**SKPL-VOLMA**

**SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK**

**VOLMA (Voting Online Mahasiswa)**

Untuk:

Dosen Dani Hamdani

Dipersiapkan oleh:

Deri Indrawan (1301180097)

Ananta Ihza Ramadhan (1301184369)

Ihsan Ahsanu Amala (1301184026)

Muhammad Salman Farhan (1301180432)

Faiza Aulia Rahma Putra (1301184008)

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Program Studi S1 Informatika - Fakultas Informatika** | **Nomor Dokumen** | | **Halaman** |
| **SKPL-VOLMA** | | hlm/ 34 halaman |
| **Revisi** |  | Tgl: 14 Maret 2020 |

# **Daftar Perubahan**

|  |  |
| --- | --- |
| **Revisi** | **Deskripsi** |
| **A** | Perubahan pada Usecase Diagram, Usecase Update Profile, Usecase Update Password, Usecase Scenario delete kandidat, Usecase Scenario view kandidat, view hasil voting, dan class diagram. Mengubah konsistensi nama |
| **B** | Mengubah update profile hanya bisa dilakukan oleh user pemilih |
| **C** |  |
| **D** |  |
| **E** |  |
| **F** |  |
| **G** |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **INDEX** | **-** | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| Tgl |  |  |  |  |  |  |  |  |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

# **Daftar Halaman Perubahan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Halaman** | **Revisi** | **Halaman** | **Revisi** |
| **13**  **20**  **22**  **27**  **30**  **31**  **35** | **Usecase Diagram**  **Usecase Scenario Delete Kandidat**  **Usecase Scenario View Kandidat**  **View Hasil Voting**  **Usecase Update Profile**  **UseCase Update Password**  **Class diagram** |  |  |

**Daftar Isi**

**Daftar Perubahan…………………………..………………...………………………………….2**

**Daftar Halaman Perubahan.……………………………………………………………………3**

**Daftar Isi……………………....….……………………………………………………………...4**

**1. Pendahuluan……………..………………………………………………………………5**

1.1. Tujuan Penulisan Dokumen..……………….…………………………………….5

1.2. Ruang Lingkup / Cakupan Dokumen…………………………………………......5

1.3. Definisi, Singkatan, dan Akronim………………………………………………...5

1.4. Referensi…………………………………………………………………………..6

**2. Deskripsi Global Perangkat Lunak……………………………………………..……...7**

2.1. Statement of Objective Perangkat Lunak……………………….………………...7

2.2. Perspektif dan Fungsi Perangkat Lunak…………………………………………..7

2.3. Profil dan Karakteristik Pengguna………………………………………………...7

2.4. Arsitektur Lingkungan Operasi Perangkat Lunak (digambarkan)...........................8

2.5. Kebutuhan Perangkat Keras……………………………………………………….9

2.5.1. Kebutuhan Perangkat Keras……………………………………………………….9

2.5.2. Fakta Perangkat Keras server..…………………………....……………………….9

2.5.3. Fakta Perangkat Keras Smartphone…………………....………………………….9

2.6. Asumsi dan Batasan Perangkat Lunak………………………………………..….10

2.6.1 Kebutuhan Perangkat Lunak………………………………………...…………...10

2.6.2 Fakta Perangkat Lunak……………………………………………...………....…11

**3. Deskripsi Rinci Perangkat Lunak………………………………………………..…....11**

3.1. Deskripsi Kebutuhan…………………………………………………………..…11

3.1.1. Kebutuhan Fungsional………………………………………………………...…12

3.1.2. Kebutuhan Non-Fungsional………………………………………………..….…13

3.2 Pemodelan Analisis………………………………………………………..….….14

3.2.1. Use Case Diagram…...……………………………………………………..…….14

3.2.2. Use Case Scenario…...……………………………………………………..…….14

3.3. Class Diagram………………………………………………………………....…34

**1.** **Pendahuluan**

## **1.1.** **Tujuan Penulisan Dokumen**

Dokumen ini berisi penjelasan Spesifikasi Kebutuhan Perangkat Lunak (SKPL) atau Software Requirement Specification (SRS) dengan pendekatan berorientasi proses dari perangkat lunak yang akan dibuat.

