

**RANCANG BANGUN *BACKEND WEBSITE* PEMUNGUTAN SUARA  
DENGAN MENGGUNAKAN *FRAMEWORK EXPRESS.JS***

**(Skripsi)**

**Oleh**

**MUHAMMAD NAUVAL SAIHOLAU  
NPM 1915061057**



**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
2023**

**RANCANG BANGUN *BACKEND WEBSITE* PEMUNGUTAN SUARA  
DENGAN MENGGUNAKAN *FRAMEWORK EXPRESS.JS***

**Oleh**

**MUHAMMAD NAUVAL SAIHOLAU**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
SARJANA TEKNIK**

**Pada**

**Jurusan Teknik Elektro  
Fakultas Teknik Universitas Lampung**



**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
2023**

## **ABSTRAK**

### **RANCANG BANGUN BACKEND WEBSITE PEMUNGUTAN SUARA DENGAN MENGGUNAKAN FRAMEWORK EXPRESS.JS**

**Oleh**

**MUHAMMAD NAUVAL SAIHOLAU**

Dalam proses demokrasi, pemungutan suara merupakan cara yang penting untuk menentukan pilihan. Pemungutan suara secara tradisional memiliki banyak permasalahan, seperti potensi kesalahan, biaya yang besar, dan waktu yang lama. Karena itu, sistem pemungutan suara online yang lebih maju dan efektif sangat diperlukan. Sudah ada beberapa penelitian dan pengembangan sistem pemungutan suara online. Namun, sistem tersebut masih kurang sempurna, karena tidak memperhatikan keamanan data, tidak menggunakan desain yang responsif, dan kurang memiliki fleksibilitas. Penelitian ini memiliki tujuan untuk merancang dan membangun *backend* dari situs web pemungutan suara yang bernama PilihGeh dengan menggunakan *framework* Express.Js yang mendukung keamanan, validitas suara, dan sistem yang fleksibel. Metode Iterative Development Model digunakan dalam pelaksanaan dan perancangan penelitian ini dengan membagi proses menjadi bagian kecil yaitu iterasi. Proses penelitian mencakup analisis, desain sistem, coding, testing, dan deployment. Pada tahap analisis, data-data yang diperlukan untuk membuat sistem dikumpulkan untuk mendapatkan kebutuhan fungsional dan non fungsional. Berdasarkan analisis kebutuhan, tahap desain sistem melakukan perancangan sistem dengan menggunakan UML yang terdiri dari Use Case Diagram, Activity Diagram, dan Entity Relationship Diagram. Selanjutnya,

dilakukan tahap coding dengan menggunakan framework Express.js untuk empat iterasi. Kemudian, dilakukan pengujian fungsionalitas dengan menggunakan aplikasi Postman dengan metode blackbox testing untuk setiap iterasi. Terakhir, dilakukan deployment backend yang telah dibangun dengan menggunakan layanan Railway. Sistem backend ini mendukung fitur autentikasi, penyelenggaraan, pengelolaan, dan pelaksanaan pemungutan suara yang fleksibel, fitur mengelola profil serta fitur enkripsi. Penelitian ini berhasil untuk mengembangkan backend situs web pemungutan suara online yang bernama PilihGeh. Backend situs web ini dibangun dengan menggunakan framework Express.js dan metode *Iterative Development Model*. Backend situs web ini menyediakan fitur-fitur seperti fitur login dan register, fitur pemungutan suara dan hasil suara, dan fitur edit profil. Backend situs web ini juga diuji fungsionalitasnya dengan menggunakan aplikasi Postman dengan metode blackbox testing untuk 132 skenario. Hasil pengujian menunjukkan bahwa semua skenario berjalan sesuai dengan yang diharapkan. Backend situs web ini juga berhasil di deploy dengan menggunakan layanan Railway dengan tautan <https://backend-pilihgeh-production.up.railway.app/>.

Kata kunci: Pemungutan Suara Online, Backend. Express.js, Iterative Development Model. Postman, Railway

## **ABSTRAK**

### **DESIGN AND DEVELOPMENT OF BACKEND WEBSITE FOR VOTING USING EXPRESS.JS FRAMEWORK**

**Oleh**

**MUHAMMAD NAUVAL SAIHOLAU**

*In the process of democracy, voting is an important way to determine the choice. However, traditional voting has many problems, such as the risk of errors, high costs, and long time. Therefore, a more advanced and effective online voting system is needed. Several studies and development of online voting systems have been conducted. However, the system still has weaknesses, such as not prioritizing data security, not using responsive design, and lacking flexibility. This study aims to design and build the backend of a voting website called PilihGeh using the Express.js framework that supports security, validity of votes, and a flexible system. The Iterative Development Model method is used in the implementation and design of this study by dividing the process into small parts called iterations. The research process includes analysis, system design, coding, testing, and deployment. In the analysis stage, data needed to create the system are collected to obtain functional and non-functional requirements. Based on the analysis of requirements, the system design stage designs the system using UML consisting of Use Case Diagrams, Activity Diagrams, and Entity Relationship Diagrams. Next, the coding stage is performed using the Express.js framework for four iterations. Then, functionality testing is performed using the Postman application with the blackbox testing method for each iteration. Finally, the deployment of the backend that has been*

*built using the Railway service is carried out. The backend system supports authentication, organization, management, and implementation of flexible voting features, profile management features and encryption features. This study succeeded in developing a backend website for online voting called PilihGeh. The backend website is built using the Express.js framework and Iterative Development Model method. The backend website provides features such as login and register features, voting and voice results features, and profile editing features. The backend website is also tested for functionality using the Postman application with the blackbox testing method for 132 scenarios. The test results show that all scenarios run as expected. The backend website was also successfully deployed using the Railway service with the link <https://backend-pilihgeh-production.up.railway.app/>.*

Keywords: Online Voting, Backend. Express.js, Iterative Development Model. Postman, Railway

Judul Skripsi : **RANCANG BANGUN *BACKEND*  
WEBSITE PEMUNGUTAN SUARA  
DENGAN MENGGUNAKAN  
FRAMEWORK EXPRESS.JS**

Nama Mahasiswa : **Muhammad Nauval Saiholau**

Nomor Pokok Mahasiswa : 1915061057

Program Studi : Teknik Informatika

Fakultas : Teknik

**MENYETUJUI**

**1. Komisi Pembimbing**

Pembimbing Utama

Pembimbing Pendamping



**Wahyu Eko Sulistiono, S.T., M.Sc.**  
NIP 19741201 200112 1 001

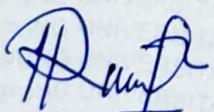


**Yessi Mulyani, S.T., M.T.**  
NIP 19731226 200012 2 001

**2. Mengetahui**

Ketua Jurusan  
Teknik Elektro

Ketua Program Studi  
Teknik Informatika



**Herlinawati, S.T., M.T.**  
NIP 19710314 199903 2 001

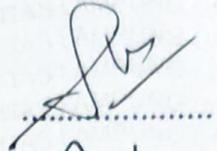


**Yessi Mulyani, S.T., M.T.**  
NIP 19731226 200012 2 001

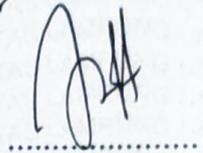
## MENGESAHKAN

### 1. Tim Penguji

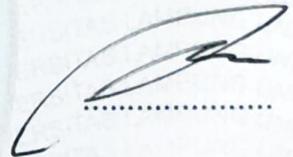
Ketua : **Wahyu Eko Sulistiono, S.T., M.Sc.**



Sekretaris : **Yessi Mulyani, S.T., M.T.**



Penguji : **Mona Arif Muda, S.T., M.T.**



### 2. Dekan Fakultas Teknik



**Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. }**  
NIP 19750928 200112 1 002

Tanggal Lulus Ujian Skripsi : **30 Oktober 2023**

## SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini, menyatakan bahwa skripsi saya dengan judul "Rancang Bangun Backend Website Pemungutan Suara Dengan Menggunakan Framework Express.js" dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh oranglain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 30-10-2023

Pembuat Pernyataan



Muhammad Nauval Saiholau

NPM. 1915061057

## RIWAYAT HIDUP



Penulis, anak kedua dari Bapak Ahmad Ependi, S.Si. dan Ibu Rosmi Susilawati, M.Pd. lahir di Tangerang pada 3 Juni 2001. Penulis menamatkan SD Islam Mutiara Insani (2013), Pesantren Darul Quran Mulia (2016), dan SMA Negeri 22 Kabupaten Tangerang (2019). Pada tahun 2019, penulis masuk Program Studi Teknik Informatika, Fakultas Teknik Universitas Lampung melalui SBMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan, antara lain:

1. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Komunikasi dan Informasi, Divisi Media Informasi pada tahun 2019 hingga 2020.
2. Melaksanakan Kuliah Kerja Nyata di Desa Pasir Kupa Kalanganyar, Kecamatan Lebak, Kabupaten Banten pada bulan Januari sampai dengan Februari 2022.
3. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya dengan mengambil kelas Microcredential Game Developer dari ICE Institute pada periode 15 Februari 2022 sampai dengan 1 Juli 2022.
4. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya dengan mengambil kelas Node JS for Back End Web Developer dari Hacktiv8 pada periode 18 Agustus 2022 sampai dengan 31 Desember 2022
5. Melakukan kerja praktik di Kementerian Ketenagakerjaan Republik Indonesia sebagai *Front End Developer* pada tahun 2022.

## **MOTTO**

“Boleh jadi kamu membenci sesuatu, padahal ia amat baik bagimu, dan boleh jadi (pula) kamu menyukai sesuatu, padahal ia amat buruk bagimu; Allah mengetahui, sedang kamu tidak mengetahui.”

**(QS. Al-Baqarah: 216)**

““Karena sesungguhnya sesudah kesulitan ada kemudahan. Sesungguhnya sesudah kesulitan ada kemudahan.”

**(QS. Al-Inshirah: 5-6)**

"Allah akan membantu hamba-Nya selama hamba-Nya membantu agama-Nya dan jalan-Nya."

**(Abu Bakar Ash-Shiddiq)**

"Tetap mengalir, seperti aliran air, melalui setiap liku kehidupan dengan ketabahan dan ketulusan."

**(Penulis)**

## **PERSEMBAHAN**

**Bismillaahirrohmaanirrahim,**

**Dengan mengucapkan syukur kepada Allah SWT,**

**Kutujukan karya tulis ini untuk orang-orang yang kucintai dengan sepenuh jiwa:**

**Orangtua tersayang,**

**Keluargaku,**

**Sahabat-Sahabatku,**

**Dan Orang-orang yang telah mewarnai hidupku**

**Terima kasih atas segala kasih dan dukungan kalian,**

**Kalian adalah anugerah terindah yang kudapatkan**

## SANWACANA

Puji syukur ke hadirat Allah SWT yang telah melimpahkan rahmat dan hidayat-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi/tugas akhir ini dengan judul “Rancang Bangun Backend Website Pemungutan Suara Dengan Menggunakan Framework Express.js”. Dalam pelaksanaan dan pembuatan skripsi/tugas akhir ini penulis menerima dukungan baik secara moril maupun materil yang sangat berharga dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah membantu, khususnya kepada:

1. Orangtua tercinta, Bapak Ahmad Ependi, S.Si. dan Ibu Rosmi Susilawati, M.Pd., kakak dan adik serta seluruh keluarga penulis yang tak pernah lelah mendoakan, mendukung, dan membantu penulis;
2. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
3. Ibu Herlinawati, S.T., M.T., selaku Ketua Jurusan Teknik Elektro Universitas Lampung;
4. Ibu Yessi Mulyani, S.T., M.T., selaku Ketua Program Studi Teknik Informatika Universitas Lampung sekaligus Pembimbing Pendamping yang memberikan dukungan serta bimbingan agar menjadi lebih baik,
5. Bapak Wahyu Eko Sulistiono, S.T., M.Sc., selaku Pembimbing Utama yang selalu meluangkan waktunya untuk memberikan bimbingan dan dukungan serta memudahkan penulis dalam menyelesaikan penelitian ini.
6. Ibu Dr. Eng. Mardiana, S.T., M.T., selaku Pembimbing Akademik yang selalu memberikan bimbingan dan dukungan kepada penulis selama menyelesaikan perkuliahan;
7. Bapak Mona Arif Muda, S.T., M.T., selaku Penguji yang telah membantu proses kelancaran pengerjaan penelitian dan memberikan banyak saran serta

masukan terhadap penelitian ini;

8. Mbak Rika selaku Admin Program Studi Teknik Informatika yang telah banyak membantu penulis dalam segala urusan administrasi selama perkuliahan;
9. Seluruh dosen dan staf Jurusan Teknik Informatika Unila yang memberi masukan dan mempermudah proses pembuatan skripsi / tugas akhir ini;
10. Sobat Kriyay, Muhammad Bintang Pramadipta, Muhammad Akram Al Habsi, Muhammad Ranzha Nicola, dan Ferry Muhamad yang selalu memberikan dukungan dan bantuan kepada penulis selama menjalani perkuliahan;
11. Keluarga besar Teknik Elektro Angkatan 2019 yang telah menjadi teman seperjuangan sejak mahasiswa baru. Terimakasih telah mewarnai masa perkuliahan penulis.

Penulis berharap agar laporan ini dapat menjadi referensi bagi pengembangan keilmuan di bidang teknik informatika. Oleh karena itu, semoga penelitian ini bermanfaat bagi yang membacanya.

Bandar Lampung, 30-10-2023

Penulis,



Muhammad Nauval Saiholau

## DAFTAR ISI

<b>DAFTAR ISI.....</b>	<b>i</b>
<b>DAFTAR GAMBAR.....</b>	<b>iv</b>
<b>DAFTAR TABEL.....</b>	<b>ix</b>
<b>BAB 1    PENDAHULUAN.....</b>	<b>1</b>
1.1    Latar Belakang .....	1
1.2    Rumusan Masalah .....	2
1.3    Batasan Masalah.....	2
1.4    Tujuan .....	3
1.5    Sistematika Penulisan.....	3
<b>BAB 2    TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1    Website.....	5
2.2    Rancang Bangun Website .....	5
2.3    Framework .....	6
2.3.1    NodeJs .....	7
2.4    Application Programming Interface (API).....	8
2.4.1    REST .....	9
2.5    Database .....	10
2.5.1    PostgreSQL .....	10
2.6    Visual Studio Code .....	11
2.7    Unified Modelling Language .....	12
2.8    Iterative Development Model .....	13

2.9	Penelitian Terdahulu .....	14
<b>BAB 3</b>	<b>METODOLOGI PENELITIAN .....</b>	<b>16</b>
3.1	Waktu Jadwal dan Tempat Rancang Bangun.....	16
3.2	Alat dan Bahan.....	17
3.3	Tahapan Dalam Pelaksanaan Tugas Akhir .....	18
3.3.1	Analisis.....	18
3.3.2	Desain Sistem.....	20
3.3.3	Pengodean .....	44
3.3.4	Pengujian.....	44
3.3.5	Implementasi .....	45
<b>BAB 4</b>	<b>HASIL DAN PEMBAHASAN .....</b>	<b>46</b>
4.1	Hasil .....	46
4.1.1	Iterasi 1.....	46
4.1.2	Iterasi 2.....	48
4.1.3	Iterasi 3.....	51
4.1.4	Iterasi 4.....	61
4.2	Pembahasan.....	69
4.2.1	Metode <i>Iterative Development Model</i> .....	69
4.2.2	Backend PilihGeh.....	70
<b>BAB 5</b>	<b>KESIMPULAN DAN SARAN .....</b>	<b>72</b>
5.1	Kesimpulan .....	72
5.2	Saran .....	72
	<b>DAFTAR PUSTAKA .....</b>	<b>74</b>
	<b>LAMPIRAN.....</b>	<b>77</b>
A.	Hasil Pengodean (Coding) Iterasi 1 .....	77
B.	Hasil Pengodean (Coding) Iterasi 2 .....	81

C.	Hasil Pengodean (Coding) Iterasi 3 .....	85
D.	Hasil Pengodean (Coding) Iterasi 4 .....	103
E.	Tampilan <i>Front End</i> PilihGeh.....	118
F.	Route .....	129

