# BAB III

**ANALISA DAN PERANCANGAN**

## Analisa Permasalahan

Seperti yang sudah diketahui sebelumnya salah satu organisasi yang mengeluarkan peringkat *website* Perguruan Tinggi adalah Webometrics. The “Webometrics Rangking of World Universities” adalah sebuah inisiatif dari Cybermetrics Lab, sebuah kelompok penelitian milik *Consejo Superior de Investigaciones Cientificas* (CSIC), badan penelitian terbesar di Spanyol. Cybermetrics Lab sendiri dikhususkan untuk melakukan analisis kuantitatif Internet dan *Web Content*, khusus yang berkaitan dengan proses generasi dan komunikasi ilmiah dari pengetahuan ilmiah.

Padarilisan Webometrics edisi Juli 2018 terdapat perubahan indicator dari edisi sebelumnya. Perubahan tersebut terletak pada masing-masing indikator. Semula bobot indikator *Presence* sebesar 35% kemudian dirubah menjadi 5%, sedangkan *Excellence* yang semula 5% dirubah menjadi 35%. Sehingga sekarang untuk pembobotan Webometrics edisi Juli 2018 secara keseluruhan menjadi 5% untuk *Presence*, 50% untuk *Visibility*, 10% untuk *Transparency*, dan 35% untuk *Excellent*.

Dilihat dari perubahan tersebut, dapat diketahui bahwa pemeringkatan Webometrics sekarang bersifat subjektif dimana kriteria yang berpengaruh

34

terhadap Perguruan Tinggi bergantung pada penelitian dosen yaitu sebesar 45% (bobot *Transparency* ditambah *Excellent*). Yang mana pada edisi sebelumnya yaitu Januari 2018, penelitian dosen hanya berdampak 15% (10% *Transparency* dan 5% *Excellent*). Hal tersebut tentunya akan membuat sistem perangkingan pada Webometrics tidak lagi berdasarkan kualitas dari situs web melainkan 45% dari penelitian dosen dan 55% dari kualitas situs web.

## Usulan Pemecahan Masalah

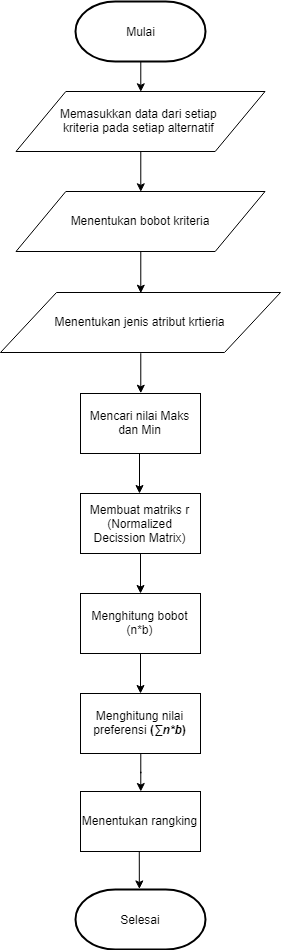
Berdasarkan analisa permasalahan diatas dapat diberikan sebuah usulan pemecahan masalah yaitu membuat sistem informasi perangkingan *Website* Perguruan Tinggi se-Jawa Timur menggunakan metode Simple Additive Weighting dan berdasarkan kriteria *Alexa Rank, SEMRush Rank, Backlink, Usia website, dan Google Index.*

Dengan menggunakan kriteria-kriteria yang telah disebutkan bertujuan sistem yang akan dibuat murni menilai kualitas/popularitas dari *website* Perguruan Tinggi se-Jawa Timur. Tidak hanya itu, nantinya sistem ini juga diharapkan dapat di terapkan di Provinsi-provinsi lain yang membutuhkan sistem tersebut dalam perangkingan *Website* PerguruanTingginya.