## **1.2.** **Ruang Lingkup / Cakupan Dokumen**

Tujuan penulisan dokumen ini adalah untuk menguraikan proses-proses tahapan pembuatan perangkat lunak yang akan dibangun. Bagi pihak pengembang, SKPL ini dapat digunakan sebagai acuan dalam mengembangkan perangkat lunak secara detail agar perangkat lunak yang dibangun sesuai dengan kebutuhan yang diinginkan pengguna. Diharapkan dengan adanya dokumen SKPL ini pengembangan perangkat lunak lebih terarah kepada tujuan dari pengembangan perangkat lunak ini. Sedangkan bagi pengguna, dokumen SKPL ini digunakan untuk mencatat spesifikasi kebutuhan perangkat lunak yang akan dibangun dan harapan yang diinginkan pengguna.

## **1.3.** **Definisi, Singkatan, dan Akronim**

Berikut adalah daftar definisi dan istilah penting yang digunakan dalam dokumen SKPL ini :

* SKPL : Spesifikasi Kebutuhan Perangkat Lunak Dokumen hasil analisis yang berisi spesifikasi kebutuhan perangkat lunak.
* Use Case Diagram : Gambaran grafik dari beberapa atau semua aktor, Use case dan interaksinya.
* VOLMA : Voting Online Mahasiswa.
* SRS : Software Requirement Specification.
* Voting Online : Keputusan yang diambil berdasarkan suara terbanyak melalui media elektronik.
* Operating System : adalah [perangkat lunak sistem](https://id.wikipedia.org/wiki/Perangkat_lunak_sistem) yang mengatur sumber daya dari [perangkat keras](https://id.wikipedia.org/wiki/Perangkat_keras) dan [perangkat lunak](https://id.wikipedia.org/wiki/Perangkat_lunak), serta sebagai [daemon](https://id.wikipedia.org/wiki/Daemon) untuk [program komputer](https://id.wikipedia.org/wiki/Program_komputer).
* HTML : *Hypertext Markup Language* (HTML) adalah sebuah [*bahasa markah*](https://id.wikipedia.org/wiki/Bahasa_markah) yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah [penjelajah web](https://id.wikipedia.org/wiki/Penjelajah_web) Internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi.
* Bootstrap : Bootstrap adalah kerangka kerja CSS yang [sumber terbuka](https://id.wikipedia.org/wiki/Perangkat_lunak_sumber_terbuka) dan [bebas](https://id.wikipedia.org/wiki/Perangkat_lunak_bebas) untuk merancang [situs web](https://id.wikipedia.org/wiki/Situs_web) dan [aplikasi web](https://id.wikipedia.org/wiki/Aplikasi_web). Kerangka kerja ini berisi templat desain berbasis [HTML](https://id.wikipedia.org/wiki/HTML) dan [CSS](https://id.wikipedia.org/wiki/Cascading_Style_Sheets) untuk [tipografi](https://id.wikipedia.org/wiki/Tipografi), formulir, tombol, navigasi, dan komponen antarmuka lainnya, serta juga ekstensi opsional [JavaScript](https://id.wikipedia.org/wiki/JavaScript).
* CodeIgniter : CodeIgniter merupakan aplikasi [sumber terbuka](https://id.wikipedia.org/wiki/Sumber_terbuka) yang berupa kerangka kerja PHP dengan model [MVC](https://id.wikipedia.org/wiki/MVC) (Model, View, Controller) untuk membangun [situs web](https://id.wikipedia.org/wiki/Situs_web) dinamis dengan menggunakan [PHP](https://id.wikipedia.org/wiki/PHP).
* JQuery : adalah pustaka JavaScript lintas-platform yang didesain untuk menyederhanakan client-side scripting pada HTML.

**1.4.** **Referensi**

* Operating System :<https://id.wikipedia.org/wiki/Sistem_operasi> (diakses pada 14, Januari 2019)
* HTML : <https://id.wikipedia.org/wiki/HTML> (diakses pada 14, Januari 2019)
* Bootstrap : <https://id.wikipedia.org/wiki/Bootstrap_(kerangka_kerja)> (diakses pada 14, Januari 2019)
* CodeIgniter : <https://id.wikipedia.org/wiki/CodeIgniter> (diakses pada 14, Januari 2019)
* Jquery : <https://id.wikipedia.org/wiki/JQuery> (diakses pada 14, Januari 2019)

# 

# **2.** **Deskripsi Global Perangkat Lunak**

## **2.1.** **Statement of Objective Perangkat Lunak**

VOLMA merupakan aplikasi berbasis web yang bertujuan untuk membantu mahasiswa dalam melakukan pemilihan umum secara online. VOLMA mempunyai beberapa fitur diantaranya adalah fitur voting yang berguna untuk melakukan voting dari sisi user sebagai pemilih. Hasil voting akan dikalkulasi oleh sistem dan mengenerate kandidat terpilih. Aplikasi ini juga menyediakan fitur lihat info kandidat yang berguna menjadi acuan pemilih dalam melakukan voting.