## DAFTAR GAMBAR

Gambar 2.1 Tampilan Awal Aplikasi Visual Studio Code .....	12
Gambar 2.2 Model Iteratif.....	13
Gambar 3.1 Model Iteratif.....	18
Gambar 3.2 Use Case Diagram Aktor Pengguna Sebelum Register .....	20
Gambar 3.3 Use Case Diagram Pengguna Setelah Login.....	21
Gambar 3.4 Use Case Diagram Pengguna Sebagai Admin Pemungutan Suara ...	22
Gambar 3.5 Use Case Diagram Pengguna Sebagai Pemilih .....	23
Gambar 3.6 Activity Diagram Daftar Akun.....	24
Gambar 3.7 Activity Diagram Login .....	25
Gambar 3.8 Activity Diagram Logout .....	26
Gambar 3.9 Activity Diagram Edit Profil .....	27
Gambar 3.10 Activity Diagram Menyelenggarakan Pemilihan.....	28
Gambar 3.11 Activity Diagram Pengguna Sebagai Admin Mengelola Kandidat	29
Gambar 3.12 Activity Diagram Pengguna Sebagai Admin Mengelola Timeline	30
Gambar 3.13 Activity Diagram Pengguna Sebagai Admin Mengelola Admin Pemilihan.....	31
Gambar 3.14 Activity Diagram Mendaftarkan Calon Pemilih .....	32
Gambar 3.15 Activity Diagram Pengguna Sebagai Admin Melihat Detail Pemungutan Suara.....	33
Gambar 3.16 Activity Diagram Pengguna Sebagai Admin Mengirim Ajakan Mengikuti Pemilihan .....	34

Gambar 3.17 Activity Diagram Pengguna Sebagai Admin Melakukan Verifikasi Suara.....	35
Gambar 3.18 Activity Diagram Pengguna Sebagai Admin Mengunduh Hasil Pemungutan Suara.....	36
Gambar 3.19 Activity Diagram Pengguna Sebagai Admin Mengubah Detail Pemilihan.....	37
Gambar 3.20 Activity Diagram Pengguna Sebagai Admin Melakukan <i>Emergency Stop Vote</i> .....	38
Gambar 3.21 Activity Diagram Pengguna Sebagai Admin Menghapus Seluruh Data Suara Masuk .....	39
Gambar 3.22 Activity Diagram Pengguna Sebagai Admin Menutup/Membuka Akses Pemungutan Suara .....	40
Gambar 3.23 Activity Diagram Pengguna Menerima Ajakan Mengikuti/Mengelola Pemungutan Suara.....	41
Gambar 3.24 Activity Diagram Pengguna Melihat Detail Pemungutan Suara.....	41
Gambar 3.25 Activity Diagram Pengguna Melaksanakan Pemungutan Suara.....	42
Gambar 3.26 Entity Relationship Diagram Database PilihGeh.....	43
Gambar A.1 Fungsi registerNoDevice .....	77
Gambar A.2 Fungsi loginNoDevice.....	78
Gambar A.3 Fungsi verifyOtp .....	79
Gambar A.4 Fungsi resendOTP .....	80
Gambar B.1 Fungsi getProfile.....	81
Gambar B.2 Fungsi editProfile (1).....	81
Gambar B.3 Fungsi editProfile (2).....	82
Gambar B.4 Fungsi verifyEditProfile .....	83
Gambar B.5 Fungsi createPemilihan.....	84
Gambar B.6 Fungsi getListKelolaPemilihan .....	84

Gambar B.7 Fungsi getPemilihanById .....	85
Gambar C.1 Fungsi getKandidat.....	85
Gambar C.2 Fungsi getKandidatByNomor.....	86
Gambar C.3 Fungsi addKandidat.....	86
Gambar C.4 Fungsi addEditPhotoKandidat.....	87
Gambar C.5 Fungsi editKandidat.....	88
Gambar C.6 Fungsi deleteKandidat.....	89
Gambar C.8 Fungsi getTimeline.....	90
Gambar C.9 Fungsi addTimeline.....	91
Gambar C.10 Fungsi updateTimeline.....	92
Gambar C.11 Fungsi deleteTimeline.....	93
Gambar C.12 Fungsi getListAdmin.....	93
Gambar C.13 Fungsi getAdminById.....	94
Gambar C.14 Fungsi addAdmin (1).....	94
Gambar C.15 Fungsi addAdmin (2).....	95
Gambar C.16 Fungsi addAdminExcel (1).....	95
Gambar C.17 Fungsi addAdminExcel (2).....	96
Gambar C.18 Fungsi addAdminExcel (3).....	97
Gambar C.19 Fungsi deleteAdmin.....	97
Gambar C.20 Fungsi getListUser.....	98
Gambar C.21 Fungsi getAlIListUser.....	98
Gambar C.22 Fungsi addUser.....	99
Gambar C.23 Fungsi addUserExcel (1).....	100
Gambar C.24 Fungsi addUserExcel (2).....	101
Gambar C.25 Fungsi sendTokenAllUser (1).....	101
Gambar C.26 Fungsi sendTokenAllUser (2).....	102

Gambar C.27 Fungsi sendTokenUserById .....	102
Gambar C.28 Fungsi deleteUser .....	103
Gambar D.1 Fungsi joinPemilihan.....	104
Gambar D.2 Fungsi joinAdminPemilihan.....	104
Gambar D.3 Fungsi countVote .....	105
Gambar D.4 Fungsi checkVoteAvailable .....	106
Gambar D.5 Fungsi addVote (1).....	107
Gambar D.6 Fungsi addVote (2).....	108
Gambar D.7 Fungsi verifyVoteUser .....	108
Gambar D.8 Fungsi resendTokenVerifyVote .....	109
Gambar D.9 Fungsi getVote .....	109
Gambar D.10 Fungsi getVotePending .....	110
Gambar D.11 Fungsi getVoteVerified .....	110
Gambar D.12 Fungsi getVoteInvalid .....	111
Gambar D.13 Fungsi changeTitle .....	111
Gambar D.14 Fungsi deleteAllVoteFromPemilihan.....	112
Gambar D.15 Fungsi setEmergencyStopVote .....	112
Gambar D.16 Fungsi setStatusToStop .....	113
Gambar D.17 Fungsi verifyVoteUser .....	114
Gambar D.18 Fungsi countVote .....	115
Gambar D.19 Fungsi downloadData (1) .....	116
Gambar D.20 Fungsi downloadData (2).....	117
Gambar D.21 Fungsi downloadData (3).....	118
Gambar E.1 Halaman Login .....	118
Gambar E.2 Halaman Daftar Akun.....	119
Gambar E.3 Halaman Verifikasi OTP .....	119

Gambar E.4 Halaman Profil.....	119
Gambar E.5 Halaman Menu Pemilihan .....	120
Gambar E.6 Halaman Menu Kelola Pemilihan.....	120
Gambar E.7 Form Pembuatan Pemilihan.....	121
Gambar E.8 Halaman Dashbord Kelola Pemilihan (Data Kandidat).....	121
Gambar E.9 Form Menambahkan Kandidat .....	122
Gambar E.10 Form Mengubah Data Kandidat .....	122
Gambar E.11 Form Upload Foto Kandidat .....	123
Gambar E.12 Halaman Real Time Count Admin .....	124
Gambar E.13 Halaman Timeline.....	125
Gambar E.14 Halaman Data Pemilih .....	125
Gambar E.15 Halaman Data Admin Pemilihan .....	126
Gambar E.16 Halaman Data Suara Masuk .....	126
Gambar E.17 Halaman Pengaturan Tambahan Pemilihan.....	127
Gambar E.18 Halaman Dashboard Pemilihan (Pemilih) .....	128
Gambar E.19 Halaman Pengambilan Foto.....	129
Gambar F.1 Route (1) .....	129
Gambar F.2 Route (2) .....	130

## DAFTAR TABEL

Tabel 3.1 Tabel Waktu Penelitian.....	16
Tabel 3.2 Alat.....	17
Table 3.1 Tabel Relasi Dengan Referensi Field Dari Tabel User.....	43
Table 3.2 Tabel Relasi dengan Referensi Field dari Tabel Pemilihan.....	44
Table 3.3 Tabel Relasi dengan Referensi Field Dari Tabel User_tempRegis.....	44
Table 4.1 Blackbox Testing Register .....	47
Table 4.2 Blackbox Testing Login' .....	48
Table 4.3 Blackbox Testing Verifikasi OTP .....	48
Table 4.4 Blackbox Testing Resend OTP .....	48
Table 4.5 Blackbox Testing Get Profile.....	50
Table 4.6 Blackbox Testing Edit Profile.....	50
Table 4.7 Blackbox Testing Verifikasi Edit Profil (Email/Nomor Handphone) ..	50
Table 4.8 Blackbox Testing Membuat Pemilihan.....	51
Table 4.9 Blackbox Testing Mendapatkan Daftar Pemilihan yang Dikelola.....	51
Table 4.10 Blackbox Testing Mendapatkan Detail Pemilihan.....	51
Table 4.11 Blackbox Testing Menambahkan Kandidat.....	55
Table 4.12 Blackbox Testing Menambahkan/Memperbarui Foto Kandidat.....	55
Table 4.13 Blackbox Testing Memperbarui Kandidat.....	56
Table 4.14 Blackbox Testing Menghapus Kandidat .....	56
Table 4.15 Blackbox Testing Mendapatkan Timeline Pemilihan.....	56
Table 4.16 Blackbox Testing Menambahkan Timeline .....	57

Table 4.17 Blackbox Testingg Memperbarui Data Timeline.....	57
Table 4.18 Blackbox Testingg Menghapus Timeline .....	57
Table 4.19 Blackbox Testing Mendapatkan Daftar Admin Pemilihan .....	57
Table 4.20 Blackbox Testing Mendapatkan Data Admin Berdasarkan ID.....	58
Table 4.21 Blackbox Testing Menambahkan Admin Pemilihan .....	58
Table 4.22 Blackbox Testing Menambahkan Admin Pemilihan Menggunakan File Excel.....	58
Table 4.23 Blackbox Testing Menghapus Admin.....	59
Table 4.24 Blackbox Testing Mendapatkan Daftar Pemilih Dengan Batas Data .	59
Table 4.25 Blackbox Testing Mendapatkan Data Seluruh Pemilih .....	59
Table 4.26 Blackbox Testing Menambahkan Pemilih .....	59
Table 4.27 Blackbox Testing Menambahkan Pemilih Menggunakan File Excel .	60
Table 4.28 Blackbox Testing Mengirim Ajakan Bergabung Pemilihan Ke Seluruh Pemilih .....	60
Table 4.29 Blackbox Testing Mengirim Ajakan Bergabung Pemilihan Ke Salah Satu Pemilih .....	60
Table 4.30 Blackbox Testing Menghapus Pemilih .....	61
Table 4.31 Blackbox Testing Pengguna Bergabung Ke Pemilihan Sebagai Pemilih .....	64
Table 4.32 Blackbox Testing Pengguna Bergabung Pemilihan Sebagai Admin ..	64
Table 4.33 Blackbox Testing Menghitung Total Suara Sebagai Pemilih .....	65
Table 4.34 Blackbox Testing Cek Ketersediaan Hak Pilih.....	65
Table 4.35 Blackbox Testing Pemilih Menggunakan Hak Pilih.....	66
Table 4.36 Blackbox Testing Pemilih Memverifikasi Suara .....	66
Table 4.37 Blackbox Testing Mengirim Ulang Pesan Verifikasi Suara Pemilih..	67
Table 4.38 Blackbox Testing Mendapatkan Seluruh Data Suara Masuk.....	67

Table 4.39 Blackbox Testing Mendapatkan Seluruh Data Suara Masuk Yang Belum Terverifikasi Admin .....	67
Table 4.40 Blackbox Testing Mendapatkan Seluruh Data Suara Masuk yang Sudah Terverifikasi Sah Oleh Admin .....	67
Table 4.41 Blackbox Testing Mendapatkan Seluruh Data Suara Masuk yang Sudah Terverifikasi Tidak Sah Oleh Admin .....	68
Table 4.42 Blackbox Testing Memperbarui Data Detail Pemilihan .....	68
Table 4.43 Blackbox Testing Menghapus Seluruh Data Suara Masuk.....	68
Table 4.44 Blackbox Testing Menghentikan Pemilihan Dalam Keadaan Darurat	68
Table 4.45 Blackbox Testing Menutup/Membuka Akses Pemilihan.....	68
Table 4.46 Blackbox Testing Admin Memverifikasi Suara Masuk.....	69
Table 4.47 Blackbox Testing Menghitung Total Suara Masuk .....	69
Table 4.48 Blackbox Testing Mengunduh Data Pemilihan .....	69

## **BAB 1 PENDAHULUAN**

### **1.1 Latar Belakang**

Saat ini pemungutan suara menjadi kegiatan yang penting dalam pengambilan keputusan yang demokratis dalam suatu organisasi ataupun instansi. Pemungutan suara banyak dilakukan dengan cara tradisional khususnya di Indonesia. Pemungutan suara dengan cara tradisional menggunakan kertas suara, kotak suara, dan pemeriksaan sekaligus perhitungan suara secara manual. Cara tradisional ini memiliki kelemahan seperti risiko kesalahan verifikasi suara, kesalahan saat menghitung suara, biaya dan tenaga yang diperlukan tidak sedikit, membutuhkan waktu yang lama untuk mempersiapkan maupun menyelenggarakan pemungutan suara, dan dapat terjadinya kecurangan seperti manipulasi suara dan pemilih ganda. Maka dibutuhkan suatu solusi yang dapat menghilangkan kelemahan tersebut dengan menggunakan teknologi yang tersedia saat ini.

Saat ini salah satu cara untuk menyelesaikan permasalahan dalam pemungutan suara tradisional adalah menyelenggarakan pemungutan suara secara *online* dengan menggunakan situs web sebagai media. Terdapat banyak keuntungan ketika pemungutan suara dilakukan secara *online* seperti biaya yang diperlukan lebih sedikit, mengurangi risiko kesalahan dalam perhitungan suara, dan terdapat banyak fitur yang mempermudah seperti perhitungan suara yang cepat, verifikasi suara lebih mudah, dan dapat melihat profil calon yang tersedia. Pemungutan suara secara *online* juga mempermudah peserta menggunakan hak pilih tanpa harus datang secara fisik.

Beberapa penelitian sudah pernah mengangkat kekurangan dari pemungutan suara tradisional dan membuat sistem pemungutan suara online sebagai penyelesaian masalahnya. Namun penelitian-penelitian tersebut memiliki kelemahan yang dapat

difokuskan menjadi beberapa poin yaitu masalah keamanan data, validitas suara, fitur yang minim, masalah fleksibel dan fokus dari sistem yang dibuat [1, 2, 3, 4, 5], dan rentan akan keamanan server.

Oleh karena itu diperlukan situs web yang dapat mengatasi permasalahan-permasalahan baru yang muncul pada pemungutan suara secara *online*. Situs web PilihGeh merupakan platform pemungutan suara yang dirancang untuk mengatasi permasalahan baru yang disebutkan sebelumnya.

## 1.2 Rumusan Masalah

Rumusan masalah dari rancang bangun situs web ini adalah bagaimana merancang dan membangun *Backend* situs web pemungutan suara menggunakan framework Express.js yang mendukung proses pemungutan suara secara online, menyediakan sistem yang mempermudah pemungutan suara, merancang dan implementasi API untuk berinteraksi dengan *frontend* PilihGeh, membuat *Backend* dapat mengelola pemilihan yang berbeda dalam waktu yang bersamaan, menjamin keamanan data dan kerahasiaan suara yang masuk, dan menerapkan metode *Iterative Development Model* pada proses rancang bangun *Backend* PilihGeh menggunakan framework Express.js.

## 1.3 Batasan Masalah

Batasan masalah pada tugas akhir ini sebagai berikut:

1. Rancang bangun situs web PilihGeh berfokuskan untuk penggunaan pada lingkungan masyarakat, lingkungan mahasiswa, dan lingkungan sekolah.
2. Spesifikasi hosting yang rendah (8GB memory, 8 vCPU dan 100GB Shared Disk dari Railway) membatasi performa backend website.

## 1.4 Tujuan

Tujuan dari rancang bangun *Backend* situs web ini adalah

1. Merancang dan membangun Backend yang dapat mendukung fungsionalitas situs web PilihGeh
2. Menyediakan sistem yang mempermudah pemungutan suara
3. Menyediakan OTP untuk verifikasi identitas pengguna
4. Membangun API untuk berinteraksi dengan frontend PilihGeh
5. Membangun Backend yang handal untuk mengelola banyak pemilihan dalam waktu yang bersamaan
6. Menjamin keamanan data dan kerahasiaan suara yang masuk

## 1.5 Sistematika Penulisan

Laporan ini dibagi menjadi beberapa bab untuk mempermudah dalam penguraian, di antaranya:

**BAB I** : PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan, dan sistematika penulisan skripsi.

**BAB II** : TINJAUAN PUSTAKA

Pada bab ini berisi tentang teori dasar dari Website, Rancang Bangun Website, Framework, NodeJs, ExpressJs, *Application Programming Interface*(API), REST, PostgreSQL, Sequelize, Visual Studio Code, *Unified Modelling Language*, *Iterative Development Model*.

**BAB III** : METODOLOGI PENELITIAN

Pada bab ini berisi tentang waktu jadwal dan tempat rancang bangun, penggunaan alat, dan penjelasan tahapan dalam pelaksanaan rancang bangun.