## Perancangan Sistem

* + 1. **Gambaran Umum Sistem yang Dibuat**

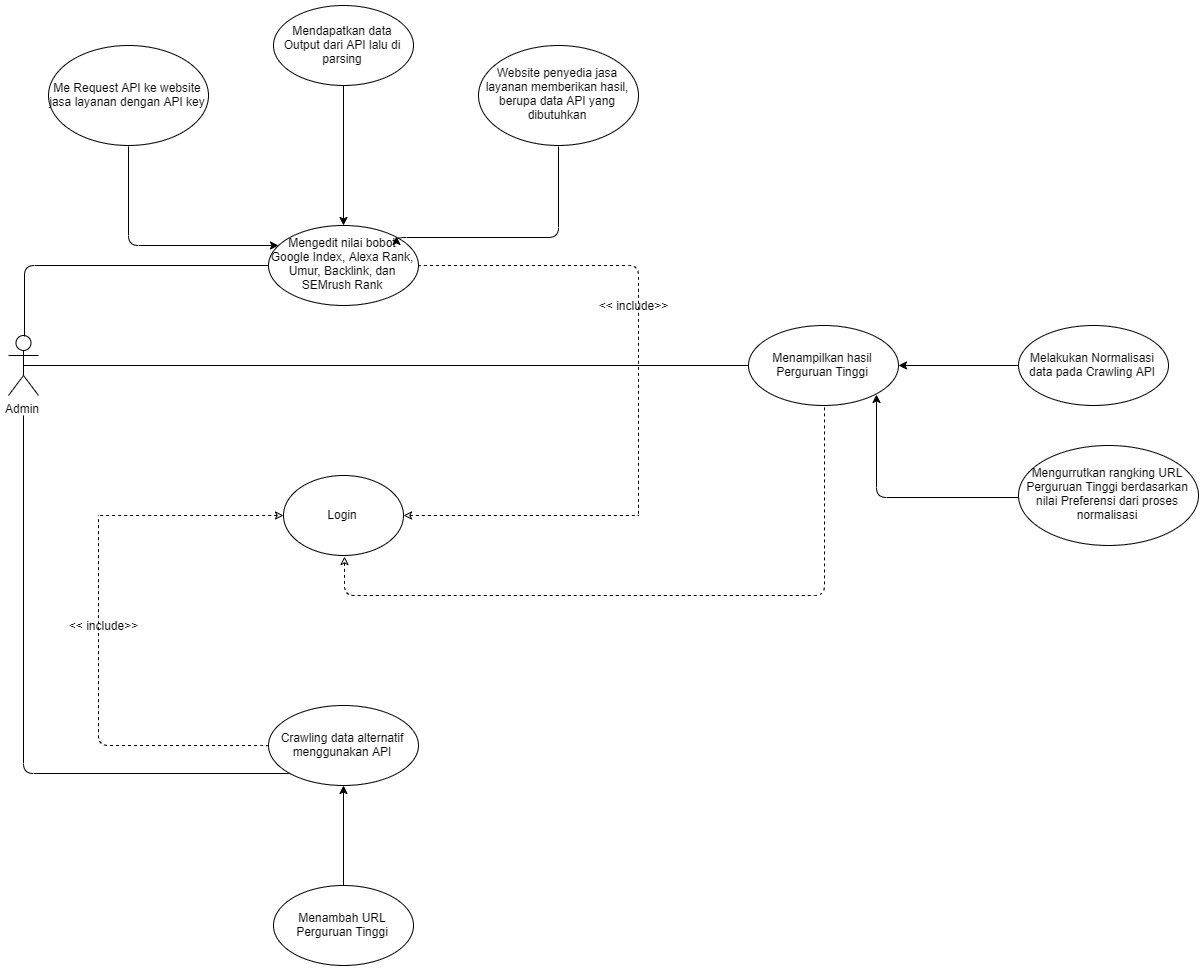
Terdapat dua aktor didalam sistem informasi perangkingan situsweb Perguruan Tinggi ini yaitu administrator dan visitor. Administrator dapat membuat, mengubah, dan menghapus isi dalam sistem, dimana isi dalam sistem tersebut merupakan kumpulan data yang nantinya akan dikelola oleh sistem. Visitor disini hanya melihat hasil perangkingan, yang mana hasil dari perangkingan tersebut merupakan hasil pengelolaan data oleh sistem. Dan berikut merupakan langkah- langkah penyelesaian yang digambarkan pada *flowchart* (Gambar3.1):



**Gambar 3.1 Flowchart SAW**

## Diagram UseCase

Pada bagian ini akan menjelaskan tentang interaksi-interaksi antara pengguna sistem (*user*) dan sistem. Interaksi yang terjadi antara lain:

* + - 1. Perangkingan

**Gambar 3.2 Diagram *Use Case* Perangkingan**

**Tabel 3.1 *Use Case Scenario* Perangkingan**

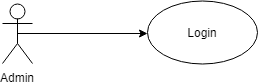
|  |  |  |
| --- | --- | --- |
| **Nama *use case*** | Perangkingan | |
| **Aktor** | Administrator | |
| **Deskripsi** | *Use case* menggambarkan kegiatan menampilkan hasil  perangkingan *website* Perguruan Tinggi. | |
| **Prakondisi** | Digunakan oleh aktor untuk menjalankan sistem  informasi pada sistem terkait pada proses lanjutansistem | |
| **Alur Utama** | **Kegiatan Aktor** | **Respon Sistem** |

|  |  |  |
| --- | --- | --- |
| Perangkingan website Perguruan Tinggi | 1. Mengedit nilai bobot Google Index, Alexa Rank, Umur, Backlink, dan SEMrushRank. 2. Menambah URL PerguruanTinggi. 3. Crawling data   Perguruan Tinggi. menggunakan API.   1. Me request API ke website penyedia jasa layanan dengan API key. 2. Mendapatkan data Output dari API, kemudian diparsing. 3. Website penyedia jasa layanan memberikan hasil berupa dataAPI   yangdibutuhkan. | 1. Mengubah nilai atribut Google Index, Alexa   Rank, Umur, Backlink, dan SEMrushRank.   1. Menyimpan data. 2. Mendapatkan data yang diperlukan. 3. Mengalihkan halaman menuju *website* penyedia jasa layanan. 4. Memparsing data sesuai dengankriteria. 5. Menyimpan hasil API |