## **2.2. Perspektif dan Fungsi Perangkat Lunak**

Perangkat lunak ini memiliki beberapa fungsi utama, antara lain :

1. Registrasi Akun user
2. Registrasi Akun admin
3. Login
4. Update Profile
5. Update Password
6. View Data Kandidat
7. Input Data Kandidat
8. Update Data Kandidat
9. Delete Data Kandidat
10. Input Voting
11. View Hasil Voting

## **2.3.** **Profil dan Karakteristik Pengguna**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kategori Pengguna** | **Tugas** | **Hak akses Ke Aplikasi** | **Kemampuan yang Harus Dimiliki** |
| User Admin | Menginputkan data kandidat, Melakukan perubahan data kandidat | * Register * Login * Update Profile * Update Password * Input data Kandidat * Update Data Kandidat * View Data Kandidat * Delete Data Kandidat | Mengoperasikan komputer / handphone |
| User Pemilih | Melakukan voting | * Register * Login * Update Profile * Update Password * Input Voting * View data Kandidat * View Hasil Voting | Mengoperasikan komputer / handphone |

## **2.4.** **Arsitektur Lingkungan Operasi Perangkat Lunak**

Asumsi perangkat lunak:

1. *User* dibagi menjadi 2 *role,* *Pemilih* dan *Admin.*

*User Admin* adalah seorang mahasiswa aktif yang bertugas mengelola data pemilih dan data kandidat

*User Pemilih* adalah seorang mahasiswa aktif yang bertugas untuk melakukan voting

1. Semua *User* dapat melakukan *register, login, update password.*
2. *User pemilih* dapat melakukan *voting*, *melihat data kandidat, melihat hasil voting, update profile.*
3. *User* *admin* dapat *input data kandidat, update data kandidat, update data kandidat, hapus data kandidat.*

## 

## **2.5.** **Kebutuhan Perangkat Keras**

## **2.5.1** **Kebutuhan Perangkat Keras**

* Intel Processor Dual Core Keatas
* RAM 2 GB
* Hard Disk 256GB
* Android 5.0
* Internet

## **2.5.2** **Fakta Perangkat Keras Server**

* Intel core i3 / ryzen 3
* RAM 4GB
* Hard Disk 256GB

## **2.5.3** **Fakta Perangkat Keras Smartphone**

* Android 5.0 - Up
* RAM 2 GB
* Internet

**2.6.** **Asumsi dan Batasan Perangkat Lunak**

## **2.6.1** **Kebutuhan Perangkat Lunak**

* *Operating System*
* HTML
* Bootstrap
* CodeIgniter
* JQuery

## **2.6.2** **Fakta Perangkat Lunak**

* *Operating System* Linux
* *Operating System* Windows
* *Operating System* Android
* *Operating System* IOS

**Batasan-batasan :**

Program ini hanya digunakan dalam ruang lingkup kampus, dimana pengguna dari program ini dikhususkan terhadap mahasiswa yang ingin melakukan pemilihan dalam lingkungan intra-kampus. Dalam Program ini terdapat beberapa batasan yang harus dipatuhi. Misalnya :

* Aplikasi harus terhubung dengan internet.
* Aplikasi hanya dapat digunakan pada platform web.
* Aplikasi hanya dapat digunakan oleh mahasiswa aktif yang dibuktikan dengan nim.

# **3.** **Deskripsi Rinci Perangkat Lunak**

## **3.1.** **Deskripsi Kebutuhan**

### **3.1.1.** **Kebutuhan Fungsional**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode kebutuhan** | **Fungsi** | **Deskripsi** |
| **1.** | **FR-01** | **Input data pemilih** | Fungsi ini digunakan oleh pemilih untuk menginputkan data pemilih ke system. |
| **2.** | **FR-02** | **Validasi data pemilih** | Fungsi ini digunakan oleh database untuk memvalidasi data pemilih dari database ke system. |
| **3.** | **FR-03** | **Input data login** | Fungsi ini digunakan oleh pemilih untuk menginputkan data login ke system. |
| **4.** | **FR-04** | **Verifikasi Login** | Fungsi ini digunakan oleh database untuk memverifikasi data login dari database ke admin. |
| **5.** | **FR-05** | **Input data kandidat** | Fungsi ini digunakan oleh admin untuk menginput data kandidat ke system. |
| **6.** | **FR-06** | **View data kandidat** | Fungsi ini digunakan oleh admin dan pemilih untuk melihat data kandidat yang tersedia didalam system. |
| **7.** | **FR-a07** | **Input voting** | Fungsi ini digunakan oleh pemilih untuk menginputkan votingnya ke dalam system. |
| **8.** | **FR-08** | **View hasil voting** | Fungsi ini digunakan oleh pemilih dan admin untuk melihat hasil voting yang terdapat didalam sistem. |