**BAB IV : HASIL DAN PEMBAHASAN**

Bab ini berisi pembahasan mengenai penjelasan alur pengerjaan selama menjalankan rancang bangun. Dan juga menjelaskan hasil dari pengerjaan yang sudah dikerjakan pada menjalankan rancang bangun.

**BAB V : KESIMPULAN DAN SARAN**

Bab ini memuat kesimpulan berdasarkan hasil pembahasan laporan skripsi dan saran dari penulis yang diharapkan dapat meningkatkan wawasan serta kemajuan bagi kita bersama.

**DAFTAR PUSTAKA** : Bab ini memuat daftar sumber kutipan teori - teori yang dijadikan acuan penulis dalam menulis laporan skripsi.

## **BAB 2 TINJAUAN PUSTAKA**

### **2.1 Website**

Website atau adalah sejumlah halaman web yang memiliki topik saling terkait, terkadang disertai pula dengan berkas-berkas gambar, video, atau berkas lainnya. Situs web merupakan kumpulan dari halaman web yang sudah dipublikasikan di jaringan internet dan memiliki domain atau URL (Unified Resource Locator) yang dapat diakses semua pengguna internet dengan cara mengetikkan alamatnya. [6]

Website adalah kumpulan dari halaman-halaman situs, yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam World Wide Web (www) di Internet. Sebuah web page adalah dokumen yang ditulis dalam format HTML (Hyper Text Markup Language), yang hampir selalu bisa diakses melalui HTTP. Semua publikasi dari website-website tersebut dapat membentuk sebuah jaringan informasi yang sangat besar. Web page layaknya sebuah buku yang dapat menampung berbagai informasi tentang banyak hal baik bersifat komersil maupun non komersil. Melalui media web inilah seseorang dapat memberikan informasi tertentu kepada orang lain yang berada diseluruh dunia. [7]

### **2.2 Rancang Bangun Website**

Rancang bangun merupakan salah satu hal yang penting dalam membuat program. Adapun tujuan dari perancangan adalah untuk memberi gambaran yang jelas serta lengkap kepada pemrogram dan ahli teknik yang terlibat. Perancangan harus berguna dan mudah dipahami sehingga dapat dengan mudah digunakan. Perancangan atau rancang merupakan serangkaian prosedur untuk menterjemahkan hasil analisa dan sebuah sistem ke dalam bahasa pemrograman untuk

mendeskripsikan dengan detail bagaimana komponen-komponen sistem tersebut dapat di implementasikan. [6]

Perancangan desain antarmuka (User Interface/UI) dan desain pengalaman pengguna (User Experience/UX) merupakan tahapan yang sangat penting dalam pengembangan sebuah perangkat lunak. Pembuatan desain ini melibatkan tahapan-tahapan seperti melakukan kegiatan untuk mengetahui kebutuhan pengguna dengan melakukan wawancara dan observasi kebutuhan pengguna, menggambarkan ide atau pandangan user menjadi dasar produk aplikasi yang akan dibuat. Membuat daftar kebutuhan pengguna. Selanjutnya menggambarkan solusi yang dibutuhkan dengan melakukan evaluasi bersama tim desain dan menggabungkan kreativitas dari masing-masing desainer aplikasi. Kemudian dilakukan pengembangan prototipe untuk mengimplementasikan ide yang sudah didapatkan dari tahapan sebelumnya menjadi sebuah aplikasi dan produk yang dapat diuji coba. [8]

Front-end mengacu pada setiap aspek dari proses desain dan lebih berfokus pada tampilan luarnya (layout). Tugas-tugas dari front-end adalah desain grafis dan pembuatan gambar, desain antarmuka, desain informasi dan lain-lainnya. [9]

Back-end adalah istilah halaman belakang, memungkinkan seseorang untuk masuk sebagai administrator dan melakukan perubahan informasi. Untuk memiliki tingkatan halaman/form yang aman, setidaknya halaman Back-end dilindungi dengan penggunaan password administrator. Dengan adanya halaman Back-end akan memudahkan seorang user untuk meng-update lainnya. [9]

### **2.3 Framework**

Framework adalah sekumpulan fungsi, class, dan aturan-aturan. Berbeda dengan library yang sifatnya untuk tujuan tertentu saja, Framework bersifat menyeluruh mengatur bagaimana membangun aplikasi. Framework memungkinkan membangun aplikasi dengan lebih cepat karena developer akan lebih memfokuskan pada pokok permasalahan. [10]

Sedangkan definisi lain dari Framework adalah kumpulan intruksi-intruksi yang dikumpulkan dalam class dan function-function dengan fungsi masing-masing untuk memudahkan developer dalam memanggilnya tanpa harus menuliskan syntax program yang sama berulang-ulang serta dapat menghemat waktu. [10]

Pada bagian perancangan *backend*, Terdapat banyak framework yang dapat digunakan seperti Spring, Django, Laravel, Flask, Express.js, Ruby on Rails, Next.js, dan lainnya.

### **2.3.1 NodeJs**

Node.js adalah sistem perangkat lunak yang didesain untuk pengembangan aplikasi web. Node.js dapat juga disebut sebagai *runtime environment*. Aplikasi ini ditulis dalam campuran Bahasa C++ dan juga JavaScript, mempunyai model event driven(basis event) dan asynchronous I/O. Tidak seperti kebanyakan bahasa JavaScript yang dijalankan pada web browser, Node.js dieksekusi sebagai aplikasi server. Node.js dapat berjalan di server karena dukungan dari V8 Engine buatan Google dan beberapa modul bawaan yang terintegrasi seperti modul http, modul filesystem, modul security dan beberapa modul penting lainnya [11].

Berbeda dengan bahasa pemrograman sisi server pada umumnya yang bersifat sinkronis atau blocking, Node.js bersifat asinkronis atau non-blocking sebagaimana halnya JavaScript bekerja. Node.js berjalan dengan basis event (event-driven). Maksud dari blocking secara sederhana adalah bahwa suatu kode program akan dijalankan hingga selesai, baru kemudian beralih ke kode program selanjutnya. [12]

Framework yang digunakan dalam rancang bangun sistem *Backend* ini yaitu Express.js dibangun di atas platform Node.js sehingga keduanya bekerja sama untuk membantu mempermudah pembuatan sistem.

#### **2.3.1.1 ExpressJs**

Express.js adalah sebuah framework Node.js yang membantu mempermudah pembuatan aplikasi berbasis Node.js dengan desain pattern yang mudah disesuaikan

serta fleksibel. ExpressJs juga merupakan framework ringan yang cocok dalam pembuatan aplikasi web dan API. [11]

ExpressJs merupakan framework web paling populer di NodeJs. Berkat dokumentasinya yang lengkap dan mudah digunakan, membuat pengembang mudah dalam memakai ExpressJs untuk mengembangkan berbagai produk seperti aplikasi web dan RESTful API. ExpressJs juga dapat menjadi dasar untuk membangun framework web yang lebih kompleks seperti SailsJs, MEAN(MongoDB, ExpressJs, AngularJs, NodeJs) dan MERN(MongoDB, ExpressJs, ReactJs, NodeJs). Express.js dibuat oleh TJ Holowaychuk dan sekarang dikelola oleh komunitas [12]

ExpressJs menyelesaikan masalah seperti mengatur sessions, mengorganisir route dengan rangkaian kondisi if berdasarkan jalur URL dan metode request HTTP, menentukan header respons yang tepat berdasarkan tipe data, dan banyak permasalahan lainnya. ExpressJs juga menyediakan struktur seperti MVC untuk aplikasi web. Aplikasi-aplikasi tersebut bisa bervariasi dari *back-end-only* REST APIs hingga *full-blown highly scalable full-stack*. Beberapa developer lebih familiar dengan Ruby dibandingkan dengan ExpressJs atau Sinatra yang memiliki perbedaan yang jauh dengan framework Ruby on Rails. [13]

## 2.4 Application Programming Interface (API)

Application Programming Interface (API) adalah suatu sistem yang mengintegrasikan dua bagian dari aplikasi. API terdiri dari beberapa elemen seperti fungsi, protocol, dan alat lainnya yang digunakan oleh pengembang untuk membuat sebuah aplikasi. Web API memiliki beberapa fitur yang mendukung operasi CRUD melalui protocol HTTP dengan menggunakan metode GET, POST, PUT dan DELETE. [14]

Dalam proses pembuatan Web API, beberapa aspek yang perlu diperhatikan adalah:

1. Menyediakan operasi CRUD (Create, Read, Update, dan Delete) yang dapat diakses melalui protocol HTTP dengan metode GET, POST, PUT, dan DELETE.
2. Memiliki response Accept Header dan HTTP status code.
3. Menghasilkan respon dalam format JSON, XML atau format lain yang diinginkan. Respon yang sering dipakai adalah dalam format JSON.
4. Web API dapat dijalankan di Apache atau web server lain yang didukung sesuai dengan bahasa pemrograman yang digunakan. [14]

Framework ExpressJs yang digunakan dalam rancang bangun sistem *Backend* ini merupakan framework yang mempermudah untuk membuat dan mengelola API.

#### 2.4.1 REST

REST (*Representational State Transfer*) merupakan gaya arsitektur dalam mendesain suatu web service. REST memiliki *resource* yang dapat diakses melalui sebuah alamat URL yang unik pada protocol HTTP. REST mempermudah klien untuk melakukan request melalui protocol HTTP dengan mudah menggunakan URL. Setiap alamat URL merujuk pada sekumpulan program yang akan dijalankan dan mengembalikan *response* kepada pengirim. [15]

REST menggunakan metode HTTP request untuk mengirimkan perintah kepada server. Mengacu pada beberapa penelitian, terdapat delapan metode HTTP request, yaitu GET, POST, PUT, DELETE, OPTIONS, HEAD, TRACE, dan CONNECT. Sedangkan pada REST API hanya digunakan empat metode yaitu GET, POST, PUT, dan DELETE. [15]

Pesan yang diterima dari server berupa kode HTTP berhasil atau gagal di dalam header dan isi pesan hasil pengolahan program itu sendiri. Berikut adalah kode-kode HTTP yang sering digunakan dalam menggunakan REST API [15]:

1. 200 OK : Perintah yang dikirim ke server benar dan berhasil dijalankan.
2. 400 Bad Request : Perintah yang dikirim ke server berisi isian yang salah.
3. 401 Unauthorized : Pengirim perintah mengirimkan kode kunci yang salah.

4. 403 Forbidden : Pengirim perintah tidak memiliki hak akses ke dalam resource yang dituju.
5. 404 Not Found : Resource yang dituju tidak ditemukan dalam server.
6. 429 Too Many Requests : Pengirim perintah mengakses mencapai/melebihi dari limit yang telah ditentukan dari batas waktu tertentu.
7. 500 Internal Server Error : Server atau potongan program dalam resource mengalami kesalahan.

#### **2.4.1.1 JSON**

JSON adalah sebuah format pesan balikan yang berukuran kecil yang mudah dibaca dan ditulis oleh manusia, dan juga mudah ditulis dan diurai oleh mesin. JSON merupakan salah satu pesan balikan yang dapat digunakan dalam sebuah REST API. JSON terbagi dalam dua struktur, yaitu [15]:

1. Gabungan name/value, atau yang biasa dikenal object atau record dalam bahasa pemrograman.
2. List value, atau yang biasa dikenal larik dalam bahasa pemrograman

### **2.5 Database**

*Database* adalah sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil dan dicari secara cepat. Secara konseptual *database* dapat diartikan sebagai ,sebuah koleksi atau kumpulan data yang saling berhubungan (relation), disusun menurut aturan tertentu secara logis, sehingga menghasilkan informasi. Kesimpulan dari pengertian diatas database adalah data yang sudah teratur agar mempermudah saat pencarian. [16]

#### **2.5.1 PostgreSQL**

PostgreSQL adalah sebuah Object-Relational Database Management System(ORDBMS) yang bersifat open source yang mendukung Structured Query

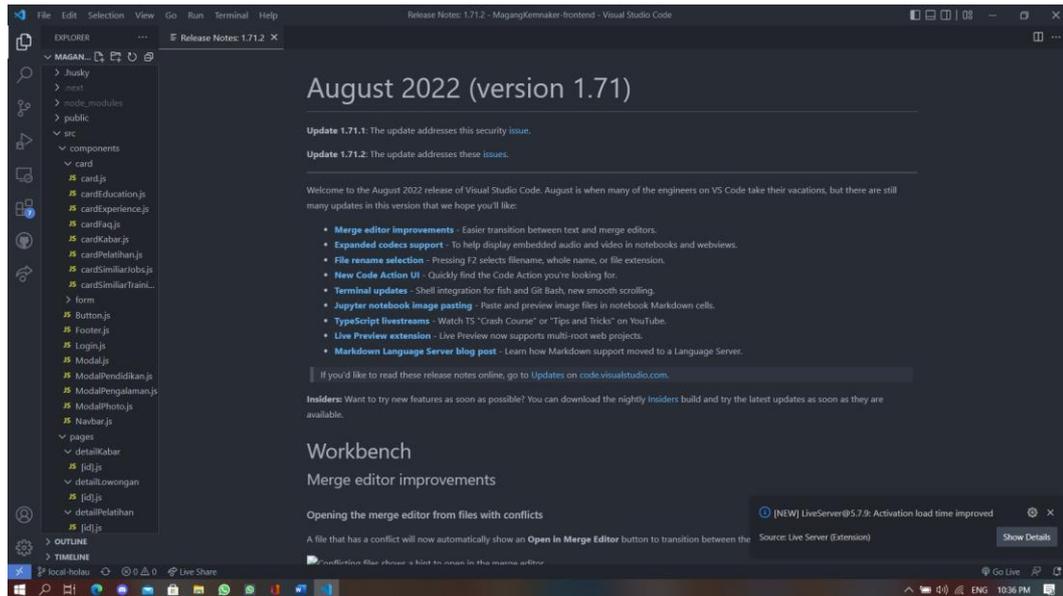
Language(SQL). Pada PostgreSQL terdapat interface untuk bahasa pemrograman lainnya seperti python, C, C++, dan lainnya[10]. PostgreSQL yang merupakan pelopor berbagai aplikasi RDBMS menjadi komersil. PostgreSQL memiliki lisensi GPL (General Public License) yang membuat PostgreSQL dapat digunakan oleh setiap orang tanpa perlu membayar lisensi baik untuk keperluan pribadi maupun komersil. [17]

### **2.5.1.1 Sequelize**

Sequelize adalah NodeJs promise-based ORM yang bisa digunakan untuk Postgres, MySQL, MariaDB, SQLite, Microsoft SQL Server, Oracle Database, Amazon Redshift dan Snowflake's Data Cloud. Sequelize menyediakan dukungan transaksi yang solid, relasi, *eager loading*, *lazy loading*, dan sebagainya. [18] Fungsi dari Sequelize adalah untuk bekerja dengan database dan relasi di dalamnya yang membuat deployment lebih mudah karena tidak perlu melakukan perubahan konteks saat menuliskan kode karena sudah terdapat interaksi menggunakan bahasa JavaScript yang sudah disediakan Sequelize melalui API.