|  |  |  |
| --- | --- | --- |
|  | 1. Melakukan Normalisasi padahasil *Crawling*API. 2. Mengurutkan rangking URL   Perguruan Tinggi  berdasarkan nilai Preferensi dari proses normalisasi.   1. Menampilkan hasil perangkingan. | kedalam  *Database*.   1. Melakukan normalisasi pada data yang telah didapatkan dari websitepenyedia jasa layanan. 2. Mengurutkan rangking Perguruan Tinggi berdasarkan normalisasi, hasil dari pengolahan data yang didapatkan dari website penyedia jasa layanan. |

|  |  |  |
| --- | --- | --- |
|  |  | 18. Menampilkan halaman  perangkingan. |
| **Alur**  **Alternatif** | Jika salah maka sistem akan kembali kepada halaman  sebelumnya. | |
| **Kesimpulan** | Aktor masuk kedalam sistem, mengubah, dan  menampilkan data. | |
| **Post Kondisi** | Sistem berhasil menyimpan dan menampilkan data. | |

* + - 1. *Login*



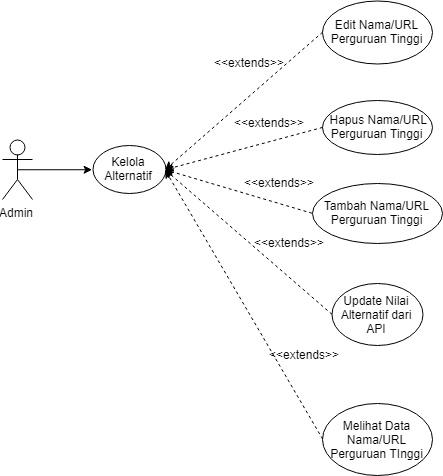
**Gambar 3.3 Diagram *Use case Login***

**Tabel 3.2 Tabel *Use Case Scenario* Login**

|  |  |
| --- | --- |
| **Nama *use case*** | *Login* |
| **Aktor** | Administrator |
| **Deskripsi** | *Use case* menggambarkan kegiatan memasukkan  *username* dan *password* untuk mengakses sistem |
| **Prakondisi** | Digunakan oleh aktor untuk membuka akses pada sistem  terkait pada proses lanjutan sistem |

|  |  |  |
| --- | --- | --- |
| **Alur Utama** | **Kegiatan Aktor** | **Respon Sistem** |
| 1. Memilih menu   *Login.*   1. Memasukkan *username* dan *password* | 1. Cek *username* dan *password.* 2. Menampilkan Halaman utama sistem sebagai   admin. |
| **Alur**  **Alternatif** | Jika salah maka sistem akan menampilkanpesan  kesalahan | |
| **Kesimpulan** | Aktor masuk kedalam sistem | |
| **Post Kondisi** | Sistem berhasil diakses | |

* + - 1. KelolaAlternatif



**Gambar 3.4 Diagram *Use case* Kelola Alternatif**

**Tabel 3.3 Tabel *Use Case Scenario* Kelola Alternatif**

|  |  |
| --- | --- |
| **Nama *use case*** | Kelola Alternatif |
| **Aktor** | Administrator |

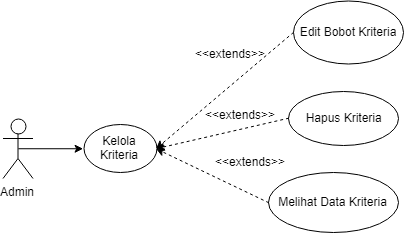
|  |  |  |
| --- | --- | --- |
| **Deskripsi** | Digunakan oleh aktor untuk melakukan Edit Nama/URL Perguruan Tinggi, Hapus Nama/URL Perguruan Tinggi,  Tambah Nama/URL Perguruan Tinggi, Update Nilai Alternatif dari API, dan Melihat Data  Nama/URL Perguruan Tinggi | |
| **Prakondisi** | Aktor melihat adanya pembaruan/kesalahan pada  data alternatif | |
| **Alur Utama** | **Kegiatan Aktor** | **Respon Sistem** |
| 1.Edit Nama/URL Perguruan Tinggi | 1.Memilih menu kelola alternatif  3.Memilih menu Edit Nama/URL Perguruan Tinggi   1. Mengedit data Nama/URL Perguruan Tinggi 2. Mengklik tombolsave | 2.Menampilkan halaman kelola alternatif  4.Menampilkan Form Edit.  7.Data tersimpan dalam database |

