wakil Ketua	Program Studi	Jumlah Suara
1	Bima Gofi	Teknik Informatika S1	Fajar Khusaeni	Teknik Informatika D3	3
2	Fajar Khusaeni	Teknik Informatika D3	Bima Gofi	Teknik Informatika S1	2
3	Fajar Khusaeni	Teknik Informatika D3	Pandu Prasetya	Teknik Informatika S1	0

Gambar 5. 32 Laporan Format PDF

Pada Gambar 5.32 merupakan keluaran dari fitur export PDF di halaman Dashboard admin guna untuk rekap data. Fitur tersebut terbukti lebih efisien dari segi waktu karena laporan langsung didapat dari sistem *e-voting*.

5.2 Pembahasan

Berdasarkan hasil implementasi dan pengujian dapat dibahas dengan uraian sebagai berikut.

5.2.1 Pembahasan Sistem E-Voting

Berdasarkan pengujian *e-voting* pada sistem pemungutan suara di BEMFT. Dapat disimpulkan bahwa sistem dapat berjalan dengan baik sesuai kebutuhan pengguna. Keunggulan sistem ini adalah user/mahasiswa dapat melakukan vote/coblos online dimana saja dikarenakan sistem ini yang berjalan secara online. Dibandingkan dengan sistem pemungutan suara sebelumnya, *e-voting* dapat memenuhi beberapa kekurangan pada pemungutan suara sebelumnya yaitu dari segi waktu pemungutan menjadi lebih efisien karena hasil perhitungan dilakukan secara otomatis oleh sistem dan data yang digunakan sistem lebih akurat karena menggunakan verifikasi *email* sebagai batasan tiap *user* untuk melakukan pemilihan dengan hak suara satu kali. *Email* yang digunakan diambil langsung dari data Fakultas Teknik yang telah terdaftar pada setiap *user*. Untuk keamanan hasil pilihan suara dari setiap mahasiswa juga terjamin karena adanya aktivasi email dari masing-masing pengguna dan data/pilihan suara yang masuk akan secara otomatis terkirim ke database dengan value/nilai pilihan suara saja. Ditambah hasil pemilihan suara juga dapat dilihat langsung secara online saat pemilihan telah selesai dilakukan.

5.2.2 Pengujian Beta Testing User

Kriteria penilaian untuk instrumen beta testing menggunakan skala Likert dengan memberikan lima pilihan jawaban. Kriteria penilaian ini menjelaskan tentang beberapa nilai yang terkandung dari beberapa kriteria dengan disimbolkan nilai angka. Penilaian dengan skala Likert dapat dilihat pada tabel berikut.

Tabel 5. 6 Kriteria Penilaian Pengujian Beta Testing User

Kriteria	Nilai
Tidak Sesuai	1
Kurang Sesuai	2
Netral	3
Sesuai	4
Sangat Sesuai	5

Data kemudian dikonversi berdasarkan kriteria interpretasi skor sebagai berikut.

Tabel 5. 7 Pedoman Interpretasi Skor Setelah Dikonversi

Angkat (dalam %)	Kategori
0 - 20	Sangat Tidak Layak
21 – 40	Tidak Layak
41 – 60	Cukup
61 – 80	Layak
81 – 100	Sangat Layak

Persentase = **total skor dari setiap poin survei/total poin survei**

1. Persentase User = **263/6**
2. Persentase Admin = **26/6**

Berdasarkan data tabel hasil pengujian *Beta Testing* kepada user/mahasiswa didapatkan persentase 87.67% dan *Beta Testing* kepada DPM sebagai admin didapatkan persentase 86.67%. Persentase tersebut dapat dijelaskan pada poin 1 dan 2 sebagai perhitungan. Dapat disimpulkan bahwa sistem yang dikembangkan ini mampu membantu/menggantikan pemilihan suara pada BEMFT Fakultas Teknik saat ini dengan beberapa kelebihan yang telah disebutkan di bab sebelumnya.