## **2.6 Visual Studio Code**

Microsoft Visual Studio Code merupakan perangkat lunak dalam mengelola kode populer yang dikembangkan oleh Microsoft, bersifat open source dengan fitur-fitur terintegrasi seperti debugging tools, GIT tools, dan didukung oleh banyak perangkat lunak tambahan sebagai extension yang dapat disesuaikan dengan keinginan pengguna. [19]



Gambar 2.1 Tampilan Awal Aplikasi Visual Studio Code

## 2.7 Unified Modelling Language

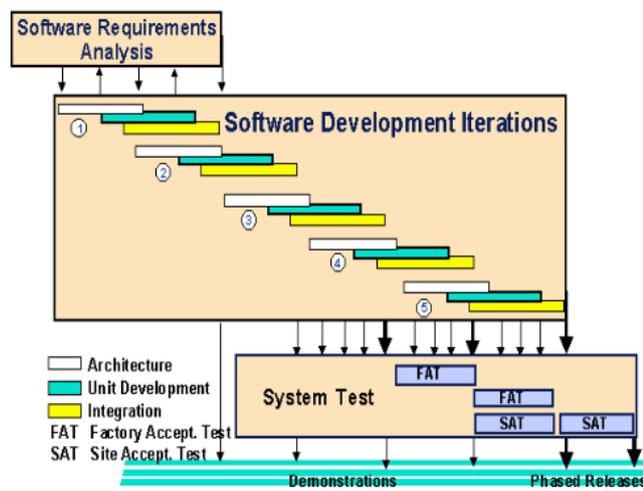
*Unified Modeling Language (UML)* merupakan sebuah metode pemodelan yang menggunakan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (Object-Oriented). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. [20]

UML sebagai sebuah bahasa yang memberikan vocabulary dan tatanan penulisan kata-kata dalam 'MS Word' untuk kegunaan komunikasi. Sebuah bahasa model adalah sebuah bahasa yang mempunyai vocabulary dan konsep tatanan / aturan penulisan serta secara fisik mempresentasikan dari sebuah sistem. Seperti halnya UML adalah sebuah bahasa standard untuk pengembangan sebuah software yang dapat menyampaikan bagaimana membuat dan membentuk model-model, tetapi tidak menyampaikan apa dan kapan model yang seharusnya dibuat yang merupakan salah satu proses implementasi pengembangan software. UML tidak hanya merupakan sebuah bahasa pemrograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti JAVA, C++,

Visual Basic, atau bahkan dihubungkan secara langsung ke dalam sebuah object-oriented database. Begitu juga mengenai pendokumentasian dapat dilakukan seperti; requirements, arsitektur, design, source code, project plan, tests, dan prototypes. Untuk dapat memahami UML membutuhkan bentuk konsep dari sebuah bahasa model, dan mempelajari 3 (tiga) elemen utama dari UML seperti building block, aturan-aturan yang menyatakan bagaimana building block diletakkan secara bersamaan, dan beberapa mekanisme umum. [20]

## 2.8 Iterative Development Model

Model iteratif mengatasi kelemahan dari model Waterfall. Berbeda dengan model waterfall yang persyaratannya hanya diperlukan sekali, persyaratan model iteratif dikumpulkan pada setiap fase [21]. Dengan pengembangan iteratif, proyek akan dibagi menjadi bagian-bagian kecil. Hal ini memungkinkan tim pengembang untuk menunjukkan hasil sebelumnya dan mendapatkan umpan balik dari pengguna sistem. Setiap iterasi merupakan proses mini-Waterfall dengan umpan balik dari satu fase yang memberikan informasi penting untuk desain tahap selanjutnya. Dengan model ini, produk perangkat lunak yang diproduksi di akhir setiap langkah bisa masuk ke produksi segera sebagai tambahan perilisan. [22]



Gambar 2.2 Model Iteratif

## 2.9 Penelitian Terdahulu

Beberapa penelitian sudah pernah membahas pemungutan suara secara *online*. Namun masing-masing penelitian memiliki kelemahan dan kelebihan. Penelitian-penelitian tersebut dijadikan referensi untuk mengembangkan situs web dan perbandingan untuk menghasilkan situs web yang diharapkan.

“**Aplikasi E-Voting untuk Pemilihan Kepala Desa Berbasis Website**” yang disusun oleh Abdul Azis, Ito Setiawan, dan Arif Risqiantoro memiliki kelemahan yaitu database tidak terenkripsi sehingga terdapat risiko keamanan data. Verifikasi untuk melakukan pemilihan menggunakan QR Code sekali pakai. Pada jurnal tidak mendetailkan sistem penggunaan QR Code ini dan detail suara yang digunakan, sehingga terdapat kemungkinan terjadinya *error* ketika QR Code di *scan* sehingga suara tidak dapat masuk ke sistem serta suara yang masuk dapat diwakilkan oleh rekan/kerabat pemilik suara. Pengembangan aplikasi E-Voting pada jurnal ini juga hanya berfokus pada pemilihan Kepala Desa pada Desa Tayem Timur. [1]

Jurnal berjudul “**Rancang Bangun Aplikasi E-Voting Berbasis Android (Studi Kasus: Pemilihan Ketua Organisasi di Lingkungan Fakultas Teknik Universitas Tanjungpura)**” juga mengangkat topik *E-Voting*. Namun sistem yang dirancang memiliki beberapa kelemahan seperti hanya menggunakan *login* akun sebagai verifikasi pemilik suara. Hal ini dapat menyebabkan suara dapat diwakilkan oleh rekan/kerabat pemilik suara sehingga secara aturan suara menjadi tidak sah namun admin tidak bisa membedakan dengan suara lainnya. Pengembangan aplikasi ini juga hanya berfokus pada satu organisasi. [2]

Jurnal yang berjudul “**Aplikasi Sistem e-Voting Ketua Umum UPN Band Veteran Jakarta Berbasis Website**” yang dituliskan oleh Anne Theresia Siahaan dan Helena Nurramdhani Irmada memiliki kelemahan di beberapa poin. Sistem ini melakukan verifikasi pemilik suara dengan cara *user* melakukan registrasi akun dengan mengisi data *user* yang dapat menunjukkan bahwa *user* merupakan anggota yang berhak ikut pemungutan suara. Hal ini menyebabkan suara memiliki kemungkinan diwakilkan oleh rekan/kerabat dari pemilik suara. Dalam jurnal ini

tidak tertulis keamanan enkripsi yang digunakan oleh sistem, yang kemudian dapat disimpulkan bahwa data pada sistem ini tidak terenkripsi. [3]

Skripsi dengan judul “**Perancangan Sistem Pemilihan Gubernur Mahasiswa Dengan Metode Electronic Voting Berbasis Website**” yang ditulis oleh R. Bima Gofiruli Maulida ini sudah memiliki verifikasi awal ketika masuk ke akun pemilih. Verifikasi dilakukan dengan cara mengirimkan token ke email yang terdaftar, pada kasus di jurnal ini, email yang terdaftar merupakan email universitas. Namun mengirimkan token ke email juga memiliki kemungkinan suara diwakilkan oleh rekan/kerabat dari pemilik suara. Sistem yang dirancang pada jurnal ini menggunakan bahasa pemrograman PHP. Sistem *E-voting* yang dirancang juga hanya berfokus ke satu organisasi saja. [4]

Jurnal yang berjudul “**Rancang Bangun Aplikasi E-Voting Ketua Organisasi Siswa Intra Sekolah (OSIS) Menggunakan Framework Laravel Berbasis Website**” ini hanya menggunakan token sebagai verifikasi pemilih. Hal ini juga memungkinkan suara diwakilkan oleh rekan/kerabat dari pemilik suara. Sistem ini juga hanya berfokus ke satu organisasi. Fitur yang disediakan oleh sistem ini user hanya sebatas melakukan pemilihan dan melihat visi misi, dan sisi admin fitur yang mendukung sistem sangat sedikit yaitu menambah data pemilih, menambah kandidat, menambah user, dan melihat hasil pemilihan. Fitur ini tentu bisa digunakan jika hanya berfokus pada satu organisasi. [5]

## BAB 3 METODOLOGI PENELITIAN

### 3.1 Waktu Jadwal dan Tempat Rancang Bangun

Tugas Akhir dilaksanakan dari Mei 2023 sampai dengan Oktober 2023 yang bertempat di Laboratorium Teknik Komputer Jurusan Teknik Elektro Universitas Lampung.

Tabel 3.1 Tabel Waktu Penelitian

No	Aktifitas	Mei	Juni	Juli	Agu	Sep	Okt
		2023					
1	Studi Literatur	■					
2	Analisis Kebutuhan	■					
3	Seminar Usul	■					
4	<i>Planning</i>	■	■	■			
5	Desain		■	■			
6	<i>Coding</i>		■	■	■	■	■
7	<i>Testing</i>		■	■	■	■	■
8	<i>Deployment</i>			■	■	■	■
9	Seminar Hasil						■
10	Perbaikan						■
11	Ujian Komprehensif						■

### 3.2 Alat dan Bahan

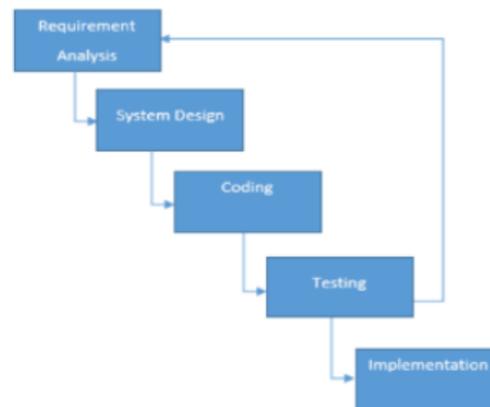
Perangkat keras dan lunak yang akan digunakan untuk rancang bangun situs web PilihGeh pada tugas akhir ini adalah sebagai berikut

Tabel 3.2 Alat

No	Perangkat	Spesifikasi	Kegunaan	Jumlah
1	Laptop	Intel Core i7- 7700HQ, RAM 16GB, SSD 1TB	Perangkat pembuat dan penguji sistem <i>Backend</i>	1
2	<i>Code Editor</i>	Visual Studio Code	<i>Software</i> yang mengeksekusi dan membangun situs web	1
3	<i>Framework</i>	ExpressJs	<i>Framework</i> untuk NodeJs yang membantu untuk pembuatan sistem <i>Backend</i>	1
4	<i>Database Editor</i>	PgAdmin	<i>Software</i> untuk membantu menampilkan database PostgreSQL	1
5	<i>API Tester</i>	Postman	<i>Software</i> untuk menguji API yang dibuat pada <i>Backend</i> .	1
6	<i>Deployment</i>	Railway	<i>Third Party Server</i> untuk <i>deployment</i> kode API	1

### 3.3 Tahapan Dalam Pelaksanaan Tugas Akhir

Metode penelitian yang diimplementasikan dalam rancang bangun sistem *Backend* ini adalah metode Iterative Development Model. Berikut visualisasi model iteratif:



Gambar 3.1 Model Iteratif

Pada model iteratif, iterasi akan dilakukan berulang kali dengan tujuan untuk mencapai hasil yang diinginkan dari fase sebelumnya. Pada model iteratif terdapat 5 tahapan yang menjadi inti dalam pengembangan, yaitu:

#### 3.3.1 Analisis

Kegiatan analisis dimulai dengan mengumpulkan informasi yang dibutuhkan dalam rancang bangun sistem *Backend* ini. Hal ini dilakukan agar mudah memahami konteks bisnis perangkat lunak serta mendapatkan pandangan umum terhadap output dan fungsi utama dari sistem. Tahap ini diperoleh informasi yang dibutuhkan untuk perancangan sistem *Backend*, yaitu kebutuhan fungsional dan non-fungsional dari situs web PilihGeh.

##### A. Kebutuhan Fungsional

- *Backend* dapat menyediakan fitur registrasi dan *login* ke akun
- *Backend* dapat mengirimkan OTP untuk autentikasi ke email/nomor Whatsapp

- *Backend* dapat memberikan akses kepada pengguna/pemilih untuk mengubah profil yaitu email, nama, nomor ponsel, NIM/NIK/NPM/NIS dan mengganti password
- *Backend* dapat mengelola proses pemungutan suara
- *Backend* dapat menyediakan fitur mendaftarkan kandidat
- *Backend* dapat menyediakan fitur mengelola timeline untuk mengatur waktu berjalannya pemungutan suara.
- *Backend* dapat menyediakan fitur mengelola pemilih.
- *Backend* dapat mengirimkan email maupun pesan Whatsapp berisi ajakan mengikuti/menjadi admin pemungutan suara
- *Backend* mendukung fitur menerima ajakan mengikuti/menjadi admin pemungutan suara
- *Backend* dapat mendukung pengiriman foto pengguna dan menyediakan data foto pemilih untuk verifikasi validitas suara yang masuk.
- *Backend* dapat mendukung sistem mengunduh hasil pemungutan suara
- *Backend* dapat mendukung sistem menghentikan pemungutan suara jika dalam keadaan darurat.
- *Backend* dapat mendukung sistem menghapus seluruh data suara yang masuk.
- *Backend* dapat mendukung sistem *takedown* pemungutan suara jika perlu.
- *Backend* dapat mendukung sistem yang dapat disesuaikan dan digunakan untuk berbagai organisasi secara bersamaan

## **B. Kebutuhan Non-Fungsional**

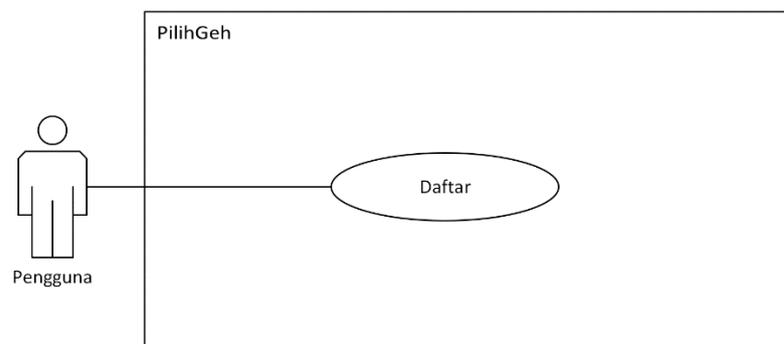
- *Backend* harus terhubung ke koneksi internet
- *Backend* dapat menjamin perlindungan data yang memadai dalam sisi kerahasiaan dan keamanan .
- *Backend* dapat memiliki skalabilitas yang tinggi.
- *Backend* dapat menahan serangan *spam*.
- *Backend* hanya dapat diakses oleh *Frontend* dari PilihGeh.

### 3.3.2 Desain Sistem

Kegiatan perencanaan dilanjutkan dengan pembuatan desain sistem situs web PilihGeh. Rancang bangun *Backend* PilihGeh di desain dengan menggunakan *Unified Modelling Language* yaitu *Use Case Diagram*, *Activity Diagram*, dan menggunakan *Entity Relationship Diagram*.

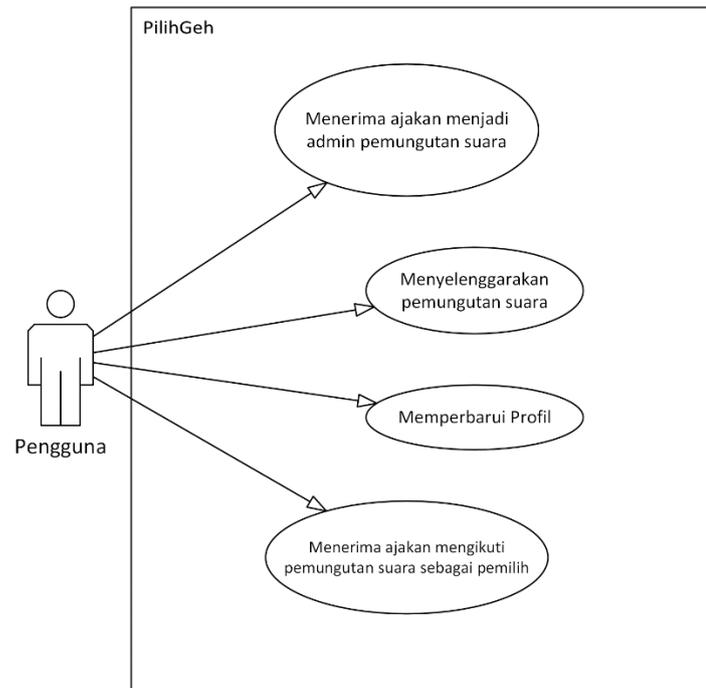
#### 3.3.2.1 Use Case Diagram

Use case diagram pada rancang bangun sistem ini dibagi menjadi 4 yaitu ketika user belum mendaftar, user belum login, user sesudah login dan user sebagai admin pemilihan.



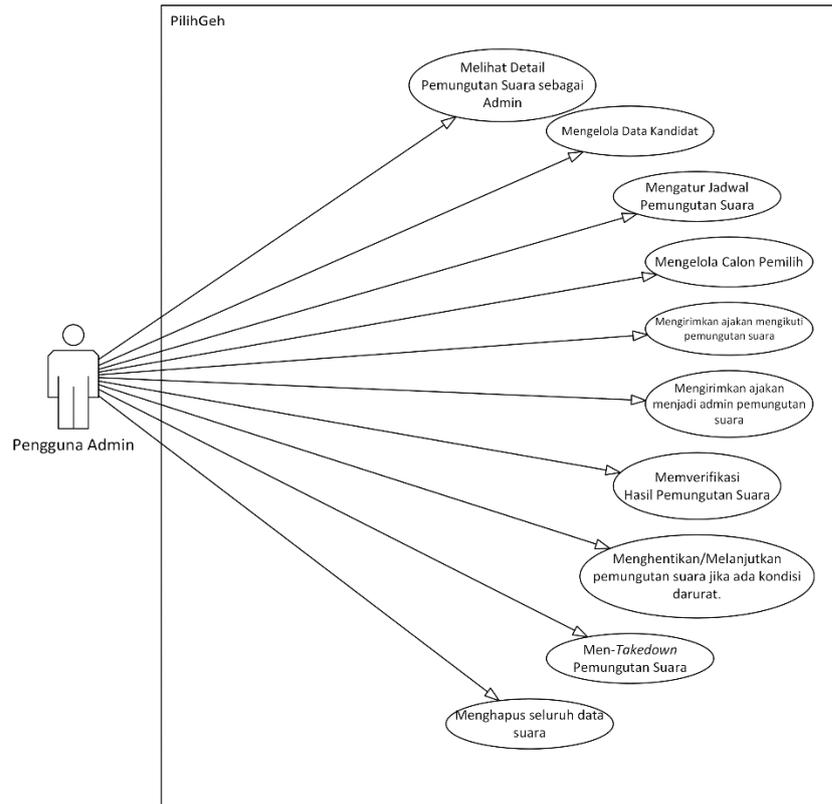
Gambar 3.2 Use Case Diagram Aktor Pengguna Sebelum Register

Pada use case diagram aktor pengguna sebelum register, dapat dilihat bahwa pengguna hanya mendapatkan akses untuk mendaftar dikarenakan untuk login, pengguna harus terdaftar terlebih dahulu.