|  |  |  |
| --- | --- | --- |
| 2. Hapus Nama/URL Perguruan Tinggi | 1.Memilih menu kelola alternatif  3. Memilih menu Hapus Nama/URL Perguruan Tinggi  5.Memilih “Ya” pada notifikasi | 2.Menampilkan halaman kelola alternatif  4.Mengirimkan Notifikasi Hapus padaaktor.  7.Data terhapus dalam database |
| 3. Tambah  Nama/URL Perguruan Tinggi | 1.Memilih menu kelola alternatif  3.Memilih menu Tambah Nama/URL PerguruanTinggi   1. Menginput data Nama/URL Perguruan Tinggi baru 2. Mengklik tombolsave | 2.Menampilkan halaman kelola alternatif  4.Menampilkan Form Tambah  7.Data tersimpan dalam database |

|  |  |  |
| --- | --- | --- |
| 4. Update Nilai Alternatif dari API | 1.Memilih menu kelola alternatif  3.Memilih menu Update Nilai Alternatif dari API | 2.Menampilkan halaman kelola alternatif   1. Sistem mengambil ulang Data Nilai Alternatif Baru dari API 2. Data Nilai Alternatif Baru tersimpan dalam database |
| 5. Melihat Data Nama/URLPerguruan Tinggi | 1.Memilih menu kelola alternatif | 1. Menampilkan halaman kelola alternatif 2. Menampilkan data alternatif |
| **Alur Alternatif** | 1. Pada menu Edit Nama/URL Perguruan Tinggi & Tambah Nama/URL  Perguruan Tinggi jika tidakmenekan | |

|  |  |
| --- | --- |
|  | save,makadatayangadapadadatabase tidak akan terjadiperubahan  2. Pada menu Hapus Nama/URL Perguruan Tinggi jika memilih“Tidak” pada notifikasi yang dikirimkan oleh sistem, maka data yang adadalam  database tidak akan hilang/terhapus. |
| **Kesimpulan** | Aktor mengelola keseluruhan data alternatif |
| **Post Kondisi** | Data alternatif dapat diperbarui |

* + - 1. Kelola Kriteria



**Gambar 3.5 Diagram *Use case* Kelola Kriteria**

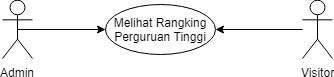
**Tabel 3.4 Tabel *Use Case Scenario* Kelola Kriteria**

|  |  |  |
| --- | --- | --- |
| **Nama *use case*** | Kelola Alternatif | |
| **Aktor** | Administrator | |
| **Deskripsi** | Digunakan oleh aktor untuk melakukan Edit Bobot  Kriteria, Hapus Kriteria, dan Melihat Data Kriteria. | |
| **Prakondisi** | Aktor melihat adanya pembaruan/kesalahan pada  data alternatif | |
| **Alur Utama** | **Kegiatan Aktor** | **Respon Sistem** |
| 1.Edit Bobot Kriteria | 1.Memilih menu kelola kriteria  3.Memilih menu Edit Bobot Kriteria 5.Mengedit dataBobot  Kriteria  6.Mengklik tombol save | 2.Menampilkan halaman kelola kriteria  4.Menampilkan Form Edit.  7.Data tersimpan dalam database |
| 2. Hapus Kriteria | 1.Memilih menu kelola kriteria  3. Memilih menu Hapus  Kriteria | 2.Menampilkan halaman kelola kriteria |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | 5.Memilih notifikasi | “Ya” | pada | 4. Mengirimkan Notifikasi Hapus padaaktor.  7.Data terhapus dalam database | |
| 3. Melihat Kriteria | Data | 1.Memilih menu kelola kriteria | | | 2.Menampilkan halaman kriteria  4.Menampilkan kriteria | kelola  data |
| **Alur Alternatif** | | 1. Pada menu Edit Bobot Kriteria jika tidak menekan save, maka data yang ada pada database tidak akan terjadi perubahan 2. Pada menu Hapus Kriteria jika memilih “Tidak” pada notifikasi yang   dikirimkan oleh sistem, maka data yang | | | | |

|  |  |
| --- | --- |
|  | ada dalam database tidak akan  hilang/terhapus. |
| **Kesimpulan** | Aktor mengelola keseluruhan data kriteria |
| **Post Kondisi** | Data kriteria dapat diperbarui |

* + - 1. Melihat Rangking Perguruan Tinggi



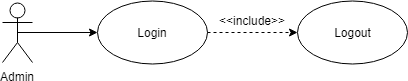
**Gambar 3.6 Diagram *Use case* Melihat Rangking Perguruan Tinggi**