### 

### **3.1.2.** **Kebutuhan Non-Fungsional**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Quality** | **Kode kebutuhan** | **Deskripsi** |
| **1.** | **User interface Delay** | **NFR-01** | Jarak antar waktu dari satu tab ke tab lainnya yaitu 5 milisekon. |
| **2.** | **Time vote management** | **NFR-02** | Input voting hanya bisa dilakukan pada pukul 07.00 - 16.00. |
| **3.** | **System Record real time** | **NFR-03** | Sistem akan langsung memberikan data input yang telah diinputkan oleh pemilih ke dalam database tanpa ada delay. |
| **4.** | **Automatic screen adjustment** | **NFR-04** | Tampilan web akan menyesuaikan dengan orientasi layar yang digunakan oleh user. |
| **5.** | **Voter Data Security** | **NFR-05** | Sistem akan memvalidasi data pemilih yang diinputkan oleh pemilih apakah valid atau tidak. |

## 

## 

## **3.2 Pemodelan Analisis**

### **3.2.1. Use Case Diagram**

#### 

#### **3.2.1.1.** **Use Case Skenario**

|  |  |
| --- | --- |
| **Use Case** | **Input kandidat** |
| **Input** | Data kandidat ( nama\_kandidat , NIM , angkatan , fakultas , deskripsi) |
| **Output** | data kandidat tersimpan di dalam database |
| **Actor** | Admin |
| **Precondition** | Jika ingin menggunakan input kandidat, admin telah login dan data kandidat belum tersimpan di database kandidat |
| **Post Condition** | Admin telah menginputkan data kandidat dan system berhasil menyimpan data kandidat di database kandidat |
| **Description** | Untuk menambahkan dan menyimpan data kandidat yang baru |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka menu kelola kandidat |  | |  | 2. Menampilkan tampilan kelola kandidat | | 3. Memilih tombol input |  | |  | 4. Menampilkan form input | | 5. Memasukkan data kandidat |  | | 6. Menekan tombol simpan |  | |  | 7. Validasi data kandidat, jika berhasil lanjut ke 8, jika tidak kembali ke 4 | |  | 8. Merekam data kandidat yang diinputkan Admin | |  | 9. Menyimpan dan memproses data kandidat ke dalam database kandidat | |  | 10. Menampilkan data kandidat yang telah disimpan | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **Login** |
| **Input** | Data login admin/pemilih |
| **Output** | Actor berhasil login |
| **Actor** | Admin, Pemilih |
| **Precondition** | Actor ingin menggunakan program dan belum melakukan login |
| **Post Condition** | Actor telah melakukan login dan berinteraksi dengan program |
| **Description** | Untuk dapat mengakses semua fungsionalitas yang ada pada aplikasi |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Menginputkan username dan password |  | | 2. Menekan tombol login |  | |  | 3. Eksekusi validasi NIM dan password yang diinputkan actor dengan yang tersimpan dalam database | |  | 4.Menampilkan form input | |  | 5. Jika sesuai tampil notif login berhasil dan lanjut ke 6 | |  | 6. Menampilkan halaman utama | |  | 7. Melakukan aktivitas di dalam program | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **Update Kandidat** |
| **Input** | Data kandidat (nama\_Kandidat, NIM, Angkatan, Fakultas, Deskripsi) |
| **Output** | Data kandidat (nama\_Kandidat, NIM, Angkatan, Fakultas, Deskripsi) yang tersimpan dalam Database |
| **Actor** | Admin |
| **Precondition** | Jika ingin menggunakan Update Kandidat, Admin harus sudah Login untuk menggunakan menu update kandidat dan data kandidat yang akan diubah sudah terdapat didalam database . |
| **Post Condition** | Admin telah mengubah data kandidat yang dipilih dan dimasukan kedalam database. |
| **Description** | Untuk mengubah data kandidat yang lama dan menyimpan data yang baru |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka menu view kandidat |  | |  | 2. Menampilkan data kandidat | | 3.Memilih kandidat yang akan diubah |  | |  | 4. Menampilkan form berisi data kandidat yang lama | | 5. Memasukan data kandidat(nama Kandidat, NIM, Angkatan, Fakultas, Deskripsi) |  | | 6. Menekan tombol submit |  | |  | 7. Menampilkan alert konfirmasi pengubahan data | | 8. Jika “ya”, lanjut ke 9, jika “tidak” kembali ke 2 |  | |  | 8. Validasi data kandidat, jika berhasil lanjut ke 9, jika tidak kembali ke 4 dan menampilkan notifikasi error | |  | 9. Simpan data kandidat ke database | |  | 10. Menampilkan data kandidat yang diubah dan notifikasi berhasil | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **Delete Kandidat** |