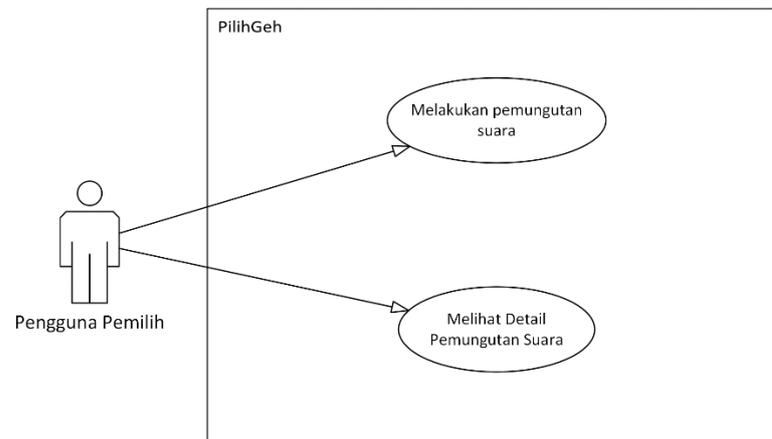
Gambar 3.3 Use Case Diagram Pengguna Setelah Login

Pada use case diagram setelah pengguna login, aktor pengguna mendapatkan akses ke fitur menyelenggarakan pemungutan suara, memperbarui profil, menerima ajakan menjadi admin pemungutan suara maupun menerima ajakan mengikuti pemungutan suara.



Gambar 3.4 Use Case Diagram Pengguna Sebagai Admin Pemungutan Suara

Pada use case diagram pengguna sebagai admin pemungutan suara, dapat dilihat pengguna dapat mengakses fitur melihat detail pemungutan suara, mengelola data kandidat, mengatur jadwal pemungutan suara, mengelola calon pemilih, mengirimkan ajakan mengikuti pemungutan suara, mengirimkan ajakan menjadi admin pemungutan suara, memverifikasi hasil pemungutan suara, dan menghentikan/melanjutkan pemungutan suara jika ada kondisi darurat. Namun ada fitur yang hanya bisa diakses oleh pengguna yang menyelenggarakan pemungutan suara yang disebut sebagai pemilik dari pemungutan suara, yaitu fitur untuk *men-takedown* pemungutan suara dan menghapus seluruh data suara.



Gambar 3.5 Use Case Diagram Pengguna Sebagai Pemilih

Pada use case diagram pengguna sebagai pemilih, terdapat fitur yang bisa diakses jika pengguna sudah menjadi pemilih dari pemungutan suara. Pengguna dapat melihat detail dari pemungutan suara seperti data kandidat, timeline, maupun perhitungan suara. Pengguna juga dapat melakukan pemilihan jika sudah memasuki waktu untuk memilih yang ditentukan dari timeline pemilihan.

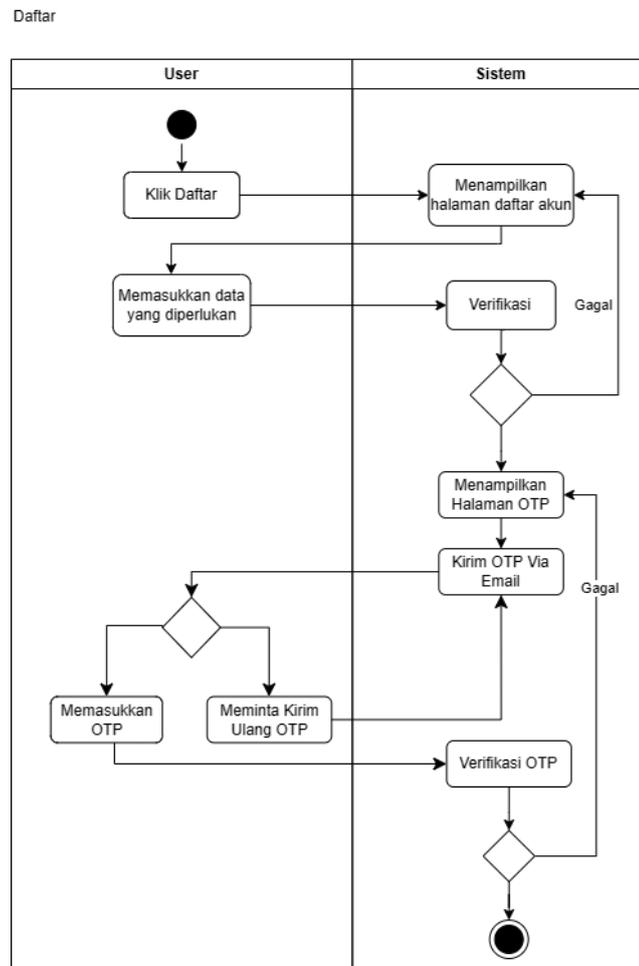
### 3.3.2.2 Activity Diagram

Activity diagram merupakan diagram yang menggambarkan alur kerja dari sebuah sistem. Diagram ini memudahkan untuk memahami proses pada sistem secara keseluruhan. Berikut activity diagram dari sistem pemilihan online PilihGeh.

#### 1. Activity Diagram Daftar Akun

Activity diagram daftar akun menjelaskan proses ketika pengguna ingin mendaftarkan akun baru. Pengguna mengklik tombol daftar yang kemudian sistem akan menampilkan halaman daftar akun. Lalu pengguna memasukkan data yang diperlukan untuk mendaftar akun. Lalu data tersebut akan diverifikasi oleh sistem. Jika verifikasi gagal, maka sistem akan menampilkan halaman daftar kembali, dan jika berhasil, maka sistem akan menampilkan halaman OTP serta mengirimkan OTP ke email pengguna. Jika pengguna meminta kirim ulang OTP, maka sistem akan mengirim ulang OTP ke email pengguna. Pengguna akan memasukkan OTP yang diterima, yang kemudian sistem akan memverifikasi OTP yang diisi oleh pengguna.

Jika gagal maka sistem akan kembali menampilkan halaman OTP, dan jika berhasil maka proses daftar akun akan selesai.

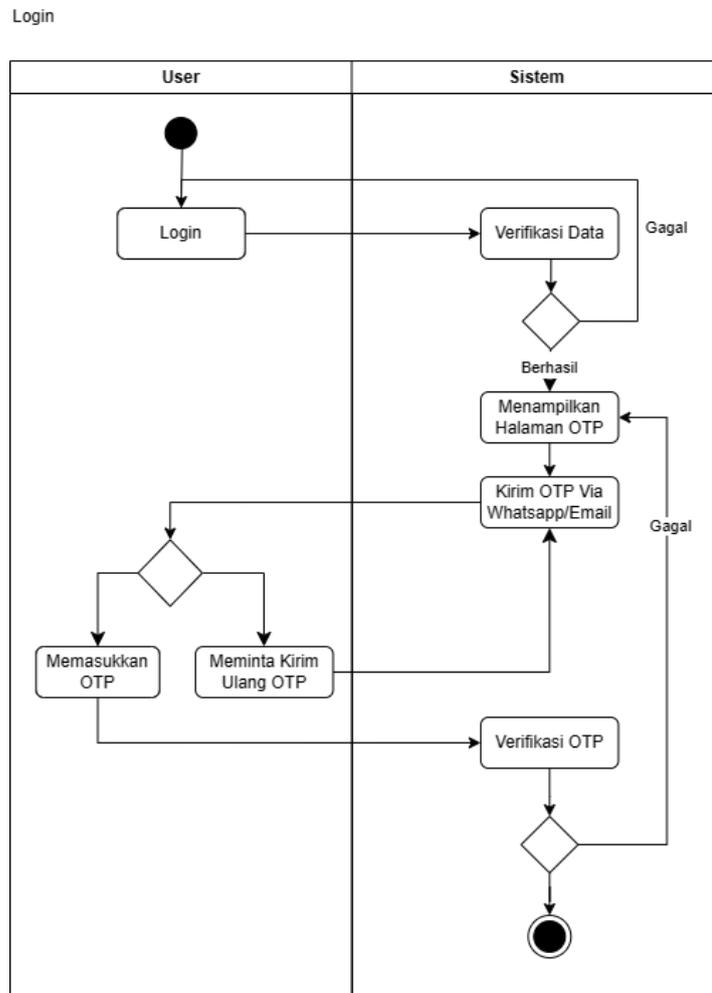


Gambar 3.6 Activity Diagram Daftar Akun

## 2. Activity Diagram Login

Activity diagram login menjelaskan langkah-langkah untuk login ke situs web PilihGeh. Proses login dimulai dengan pengguna membuka situs web dan mengisi form login dengan email/nomor handphone dan password. Kemudian, sistem memvalidasi data pengguna yang jika tidak valid maka sistem mengembalikan pengguna ke form login dan jika valid, maka sistem akan mengirimkan OTP ke Whatsapp/email pengguna dan menampilkan halaman OTP. Pengguna dapat meminta sistem untuk mengirim ulang OTP jika tidak menerima OTP. Setelah pengguna memasukkan OTP, sistem memverifikasi OTP dan menyelesaikan proses

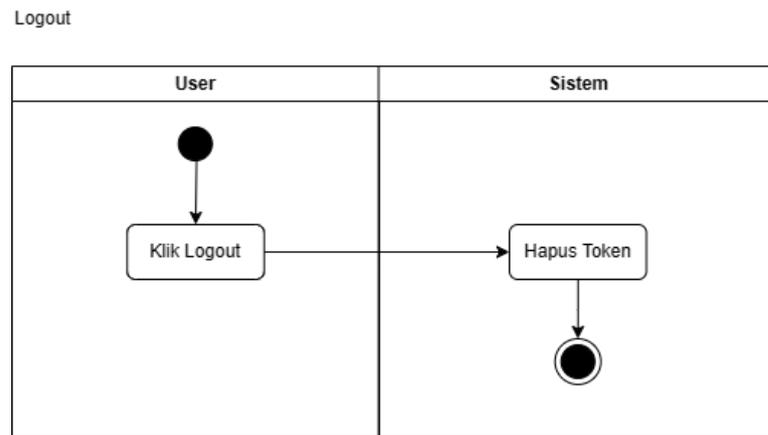
login jika OTP valid atau mengembalikan pengguna ke halaman OTP jika OTP tidak valid.



Gambar 3.7 Activity Diagram Login

### 3. Activity Diagram Logout

Activity diagram logout menjelaskan proses ketika pengguna ingin keluar dari akun. Proses logout akan dimulai jika pengguna mengklik tombol logout, maka sistem akan menghapus token login sehingga akun tidak dapat diakses kembali sampai pengguna kembali login.

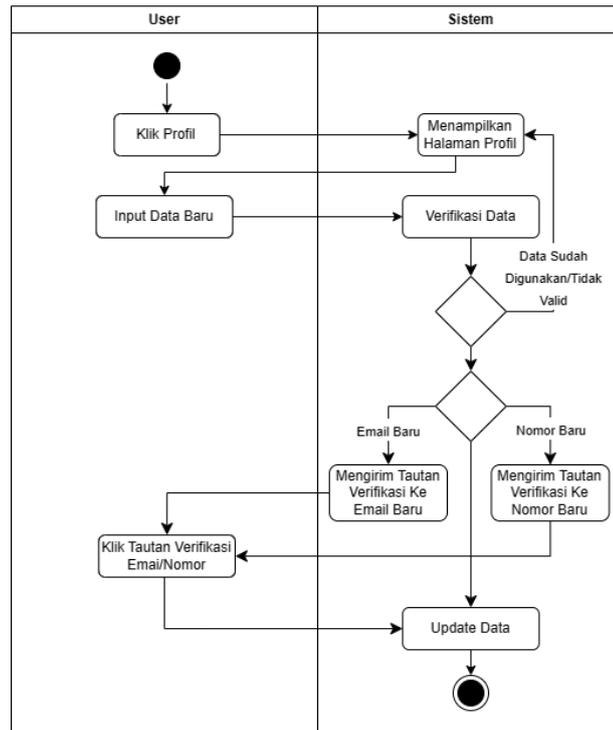


Gambar 3.8 Activity Diagram Logout

#### 4. Activity Diagram Edit Profil

Activity diagram edit profil menjelaskan alur ketika pengguna ingin meng-*edit* profil. Pengguna akan mengeklik profil yang kemudian akan menampilkan halaman profil. Lalu pengguna mengisi data baru ke dalam form. Sistem akan memverifikasi data yang diubah oleh pengguna. Jika gagal karena data yang dimasukkan tidak valid ataupun data sudah digunakan oleh pengguna lain, maka sistem akan mengembalikan ke langkah input data baru, dan jika data yang diberikan valid maka sistem akan mengecek apakah data email atau nomor pengguna berbeda dari sebelumnya (baru), jika email atau nomor pengguna baru, maka sistem akan mengirimkan tautan verifikasi untuk mengubah data email atau nomor pengguna. Jika pengguna tidak mengubah email atau nomor, maka sistem akan melakukan update data profil. Ketika pengguna mengeklik tautan verifikasi email ataupun nomor, maka sistem akan melakukan update ke data pengguna.

Edit Profil

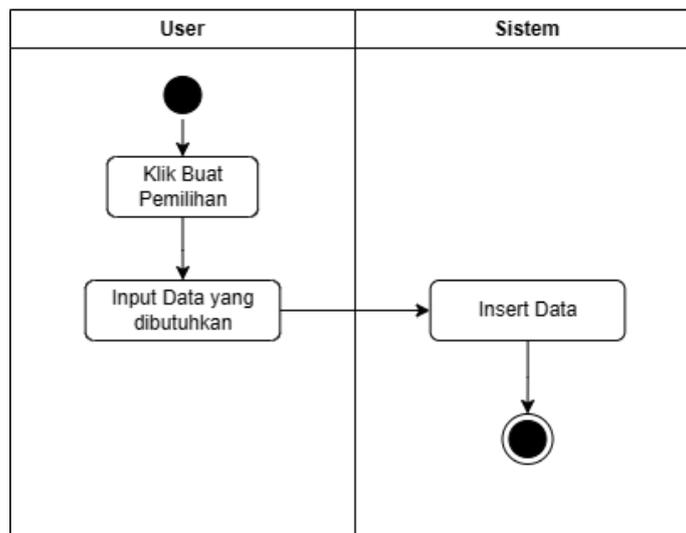


Gambar 3.9 Activity Diagram Edit Profil

## 5. Activity Diagram Menyelenggarakan Pemilihan

Activity diagram menyelenggarakan pemilihan suara menjelaskan alur ketika pengguna ingin membuat/menyelenggarakan pemilihan di PilihGeh. Pengguna akan mengklik tombol buat pemilihan lalu memasukkan data yang dibutuhkan. Setelah pengguna men-submit, maka sistem akan memasukkan data ke dalam database dan pemilihan sudah berhasil dibuat.