**Tabel 3.5 Tabel *Use Case Scenario* Melihat Rangking Perguruan Tinggi**

|  |  |  |
| --- | --- | --- |
| **Nama *use case*** | Melihat Rangking Perguruan Tinggi | |
| **Aktor** | Administrator, Visitor | |
| **Deskripsi** | Pada *use case* ini dua aktor ingin melihat rangking  dari perguruan tinggi. | |
| **Prakondisi** | Nilai tiap aspek sudah didapat | |
| **Alur Utama** | **Kegiatan Aktor** | **Respon Sistem** |

|  |  |  |
| --- | --- | --- |
| 1.Melihat Rangking PerguruanTinggi | 1.Memilih menu Rangking | 1. Menampilkan HalamanRangking 2. Menampilkan data Rangking Perguruan Tinggi |
| **Alur Alternatif** | Data tidak dapat dirangking jika aspek-aspek yang  digunakan untuk perangkingan belum dipenuhi | |
| **Kesimpulan** | Aktor mengetahui hasil rangking dari perguruan  tinggi | |
| **Post Kondisi** | Perguruan tinggi berhasil dirangking | |

* + - 1. *Logout*



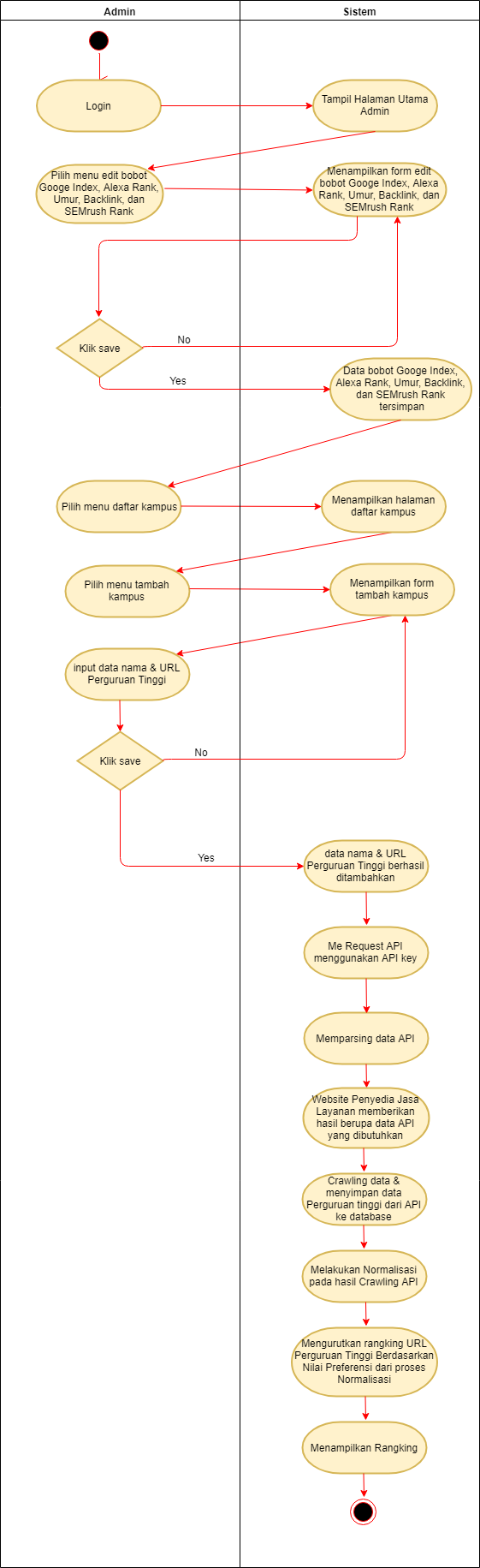
**Gambar 3.7 Diagram Usecase *Logout***

## Tabel 3.6 Tabel Use Case Scenario Logout

|  |  |  |
| --- | --- | --- |
| **Nama *use case*** | *Logout* | |
| **Aktor** | Administrator | |
| **Deskripsi** | *Use case* menggambarkan kegiatan keluar dari sistem | |
| **Prakondisi** | Aktor berada dalam sistem | |
| **Alur Utama** | **Kegiatan Aktor** | **Respon Sistem** |
| 1. Memilih menu  *Logout.* | 1. Mengirimkan perintah ke database untuk memutus akses Admin kedalam sistem. 2. Menampilkan Halaman utama bukan sebagai   admin. |
| **Kesimpulan** | Aktor keluar dari sistem | |
| **Post Kondisi** | Berhasil keluar dari sistem | |

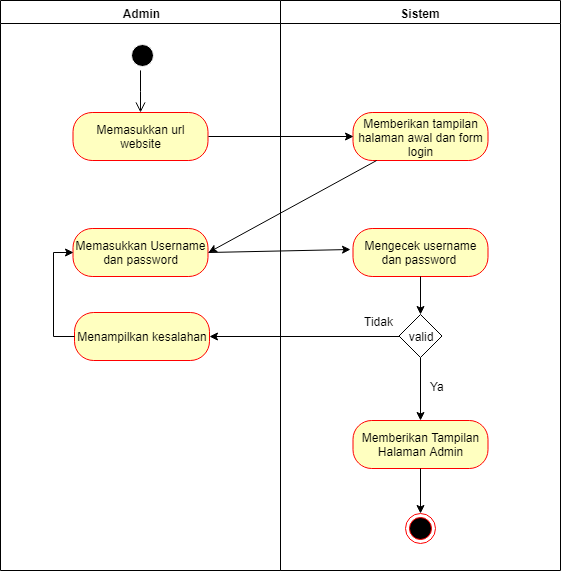
* + 1. **Diagram*Activity***

Diagram *Activity* akan menggambarkan spesifikasi sistem sacara keseluruhan. Diagram *Activity* juga akan menjelaskan bagaimana alur sebuah sistem tersebut berjalan.