BAB 6
KESIMPULAN DAN SARAN

6.1 Kesimpulan

Dari pembahasan yang sudah diuraikan maka dapat disimpulkan sebagai berikut :

1. Dengan sistem *e-voting* pada pemilihan suara calon Gubernur BEMFT UNIMMA, dapat disimpulkan bahwa *e-voting* ini dapat berjalan sesuai prosedur pemilihan suara dan dapat memangkas beberapa faktor antara lain penggunaan kertas menjadi lebih sedikit karena semua kertas suara sudah tercetak secara online melalui sistem, waktu yang efisien dalam pemungutan suara karena dapat dilakukan dimana saja dan hasil perhitungan suara juga langsung dapat dilihat oleh peserta pemilihan, serta rekap data yang langsung dapat diperoleh saat pemilihan telah berakhir dengan melakukan cetak *pdf* pada sistem sebagai laporan pemilihan suara.
2. Laporan pemilihan suara yang nantinya dikelola oleh DPM Fakultas Teknik dapat langsung di *download* dan di cetak sehingga mempermudah dalam kinerja rekap data.
3. Mahasiswa dapat mengakses pemilihan suara BEMFT melalui sistem *e-voting* ini dengan membuka aplikasi *e-voting* ini saat pemilihan telah dibuka sehingga informasi tentang pemilihan suara lebih cepat tersampaikan.

6.2 Saran

Dari hasil kesimpulan diatas, sistem *e-voting* ini memiliki potensi besar untuk dikembangkan lagi mengingat era digital semakin maju, oleh karena itu beberapa saran kiranya adalah sebagai berikut :

1. Sistem *e-voting* ini dapat dikembangkan pada bagian keamanan yaitu agar kerahasiaan pilihan suara lebih terjamin.
2. Sistem ini juga dapat dikembangkan untuk lingkup yang lebih luas.

DAFTAR PUSTAKA

Adi Hermawansyah. (2019). *Penyimpanan Data Data Pengolah Informasi*. *2*(April), 37–43.

Azwanti, N. (2017). Perancangan E-Voting berbasis Web. *Jurnal Komputer Terapan*, *3*(2), 119–132.

Lavarino, D. dan W. Y. (2016). RANCANG BANGUN E-VOTING BERBASIS WEBSITE DI UNIVERSITAS NEGERI SURABAYA. *Manajemen Informatika*, *6*(August), 72–81.

Nuryanto, E. P., & Wintoro, P. B. (2017). Rancang Bangun Sistem E-Voting Pada Pemilihan Raya (Pemira) Iib Darmajaya Berbasis Web. *Semnas Iib Darmajaya*, 90–100.

Ridwan, M., Arifin, Z., & Yulianto, Y. (2016). Rancang Bangun E-Voting Dengan Menggunakan Keamanan Algoritma Rivest Shamir Adleman (RSA) Berbasis Web (Studi Kasus : Pemilihan Ketua Bem Fmipa). *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer*, *11*(2), 22. https://doi.org/10.30872/jim.v11i2.210

Rofiq, F. N. (2017). Implementasi RESTful Web Service untuk Sistem Penghitungan Suara Secara Cepat pada Pilkada Implementation of Restful Web Servicefor Quick Count System of MayorElection. *Eksplora Informatika*, *6*(2), 159–168.

Rotikan, R., Melaira, J. G., Rarumangkay, R. R., & Lengkong, O. (2020). Penerapan PWA Untuk Aplikasi E-Voting Pemilihan Pengurus BEM dan Senat di Universitas Klabat. *CogITo Smart Journal*, *6*(2), 190. https://doi.org/10.31154/cogito.v6i2.272.190-203

Wisnu, Dyah A. M. G., Aswin Suharsono, S.T., M.T., M.Kom., Denny S. R., S.Kom., M. K. (2012). Rancang Bangun Sistem E-Voting Dengan Menerapkan Hash Dan Digital Signature Untuk Verifikasi Data Hasil Voting. *Program Studi Informatika Program Teknologi Informasi Dan Ilmu Komputer Universitas Brawijaya Malang*. http://jurnal-online.um.ac.id/data/artikel/artikelE67FDF9F7E69BEB865ABE1DC9CD3E489.pdf

Zulkiawan, A., Dengen, N., Puspitasari, N., & Aksenta, A. (2020). *Penerapan Metode User Authentication Pada Sistem*. *4*(2).