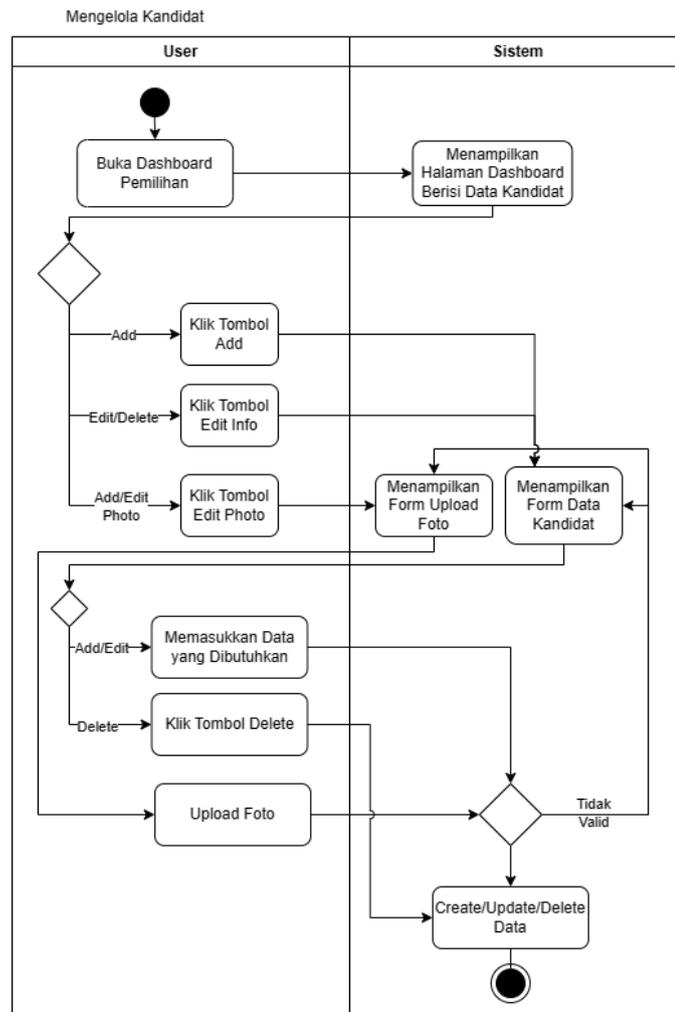
Menyelenggarakan Pemilihan



Gambar 3.10 Activity Diagram Menyelenggarakan Pemilihan

#### 6. Activity Diagram Pengguna Sebagai Admin Mengelola Kandidat

Activity diagram pengguna sebagai admin mengelola kandidat menjelaskan alur ketika pengguna sebagai admin dari pemungutan suara ingin mengelola kandidat seperti menambahkan, mengubah, maupun menghapus kandidat. Pengguna akan membuka dashboard pemilihan dan sistem akan menampilkan halaman dashboard yang berisikan data kandidat. Jika pengguna ingin menambahkan kandidat, maka pengguna akan mengeklik tombol *add*, jika pengguna ingin mengubah ataupun menghapus data kandidat, maka pengguna akan mengeklik tombol *edit info*. Lalu sistem akan menampilkan form data kandidat. Jika pengguna ingin menambahkan atau mengubah data, maka pengguna memasukkan data yang dibutuhkan yang kemudian akan divalidasi oleh sistem, jika tidak valid maka data ditolak dan menampilkan form data kandidat kembali. Jika pengguna ingin menghapus data kandidat, maka pengguna mengeklik tombol *delete* yang terdapat pada bagian bawah form data kemudian sistem akan memproses penghapusan data. Jika user ingin menambahkan atau mengubah foto kandidat, maka pengguna akan mengeklik tombol *edit photo* dan sistem akan menampilkan form upload photo. Lalu pengguna mengunggah foto dan sistem akan melakukan pengecekan file, jika tidak valid maka foto akan ditolak dan akan menampilkan form upload photo kembali, dan jika valid maka foto akan diproses untuk ditambahkan ke data kandidat.

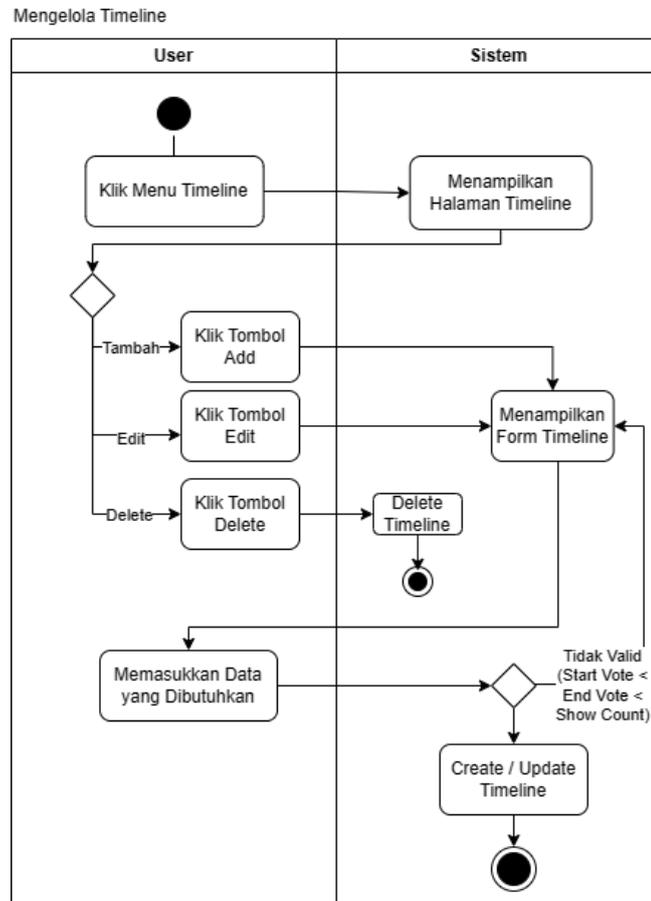


Gambar 3.11 Activity Diagram Pengguna Sebagai Admin Mengelola Kandidat

#### 7. Activity Diagram Pengguna Sebagai Admin Mengelola Timeline

Activity diagram pengguna sebagai admin mengelola timeline menjelaskan ketika pengguna sebagai admin dari pemungutan suara ingin mengelola timeline seperti menambahkan, mengubah, menghapus data timeline. Proses mengelola timeline dimulai dengan pengguna mengklik menu timeline dan sistem akan menampilkan halaman timeline. Jika pengguna ingin menambahkan timeline, maka pengguna akan klik tombol *add*, jika pengguna ingin mengubah timeline, maka pengguna akan klik tombol *edit*. Ketika mengklik tombol *add* ataupun *edit*, sistem akan menampilkan form timeline. Lalu pengguna akan memasukkan data yang dibutuhkan dan sistem akan melakukan validasi data yang diinput, jika tidak valid ( $\text{Start Vote} < \text{End Vote} < \text{Show Count}$ ) maka sistem akan menampilkan form timeline kembali. Jika valid maka sistem akan memproses data untuk

dimasukkan/diubah di database. Jika pengguna ingin menghapus timeline, maka pengguna akan mengklik tombol *delete* dan sistem akan memproses penghapusan data timeline.

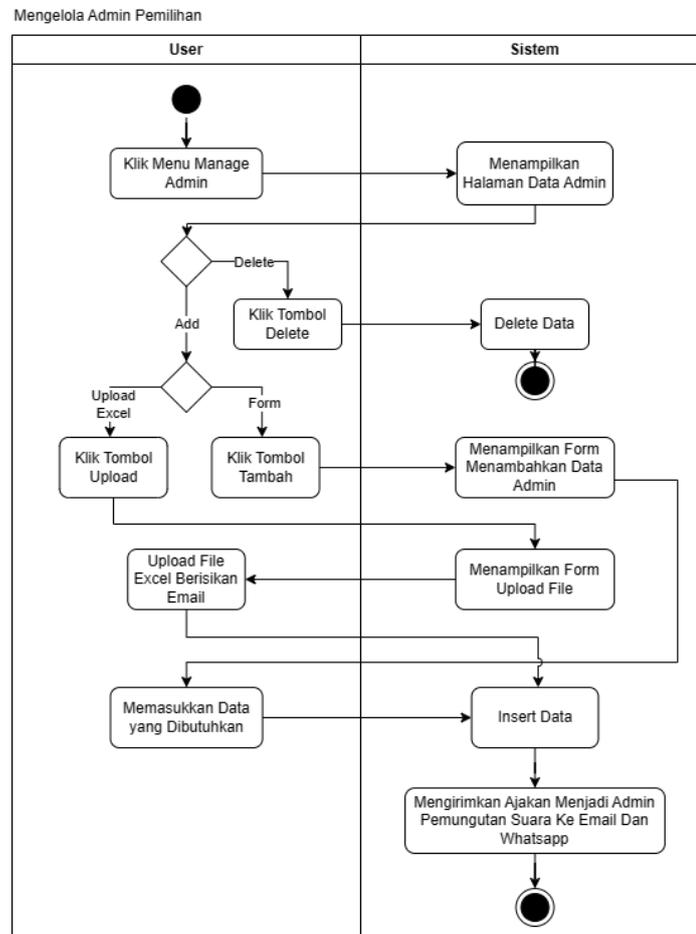


Gambar 3.12 Activity Diagram Pengguna Sebagai Admin Mengelola Timeline

#### 8. Activity Diagram Pengguna Sebagai Admin Mengelola Admin Pemilihan

Activity diagram pengguna sebagai admin mengelola admin pemilihan menjelaskan proses ketika pengguna sebagai admin pemungutan suara ingin mengelola admin pemilihan seperti menambahkan ataupun menghapus admin pemilihan. Proses ini dimulai dengan pengguna mengklik menu *manage admin* dan sistem menampilkan halaman data admin. Jika pengguna ingin menghapus admin, maka pengguna akan mengklik tombol delete dan sistem akan memproses penghapusan data admin. Jika pengguna ingin menambahkan admin, pengguna bisa menggunakan form ataupun upload excel, jika ingin menggunakan form, maka pengguna mengklik tombol tambah. Sistem akan menampilkan form

menambahkan data admin. Jika ingin menggunakan upload excel, maka pengguna mengklik tombol upload dan pengguna akan mengupload excel yang berisikan email dari admin yang ingin ditambahkan. Lalu sistem akan menambah data kedalam database dan mengirimkan tautan ajakan menjadi admin pemungutan suara ke email dan Whatsapp.

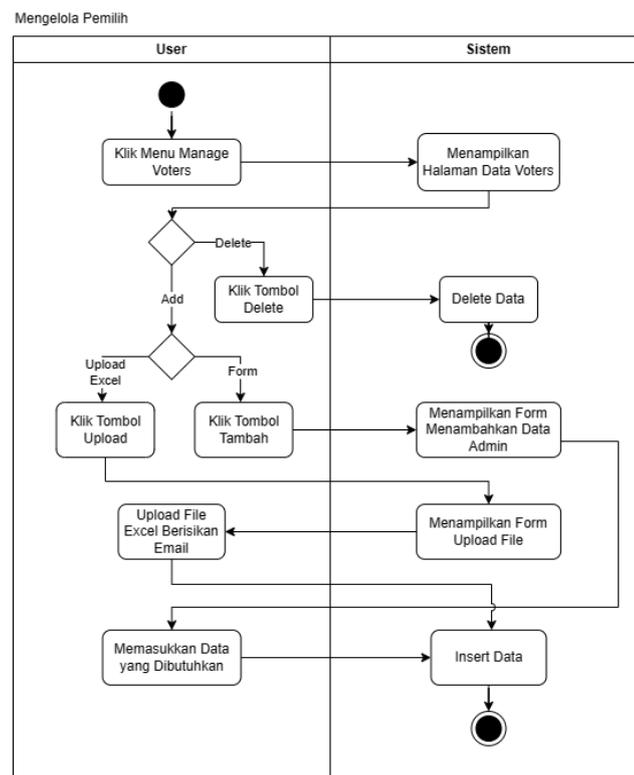


Gambar 3.13 Activity Diagram Pengguna Sebagai Admin Mengelola Admin Pemilihan

#### 9. Activity Diagram Pengguna Sebagai Admin Mengelola Pemilih

Activity diagram mendaftarkan calon pemilih menjelaskan alur ketika pengguna sebagai admin pemungutan suara ingin mengelola pemilih yang memiliki hak suara ke dalam pemilihan. Proses ini dimulai dengan pengguna mengklik menu *Manage Voters*. Sistem akan menampilkan halaman data pemilih ke pengguna. Jika pengguna ingin menghapus data pemilih, maka pengguna akan mengklik tombol delete di baris dimana data pemilih yang ingin dihapus lalu sistem akan memproses

penghapusan data pemilih. Untuk menambahkan pemilih, terdapat dua cara yaitu lewat form ataupun upload excel. Jika ingin menambahkan lewat form, maka pengguna akan mengklik tombol tambah dan sistem akan menampilkan form untuk memasukkan data email pemilih. Pengguna akan memasukkan data email dan sistem akan menambahkan email tersebut ke database. Jika ingin menambahkan lewat upload file excel, maka pengguna akan mengklik tombol upload dan sistem akan menampilkan form upload file dan pengguna mengupload file excel. Sistem akan otomatis membaca email yang terdapat pada file excel dan menambahkan email ke dalam database.

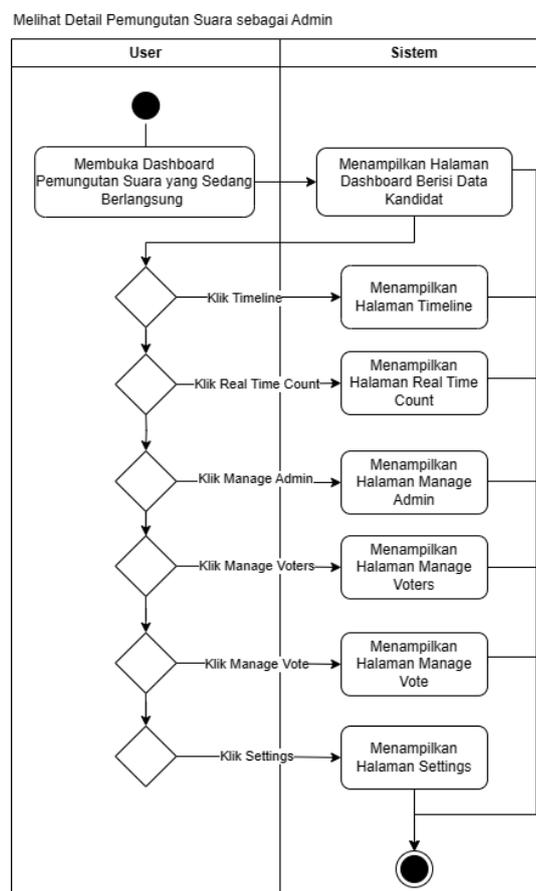


Gambar 3.14 Activity Diagram Mendaftarkan Calon Pemilih

#### 10. Activity Diagram Pengguna Sebagai Admin Melihat Detail Pemungutan Suara

Activity diagram pengguna sebagai admin melihat detail pemungutan suara menjelaskan proses ketika pengguna sebagai admin ingin melihat detail data yang terdapat pada pemungutan suara. Proses ini dimulai dengan pengguna membuka dashboard pemungutan suara dan sistem akan menampilkan halaman dashboard yang sudah berisikan data kandidat. Jika ingin melihat timeline, maka pengguna

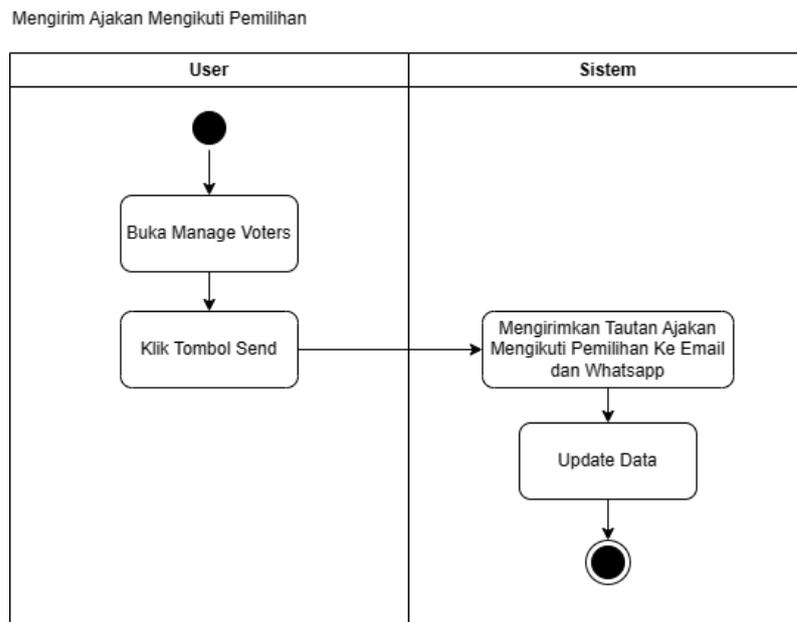
mengeklik menu timeline dan sistem akan menampilkan halaman timeline. Jika ingin melihat *real time count*, maka pengguna mengeklik menu *real time count* dan sistem akan menampilkan halaman *real time count*. Jika ingin melihat data admin, maka pengguna mengeklik menu *manage admin* dan sistem akan menampilkan halaman *manage admin*. Jika ingin melihat data pemilih, maka pengguna mengeklik menu *manage voters* dan sistem akan menampilkan halaman *manage voters*. Jika ingin melihat suara masuk, maka pengguna mengeklik menu *manage vote* dan sistem akan menampilkan halaman *manage vote*. Lalu jika ingin membuka *settings*, maka pengguna akan mengeklik menu *settings* dan sistem akan menampilkan halaman *settings*.