1. DiagramPerangkingan

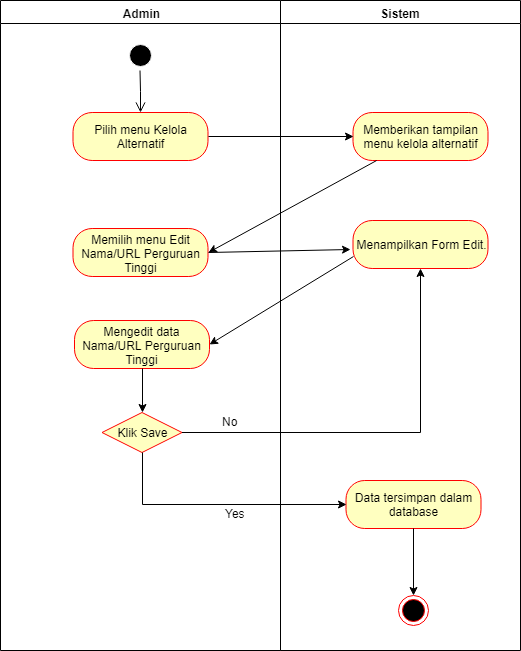
**Gambar 3.8 Diagram Activity Perangkingan**

1. Diagram *Activity Login*



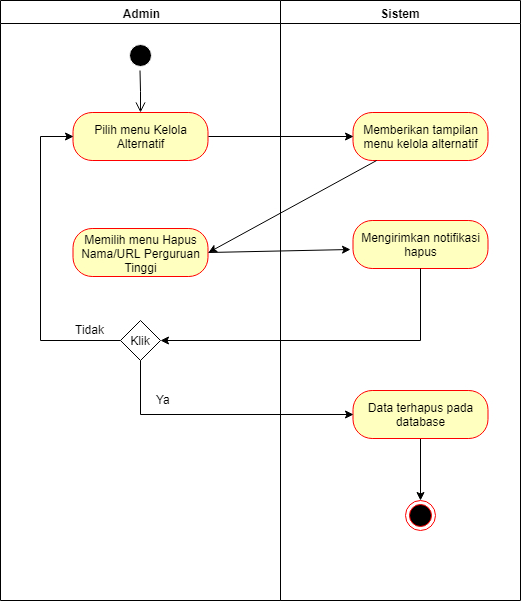
**Gambar 3.9 Diagram *Activity Login***

1. Diagram *Activity* Edit Nama/URL Perguruan Tinggi



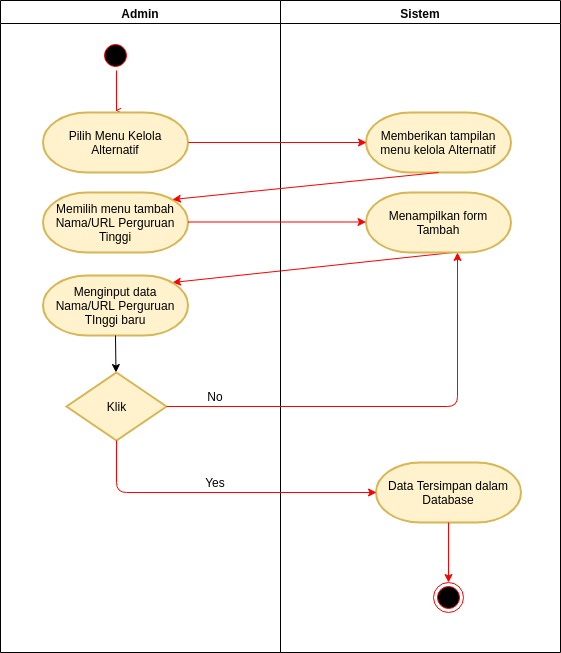
**Gambar 3.10 Diagram *Activity Edit Nama/URL* Perguruan Tinggi**