| **Input** | Data kandidat (nama kandidat , NIM) |
| **Output** | Data kandidat yang tersimpan di database kandidat |
| **Actor** | Admin |
| **Precondition** | Jika ingin menggunakan Delete kandidat , admin telah login untuk menggunakan menu delete kandidat |
| **Post Condition** | Admin telah menghapus data kandidat dan data kandidat telah dihapus di datastore |
| **Description** | Untuk Menghapus data kandidat |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka menu view kandidat |  | |  | 2. Menampilkan data kandidat | | 3. Memilih kandidat yang ingin dihapus dengan menekan tombol delete |  | |  | 4. Menampilkan alert konfirmasi penghapusan | | 5. Jika “ya” lanjut ke 6, jika “tidak” kembali ke 2 |  | |  | 6. Menghapus data kandidat dari database | |  | 7. Menampilkan notifikasi berhasil dan menampilkan data kandidat | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **View Kandidat** |
| **Input** | Data kandidat (nama kandidat , NIM) |
| **Output** | Data Kandidat yang terdapat didalam Database |
| **Actor** | Admin , Pemilih |
| **Precondition** | Jika ingin menggunakan view kandidat, admin/pemilih telah login terlebih dahulu untuk menggunakan view kandidat |
| **Post Condition** | System telah menampilkan data kandidat |
| **Description** | Untuk menampilkan data kandidat yang tersedia di dalam database |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka Menu Kelola Kandidat |  | |  | 2.Menampilkan Menu Kelola Kandidat | |  | 3.jika data kandidat terdapat dalam database maka system akan menampilkan data kandidat , jika tidak maka system akan menampilkan “data tidak ditemukan” | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **Search Kandidat** |
| **Input** | Data kandidat (nama kandidat , NIM) |
| **Output** | Data kandidat yang tersimpan di database kandidat |
| **Actor** | Admin , pemilih |
| **Precondition** | Jika ingin menggunakan searching kandidat , admin atau pemilih telah menginput data kandidat yang akan di searching. |
| **Post Condition** | System telah mendapatkan data kandidat |
| **Description** | Untuk mencari data kandidat yang tersedia di dalam database |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka menu kelola kandidat |  | | 2. Memilih menu search kandidat |  | | 2. Menginput data kandidat ( nama kandidat , NIM) |  | |  | 3.jika data kandidat terdapat dalam database maka system akan mengambil data kandidat tersebut , jika tidak maka system akan menampilkan “data tidak ditemukan” | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **Input Periode Pemilihan** |
| **Input** | Periode\_Pemilihan |
| **Output** | Periode Pemilihan terinput kedalam system |
| **Actor** | Admin |
| **Precondition** | Jika ingin menggunakan input periode pemilihan, admin harus login terlebih dahulu |
| **Post Condition** | Admin telah menginput periode pemilihan |
| **Description** | Untuk menginput periode pemilihan |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka Menu input periode pemilihan |  | |  | 2.Menampilkan Menu input periode pemilihan | | 3. Admin menginput periode pemilihan |  | |  | 4.Sistem menyimpan hasil inputan periode pemilihan dari admin | |

|  |  |
| --- | --- |
| **Use Case** | **Input Voting** |
| **Input** | Identitas\_kandidat |
| **Output** | Voting terinput kedalam database Data voting |
| **Actor** | Pemilih |
| **Precondition** | Jika ingin menggunakan input Voting , pemilih harus login terlebih dahulu |
| **Post Condition** | Pemilih telah menginput voting |
| **Description** | Untuk menginput voting |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka Menu input voting |  | |  | 2.Menampilkan Menu input voting | | 3. Melihat data kandidat | ` | | 4. Memvoting/memilih kandidat yang ditampilkan oleh sistem |  | |  | 4.Sistem menyimpan hasil inputan voting kedalam database data voting | |