Gambar 3.15 Activity Diagram Pengguna Sebagai Admin Melihat Detail Pemungutan Suara

11. Activity Diagram Pengguna Sebagai Admin Mengirim Ajakan Mengikuti Pemilihan

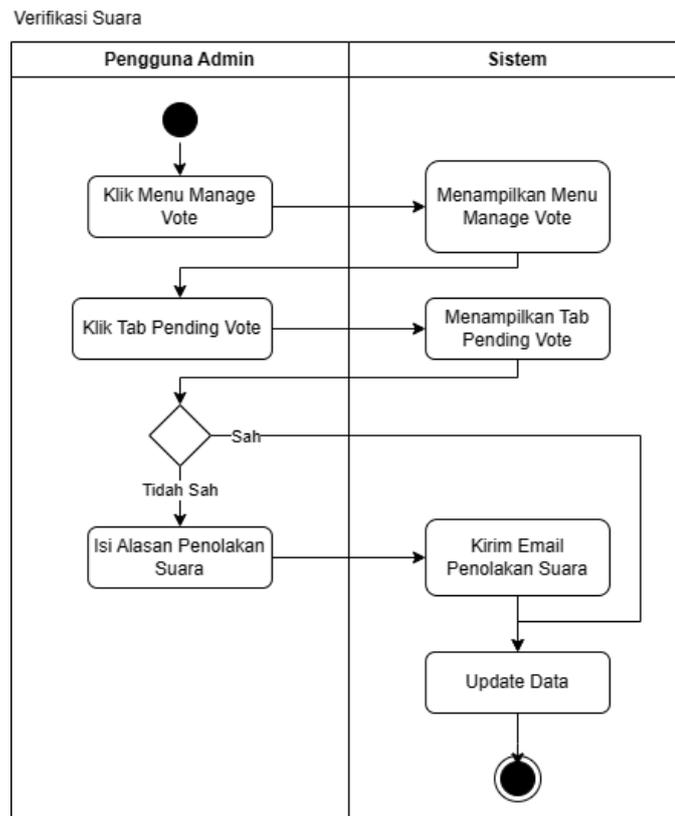
Activity diagram pengguna sebagai admin mengirim ajakan mengikuti pemilihan menjelaskan alur ketika pengguna sebagai admin ingin mengirimkan tautan ajakan ke pengguna lainnya sebagai pemilih. Proses ini dimulai dengan pengguna akan membuka *manage voters* dan kemudian mengeklik tombol *send* untuk mengirimkan email dan pesan whatsapp ajakan. Sistem akan mengirimkan tautan untuk menerima ajakan mengikuti pemilihan ke email maupun Whatsapp pengguna. Lalu sistem akan update data untuk keperluan klasifikasi.



Gambar 3.16 Activity Diagram Pengguna Sebagai Admin Mengirim Ajakan Mengikuti Pemilihan

## 12. Activity Diagram Pengguna Sebagai Admin Melakukan Verifikasi Suara

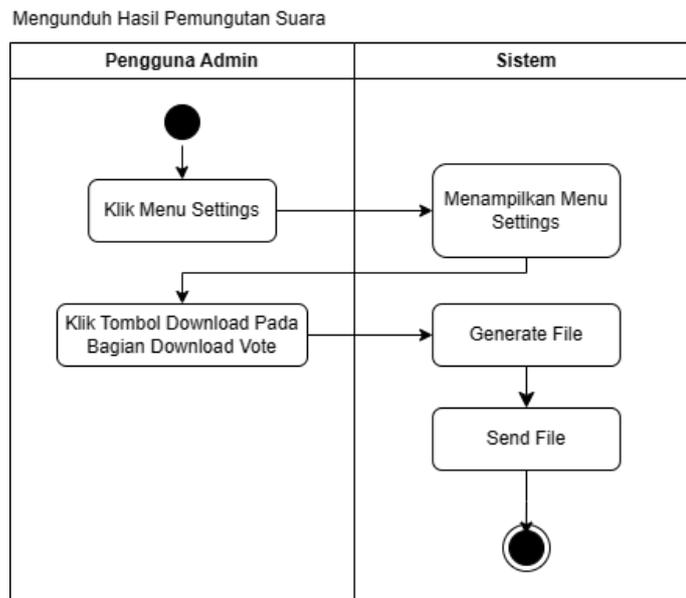
Activity diagram pengguna sebagai admin melakukan verifikasi suara menjelaskan alur ketika pengguna sebagai admin ingin melakukan verifikasi suara yang masuk. Diawal pengguna akan membuka menu *Manage Vote*. Sistem akan menampilkan menu *Manage Vote* dan pengguna akan menuju tab *pending vote* untuk melihat suara yang belum diverifikasi dan sistem akan menampilkan halaman tab *pending vote*. Jika pengguna mengesahkan suara, maka sistem akan melakukan perubahan data suara menjadi sah. Jika pengguna menolak suara (tidak sah), maka pengguna harus mengisi alasan penolakan suara dan sistem akan mengirimkan email berisikan alasan tersebut ke pengguna yang suaranya ditolak lalu sistem akan melakukan perubahan data suara menjadi tidak sah.



Gambar 3.17 Activity Diagram Pengguna Sebagai Admin Melakukan Verifikasi Suara

### 13. Activity Diagram Pengguna Sebagai Admin Mengunduh Hasil Pemungutan Suara

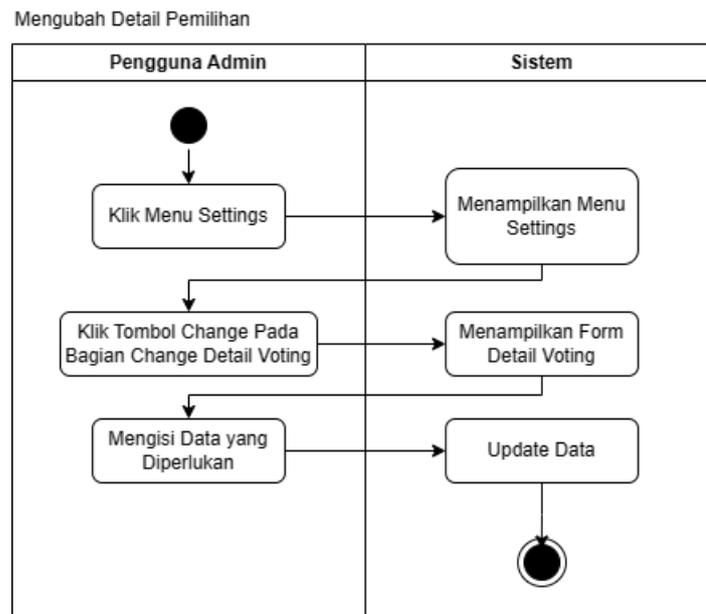
Activity diagram pengguna sebagai admin mengunduh hasil pemungutan suara ini menjelaskan ketika pengguna sebagai admin ingin mengunduh seluruh data yang terdapat pada pemungutan suara. Proses ini dimulai dengan pengguna membuka menu settings dan sistem akan menampilkan menu settings. Pengguna akan mengeklik tombol download pada bagian download vote dan sistem akan membuat file excel lalu mengirimkan ke browser pengguna.



Gambar 3.18 Activity Diagram Pengguna Sebagai Admin Mengunduh Hasil Pemungutan Suara

#### 14. Activity Diagram Pengguna Sebagai Admin Mengubah Detail Pemilihan

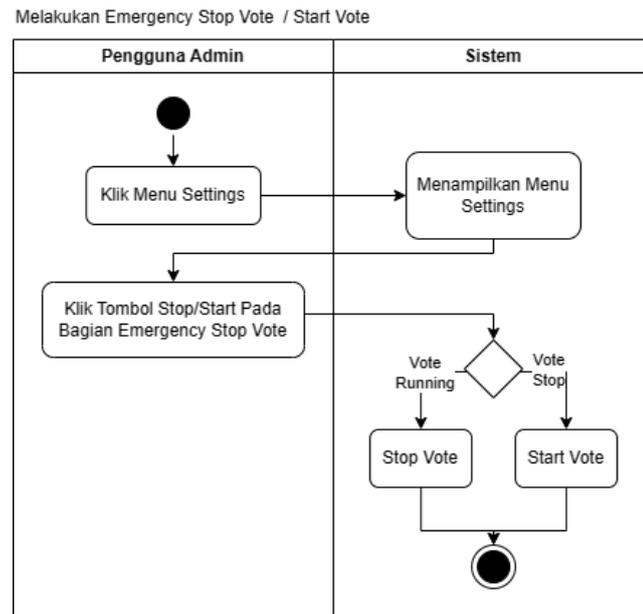
Activity diagram pengguna sebagai admin mengubah detail pemilihan ini menjelaskan ketika pengguna sebagai admin ingin mengubah judul, nama organisasi maupun deskripsi detail dari pemungutan suara. Proses ini dimulai dengan pengguna membuka menu settings dan sistem akan menampilkan menu settings. Pengguna akan mengklik tombol *change* pada bagian *Change Detail Voting*. Lalu sistem akan menampilkan form *detail voting*. Pengguna akan mengisi data yang perlu diubah dan sistem akan mengubah data sesuai dengan input pengguna.



Gambar 3.19 Activity Diagram Pengguna Sebagai Admin Mengubah Detail Pemilihan

#### 15. Activity Diagram Pengguna Sebagai Admin Melakukan Emergency Stop Vote

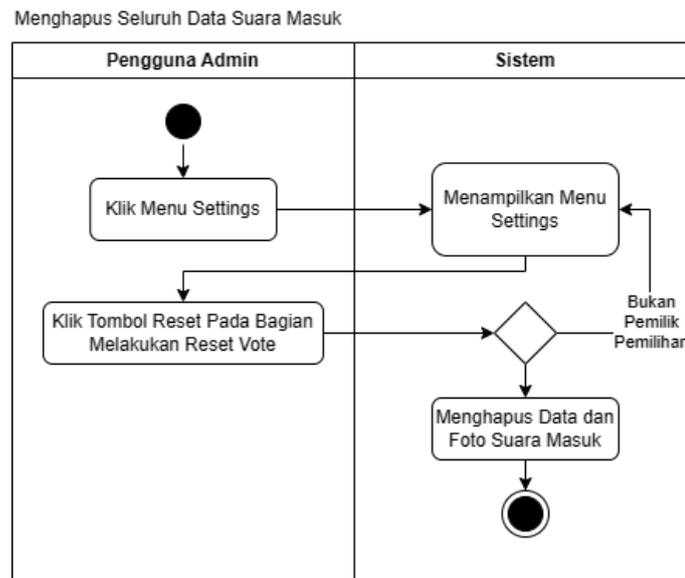
Activity diagram pengguna sebagai admin melakukan *emergency stop vote* menjelaskan ketika pengguna sebagai admin ingin proses pemungutan suara dihentikan sementara dikarenakan kondisi khusus ataupun memulai kembali. Proses ini dimulai dengan pengguna membuka menu *settings* dan sistem akan menampilkan menu *settings*. Pengguna akan mengklik tombol stop/start pada bagian *Emergency Stop Vote*. Lalu sistem akan melakukan update jika pemungutan suara sedang berjalan, maka sistem akan menghentikan pemungutan suara, jika pemungutan suara sedang berhenti, maka sistem akan memulai kembali pemungutan suara.



Gambar 3.20 Activity Diagram Pengguna Sebagai Admin Melakukan *Emergency Stop Vote*

16. Activity Diagram Pengguna Sebagai Admin Menghapus Seluruh Data Suara Masuk

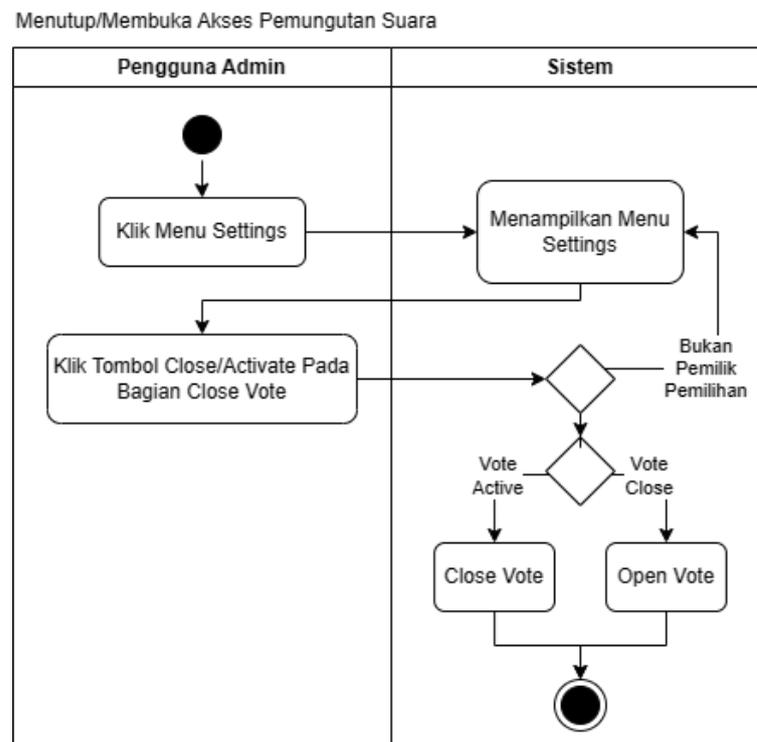
Activity diagram pengguna sebagai admin menghapus seluruh data suara masuk menjelaskan ketika pengguna sebagai admin ingin menghapus seluruh data suara masuk. Proses ini dimulai dengan pengguna membuka menu *settings* dan sistem akan menampilkan menu *settings*. Pengguna akan mengklik tombol reset pada bagian *Reset Vote*. Lalu sistem akan melakukan verifikasi jika pengguna bukan pemilik dari pemungutan suara, maka *request* akan ditolak dan kembali menampilkan menu *settings*. Jika pengguna adalah pemilik pemungutan suara, maka sistem akan menghapus data dan foto dari database.



Gambar 3.21 Activity Diagram Pengguna Sebagai Admin Menghapus Seluruh Data Suara Masuk

#### 17. Activity Diagram Pengguna Sebagai Admin Menutup/Membuka Akses Pemungutan Suara

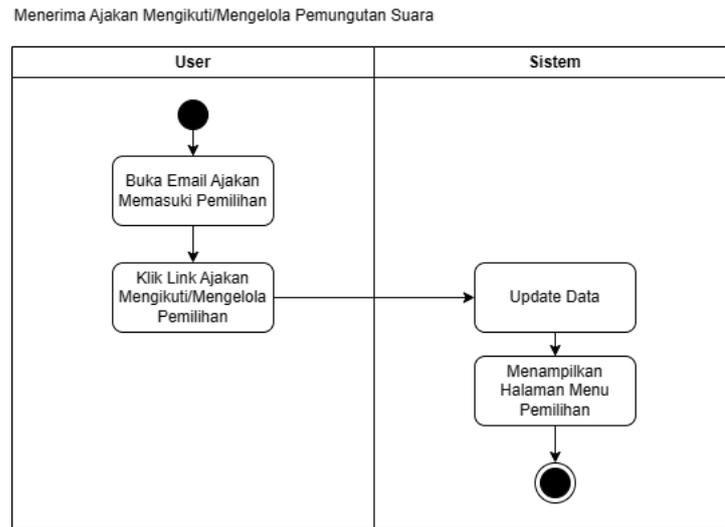
Activity diagram pengguna sebagai admin menutup/membuka akses pemungutan suara ini menjelaskan ketika pengguna sebagai admin ingin menutup/membuka akses pemilih ke pemungutan suara. Proses ini dimulai dengan pengguna membuka menu *settings* dan sistem akan menampilkan menu *settings*. Lalu pengguna akan mengklik tombol *close/activate* pada bagian *Close Vote*. Jika pengguna bukan pemilik pemungutan suara, maka *request* akan ditolak dan kembali ke menu *settings*. Jika pengguna adalah pemilih pemungutan suara, maka sistem akan memproses *request* dan jika pemilihan sedang berjalan, maka sistem akan menutup akses, jika pemilihan sudah ditutup akses, maka sistem akan membuka akses ke pemungutan suara.



Gambar 3.22 Activity Diagram Pengguna Sebagai Admin Menutup/Membuka Akses Pemungutan Suara

#### 18. Activity Diagram Pengguna Menerima Ajakan Mengikuti/Mengelola Pemungutan Suara

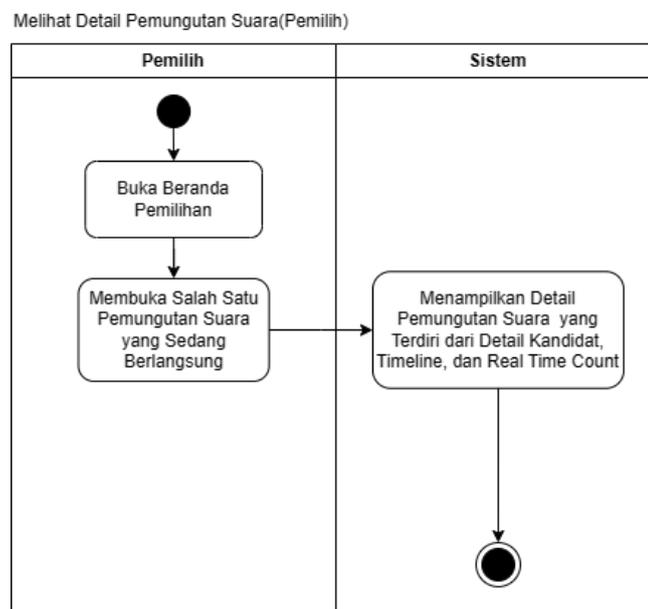
Activity diagram pengguna menerima ajakan mengikuti/mengelola pemungutan suara menjelaskan ketika pengguna ingin menerima ajakan mengikuti/mengelola pemungutan suara dengan mengklik tautan. Proses ini dimulai dengan pengguna membuka email yang berisikan tautan untuk mengikuti/mengelola pemungutan suara. Lalu pengguna mengklik tautan tersebut dan sistem akan melakukan update data sehingga pengguna dapat mengakses pemilihan dan sistem akan menampilkan menu pemilihan.