1. Diagram *Activity* Hapus Nama/URL PerguruanTinggi



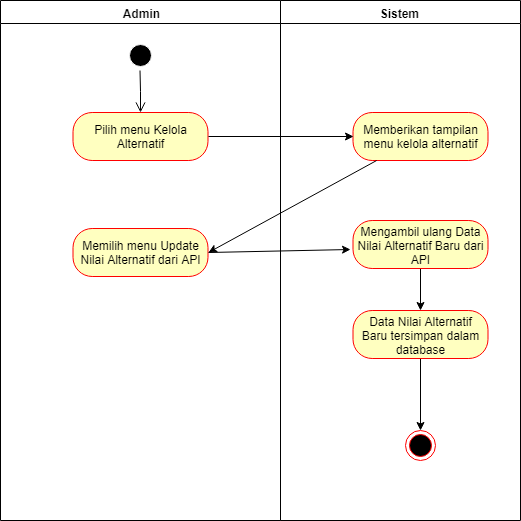
**Gambar 3.11 Diagram *Activity* Hapus Nama/URL Perguruan Tinggi**

1. Diagram *Activity* Tambah Nama/URL Perguruan Tinggi



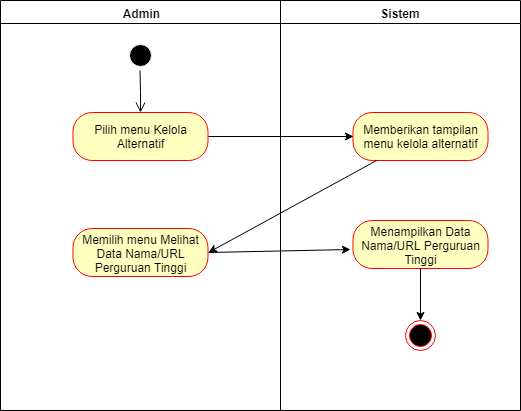
**Gambar 3.12 Diagram Tambah Nama/URL Perguruan Tinggi**

1. Diagram *Activity* Update Nilai Alternatif dari API



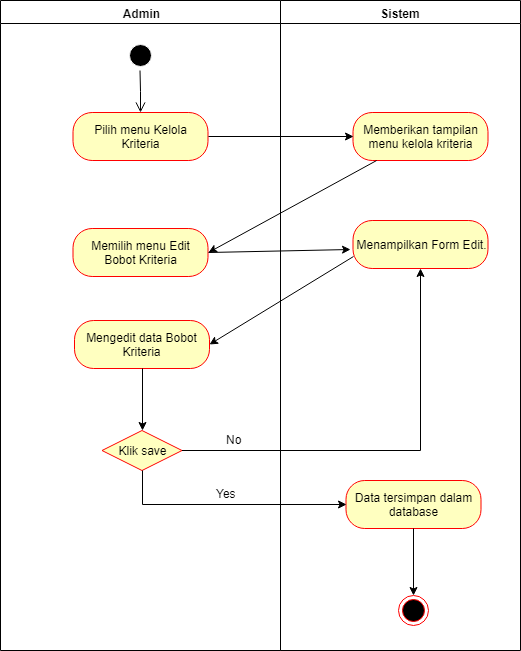
**Gambar 3.13 Diagram *Activity Update* Nilai Alternatif dari API**

1. Diagram *Activity* Melihat Data Nama/URL Perguruan Tinggi



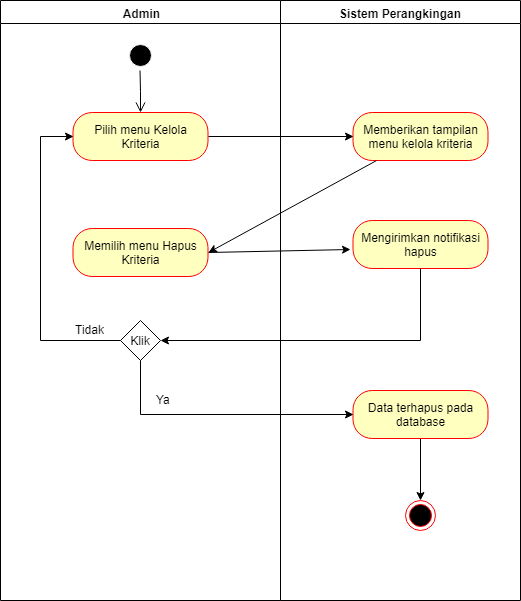
**Gambar 3.14 Diagram *Activity* Melihat Data Nama/URL Perguruan Tinggi**

1. Diagram *Activity* Edit Bobot Kriteria



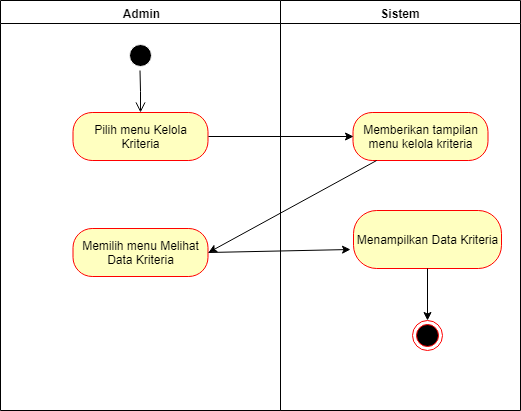
**Gambar 3.15 Diagram *Activity* Edit Bobot Kriteria**