|  |  |
| --- | --- |
| **Use Case** | **View Hasil Voting** |
| **Input** | Identitas\_kandidat |
| **Output** | hasil voting berdasarkan masing-masing kandidat yang terdapat didalam database |
| **Actor** | Pemilih, admin |
| **Precondition** | Jika ingin menggunakan view hasil voting , pemilih/admin harus login terlebih dahulu |
| **Post Condition** | Pemilih/admin telah melihat hasil voting |
| **Description** | Untuk melihat hasil voting |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka Menu view hasil voting |  | |  | 2.Menampilkan hasil voting berdasarkan masing-masing kandidat yang terdapat didalam database | |

#### 

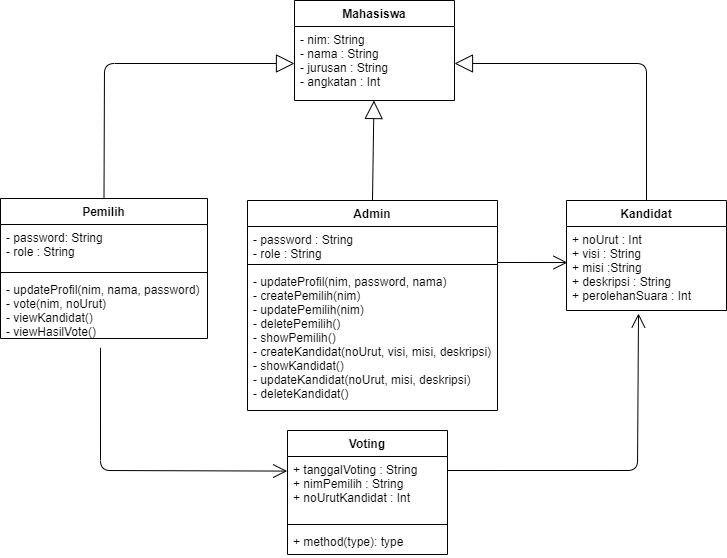
|  |  |
| --- | --- |
| **Use Case** | **Register** |
| **Input** | Data\_pemilih ( nama\_pemilih, NIM, Fakultas, angkatan) |
| **Output** | Data\_pemilih tersimpan ke dalam database data account |
| **Actor** | Pemilih |
| **Precondition** | Jika ingin menggunakan register pemilih harus datang pos register yang disediakan di daerah Telkom University |
| **Post Condition** | Data pemilih telah teregister di dalam database |
| **Description** | Untuk meregister pemilih sehingga pemilih dapat menggunakan login |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Menginputkan data pemilih kedalam sistem |  | |  | 2.Menyimpan data pemilih ke dalam database data account | |

#### 

|  |  |
| --- | --- |
| **Use Case** | **Update Profile** |
| **Input** | Profile lama , Profile baru |
| **Output** | Profile baru telah tersimpan kedalam database Data pemilih |
| **Actor** | Pemilih |
| **Precondition** | Jika ingin menggunakan update profile , pemilih harus login terlebih dahulu |
| **Post Condition** | Pemilih telah mengupdate profile |
| **Description** | Untuk mengupdate profile |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka tab edit profile |  | |  | 2.Menampilkan form edit profile yang telah terisi oleh profile yang terdapat di database . | | 3. Mengubah profile yang terdapat didalam form edit profile |  | | 4. Menekan tombol simpan |  | |  | 5.Jika gagal kembali ke nomor 2, jika berhasil lanjut ke nomor 6 | |  | 6.Sistem menyimpan hasil inputan update profile kedalam database pemilih. | |

|  |  |
| --- | --- |
| **Use Case** | **Update Password** |
| **Input** | Password lama, Password baru |
| **Output** | Password baru telah tersimpan kedalam database |
| **Actor** | Pemilih |
| **Precondition** | Jika ingin menggunakan mengupdate password, pemilih harus login terlebih dahulu |
| **Post Condition** | Pemilih telah mengupdate password |
| **Description** | Untuk mengupdate password |
| **Typical Course of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1.Membuka tab edit profile |  | | 2. Pilih edit password |  | |  | 3.Menampilkan form edit password | | 4. Masukan password lama dan password baru |  | |  | 5. Jika gagal kembali ke no 3  Jika berhasil lanjut ke no 6 | |  | 6. Data telah tersimpan kedalam database | |

**3.3 Class Diagram**

****