Gambar 3.23 Activity Diagram Pengguna Menerima Ajakan Mengikuti/Mengelola Pemungutan Suara

#### 19. Activity Diagram Pengguna Melihat Detail Pemungutan Suara

Activity diagram pengguna melihat detail pemungutan suara menjelaskan proses pengguna melihat detail yang disediakan pada pemungutan suara. Proses ini dimulai dengan pengguna membuka beranda pemilihan lalu memilih salah satu pemungutan suara yang sedang berlangsung. Sistem akan menampilkan detail data dari pemungutan suara, terdiri dari detail kandidat, timeline dan *real time count*.

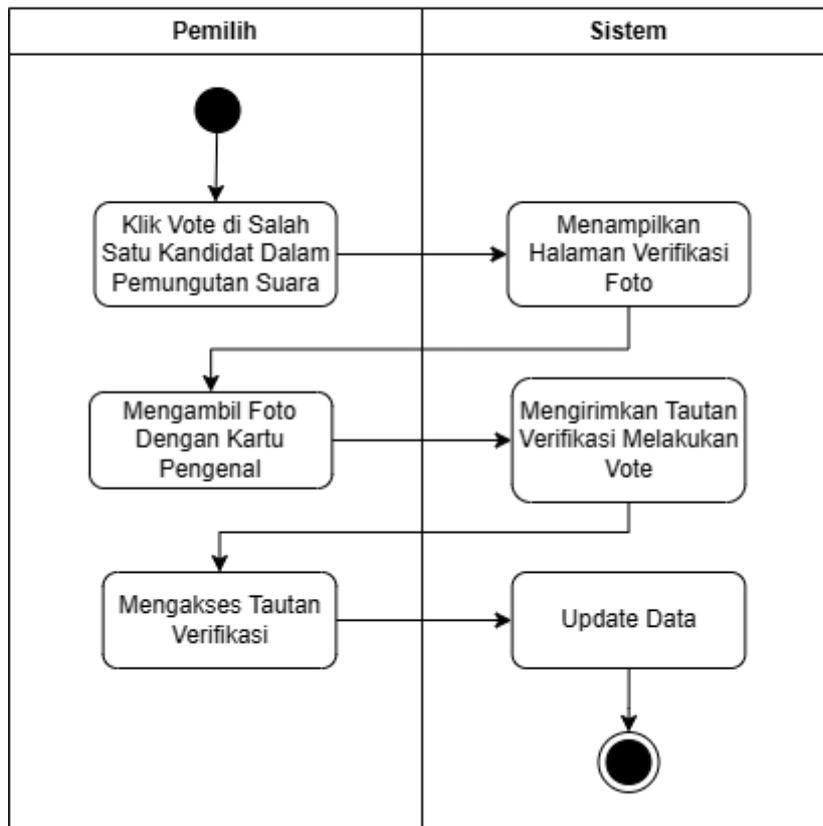


Gambar 3.24 Activity Diagram Pengguna Melihat Detail Pemungutan Suara

## 20. Activity Diagram Pengguna Melaksanakan Pemungutan Suara

Activity diagram pengguna melaksanakan pemungutan suara menjelaskan proses ketika pengguna ingin memakai hak suaranya dalam pemungutan suara. Proses ini dimulai dengan pengguna mengeklik *vote* pada salah satu kandidat dalam pemungutan suara dan sistem akan menampilkan halaman verifikasi foto. Pengguna akan mengambil foto dengan kartu pengenalan yang sesuai dengan data pada profil. Sistem akan mengirimkan tautan verifikasi melakukan vote dan pengguna akan mengakses tautan verifikasi yang dikirim lewat email ataupun Whatsapp. Sistem akan mengupdate data suara sehingga suara sudah terverifikasi oleh pengguna.

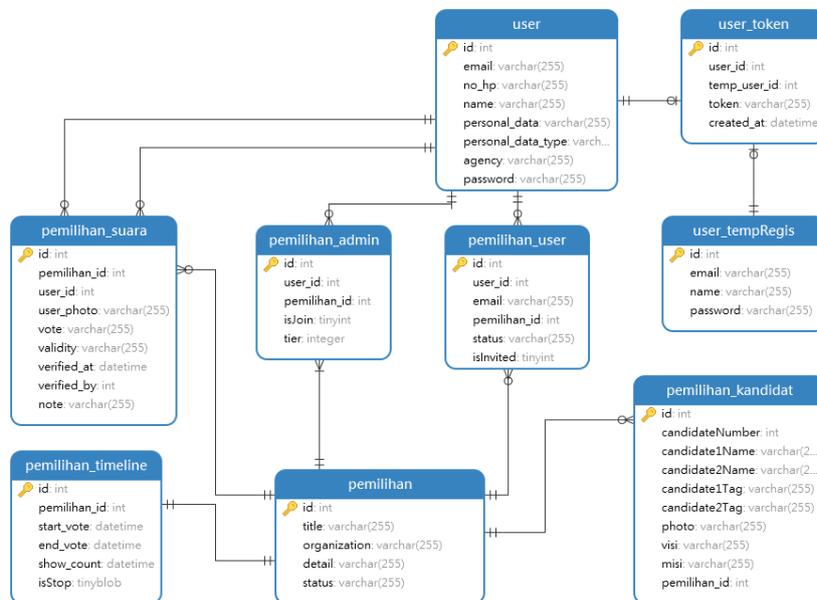
Melaksanakan Pemungutan Suara



Gambar 3.25 Activity Diagram Pengguna Melaksanakan Pemungutan Suara

### 3.3.2.3 Entity Relationship Diagram

Entity relationship diagram (ERD) merupakan diagram yang menggambarkan hubungan antar entitas dari suatu database. ERD terdiri dari 3 komponen yaitu entitas, atribut dan relasi.



Gambar 3.26 Entity Relationship Diagram Database PiliGeh

ERD pada database PiliGeh berpusat pada dua entitas, yaitu user(pengguna) dan pemilihan dan satu entitas sementara yaitu user\_tempRegis. Berikut jenis relasi dari tiap entitas ke entitas lainnya.

#### 1. Entitas User

Table 3.1 Tabel Relasi Dengan Referensi Field Dari Tabel User

Field	Table	Cardinality	Cardinality Source	Relation
User_id	User_token	Zero or One	One and only one	1:1
User_id	Pemilihan_admin	Zero or Many		1:M
User_id	Pemilihan_user	Zero or Many		1:M
User_id	Pemilihan_suara	Zero or Many		1:M
User_id	Pemilihan_suara	Zero or Many		1:M

## 2. Entitas Pemilihan

Table 3.2 Tabel Relasi dengan Referensi Field dari Tabel Pemilihan

Field	Table	Cardinality	Cardinality Source	Relation
Pemilihan_id	Pemilihan_timeline	One and only one	One and only one	1:1
Pemilihan_id	Pemilihan_suara	Zero or Many		1:1
Pemilihan_id	Pemilihan_admin	One or Many		1:M
Pemilihan_id	Pemilihan_user	Zero or Many		1:M
Pemilihan_id	Pemilihan_kandidat	Zero or Many		1:M

## 3. Entitas User\_tempRegis

Table 3.3 Tabel Relasi dengan Referensi Field Dari Tabel User\_tempRegis

Field	Table	Cardinality	Cardinality Source	Relation
temp_user_id	User_token	Zero and One	One and only one	1:1

### 3.3.3 Pengodean

Kegiatan perencanaan setelah desain kerangka sistem *Backend*, akan dilanjutkan dengan pembuatan serangkaian tes yang akan dijalankan sistem. Serangkaian tes yang dibuat akan menjadi acuan untuk pembuatan kode untuk melalui tes dengan sukses. Pembuatan rangkaian tes akan menggunakan aplikasi Postman. Sedangkan pembuatan kode akan menggunakan framework ExpressJs yang berbasis NodeJs dengan *software code editor* yaitu Visual Studio Code untuk melakukan rancang bangun sistem *Backend* website PilihGeh.

### 3.3.4 Pengujian

Proses testing dilakukan setelah proses kode telah selesai. Testing berfungsi untuk mengetahui kesalahan yang akan timbul saat sistem berjalan serta untuk memastikan bahwa kode telah memenuhi kebutuhan sistem yang telah didefinisikan

sebelumnya. Dalam proses testing ini, akan difokuskan ke bagian keamanan serta kinerja dari sistem *Backend* tanpa mengurangi kebutuhan sistem.

Pada rancang bangun situs web ini, dilakukan testing menggunakan aplikasi Postman untuk mengecek apakah API berjalan dengan semestinya. Testing juga akan dilakukan dengan Postman untuk melakukan tes terhadap berbagai keadaan yang akan dihadapi.

### **3.3.5 Implementasi**

Proses implementasi dilakukan setelah proses testing selesai. Proses ini dilakukan dengan melakukan *deployment* ke layanan pihak ketiga. Pihak ketiga yang disebutkan akan ditentukan menyesuaikan dengan fitur yang dibutuhkan, dalam hal ini layanan yang akan digunakan adalah Railway.

## **BAB 5 KESIMPULAN DAN SARAN**

### **5.1 Kesimpulan**

Berdasarkan hasil dan rancang bangun yang telah dijabarkan, berikut adalah kesimpulan yang didapat :

1. Rancang bangun yang dilakukan berhasil mengembangkan backend untuk sistem PilihGeh yang mendukung pemilihan secara fleksibel dengan menggunakan metode *iterative development model* dan menggunakan library ExpressJs.
2. Rancang bangun yang dilakukan berhasil menyediakan fitur yang sudah direncanakan seperti fitur autentikasi, otorisasi, mengelola profil, mengelola pemilihan, maupun melaksanakan pemilihan.
3. Backend web sudah diuji dalam aspek fungsionalitas dengan menggunakan metode blackbox testing dengan menguji 132 skenario ketika mengakses endpoint dengan hasil sesuai dengan yang diharapkan.
4. Backend PilihGeh telah berhasil di deployment dengan menggunakan layanan Railway dengan tautan <https://backend-pilihgeh-production.up.railway.app/>.

### **5.2 Saran**

Adapun saran untuk pengembangan selanjutnya adalah sebagai berikut:

1. Backend PilihGeh belum menyediakan fitur untuk mengirimkan OTP lewat layanan SMS.

2. Backend PilihGeh belum menyediakan cloud untuk penyimpanan gambar yang lebih luas untuk dapat mendukung dalam mengelola banyaknya suara.

## DAFTAR PUSTAKA

- [1] A. Azis, I. Setiawan and A. Risqiantoro, "Aplikasi E-Voting Untuk Pemilihan Kepala Desa Berbasis Website," *Jurnal Of Information System Management*, vol. 1, no. 1, pp. 1-7, 2019.
- [2] R. Prananda, H. Anra and H. S. Pratiwi, "Rancang Bangun Aplikasi E-Voting Berbasis Android (Studi Kasus : Pemilihan Ketua Organisasi di Lingkungan Fakultas Teknik Universitas Tanjungpura)," *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 5, no. 1, pp. 17-21, 2017.
- [3] A. T. Siahaan and H. N. Irmada, "Aplikasi Sistem e-Voting Ketua Umum UPN Band Veteran Jakarta Berbasis Website," *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya*, vol. 2, no. 2, pp. 742-751, 2021.
- [4] R. B. G. Maulida, "Perancangan Sistem Pemilihan Gubernur Mahasiswa Dengan Metode Electronic Voting Berbasis Website," Teknik Informatika, Universitas Magelang, Magelang, 2021.
- [5] J. Pratama and R. Rachman, "Rancang Bangun Aplikasi E-Voting Ketua Organisasi Siswa Intra Sekolah (OSIS) Menggunakan Framework Laravel Berbasis Website (Studi Kasus: SMPN 39 Bandung)," *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 5, no. 4, pp. 714-718, 2022.
- [6] Y. S. Novitasari, Q. j. Adrian and W. Kurnia, "Rancang Bangun Sistem Informasi Media Pembelajaran Berbasis Website (Studi Kasus: Bimbingan Belajar De Potlood)," *Jurnal Teknologi dan Sistem Informasi (JTSI)*, vol. 2, no. 3, pp. 136-147, 2021.

- [7] F. Ulum and R. Muchtar, "Pengaruh E-Service Quality Terhadap E-Customer Satisfaction Website Start-Up Kaosyay," *Jurnal TEKNO KOMPAK*, vol. 12, no. 2, pp. 68-72, 2018.
- [8] E. Susanti, E. Fatkhiyah and E. Efendi, "Pengembangan UI/UX Pada Aplikasi M-Voting Menggunakan Metode Design Thinking," *Simposium Nasional RAPI XVIII*, pp. 364-370, 2019.
- [9] A. Mulyana and M. Aria, "Perancangan Digital Signage Sebagai Papan Informasi Digital," *Majalah Ilmiah UNIKOM*, vol. 13, no. 2, pp. 111-118, 2015.
- [10] D. Irawan and Z. Novianto, "Perancangan E-Learning Pada SMAN 1 Kota Lubuklinggau Menggunakan Framework Codeigniter (CI)," *Jurnal Digital Teknologo Informasi*, vol. 3, no. 2, pp. 53-60, 2020.
- [11] R. Fajrin, "Pengembangan Sistem Informasi Geografis Berbasis Node.JS untuk Pemetaan Mesin dan Tracking Engineer dengan Pemanfaatan Geolocation pada PT IBM Indonesia," *Jurnal Komputer Terapan*, vol. 3, no. 1, pp. 33-40, 2017.
- [12] A. Firdaus, S. Widodo, A. Sutrisman, S. G. F. Nasution and R. Mardiana, "Rancang Bangun Sistem Informasi Perpustakaan Menggunakan Web Service pada Jurusan Teknik Komputer Polsri," *Jurnal Informatika*, vol. 5, no. 2, pp. 81-87, 2019.
- [13] A. Mardan and A. Chakkaev, *Express.js Guide: The Comprehensive Book on Express.js*, eBook, 2014.
- [14] S. N. Yanti and E. Rihyanti, "Penerapan Rest API untuk Sistem Informasi Film Secara Daring," *Jurnal Informatika Universitas Pamulang*, vol. 6, no. 1, pp. 195-201, 2021.

- [15] M. I. Perkasa and E. B. Setiawan, "Pembangunan Web Service Data Masyarakat Menggunakan REST API dengan Access Token," *ULTIMA Computing*, vol. X, no. 1, pp. 19-26, 2018.
- [16] N. Nurmalasari, A. Anna and R. Arissusandi, "Rancang Bangun Sistem Informasi Akuntansi Laporan Laba Rugi Berbasis Web Pada PT. United Tractors Pontianak," *Evolusi: Jurnal Sains dan Manajemen*, vol. 7, no. 2, pp. 6-14, 2019.
- [17] B. Setiono, T. B. Adji and D. Adhipta, "Komparasi Kinerja Basis Data Oracle dan PostgreSQL Berdasarkan Estimasi Cost dan Execution Time Serta TPC-C Benchmark," Universitas Gadjah Mada, Yogyakarta.
- [18] Sequelize, "Introduction," Sequelize Contributors, July 2019. [Online]. Available: <https://sequelize.org/docs/v6/>. [Accessed 30 Mei 2023].
- [19] S. Famu, "Rancang Bangun Sistem Informasi Manajemen Berbasis Web Menggunakan NextJS di CV.Sanjaya Abadi baru," Universitas Putera Batam, Batam, 2022.
- [20] A. Mubarak, "Rancang Bangun Aplikasi Web Sekolah Menggunakan UML (Unified Modeling Language) dan Bahasa Pemrograman PHP(PHP Hypertext Preprocessor) Berorientasi Objek," *Jurnal Informatika dan Komputer*, vol. 2, no. 1, pp. 19-25, 2019.
- [21] G. Gurung and R. Shah, "Software Development Life Cycle Models-A Comparative Study," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 6, no. 4, pp. 30-37, 2021.
- [22] N. M. A. Munassar and A. Govardhan, "A Comparison Between Five Models Of Software Engineering," *International Journal of Computer Science Issues*, vol. 7, no. 5, pp. 94-101, 2010.