1. Diagram *Activity* Hapus Kriteria



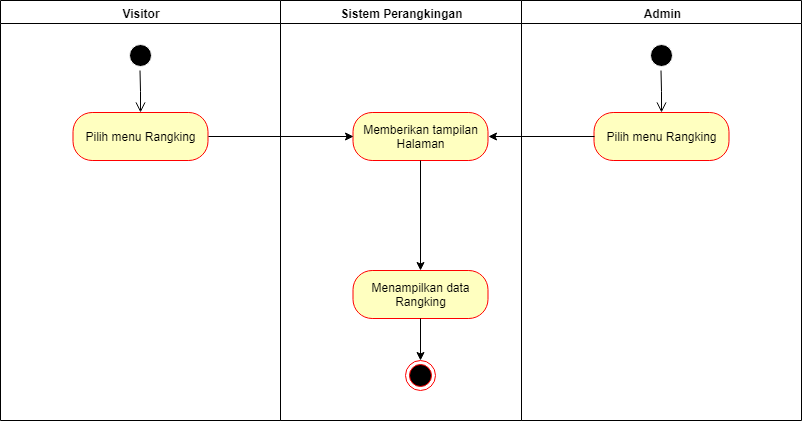
**Gambar 3.16 Diagram *Activity* Hapus Kriteria**

1. Diagram *Activity* Melihat Data Kriteria



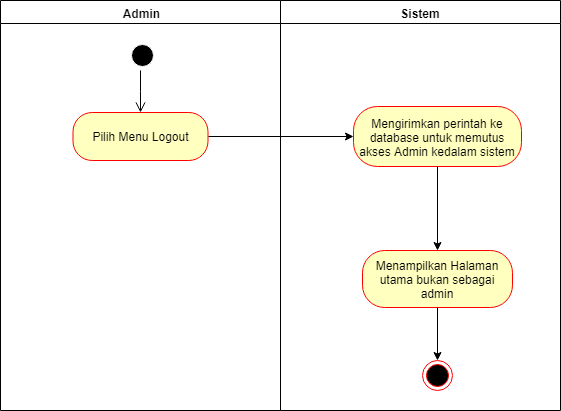
**Gambar 3.17 Diagram *Activity* Melihat Data Kriteria**

1. Diagram *Activity* Melihat Rangking Perguruan Tinggi



**Gambar 3.18 Diagram *Activity* Melihat Rangking Perguruan Tinggi**

1. Diagram *Activity Logout*

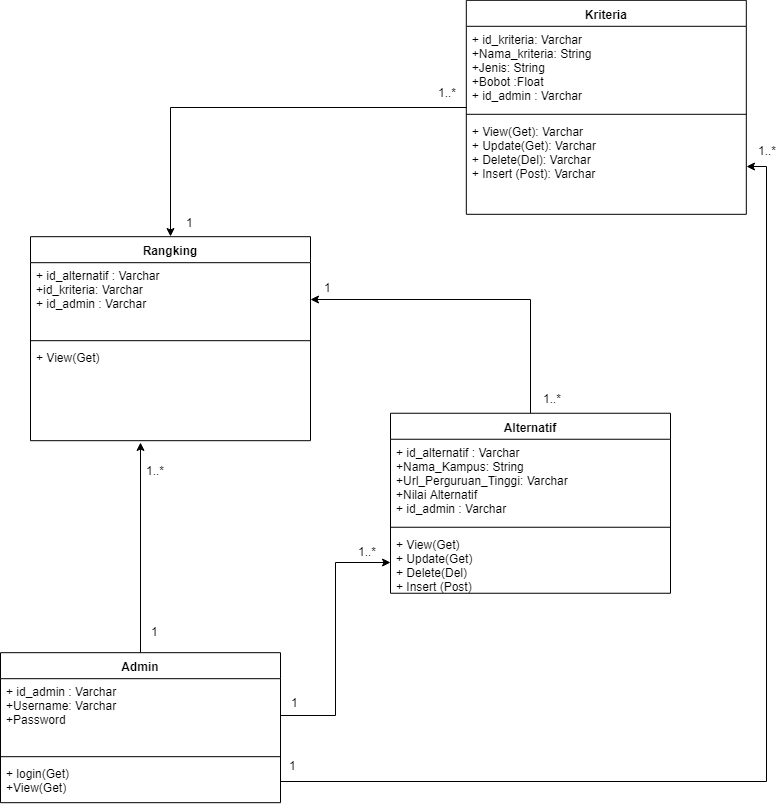


**Gambar 3.19 Diagram *Activity Logout***

* + 1. **Diagram *Class***

Terdapat 5 class yakni Rangking, Kriteria, Alternatif, Admin dan Login.

Berikut selengkapnya dapat dilihat pada gambar (Gambar 3.20):

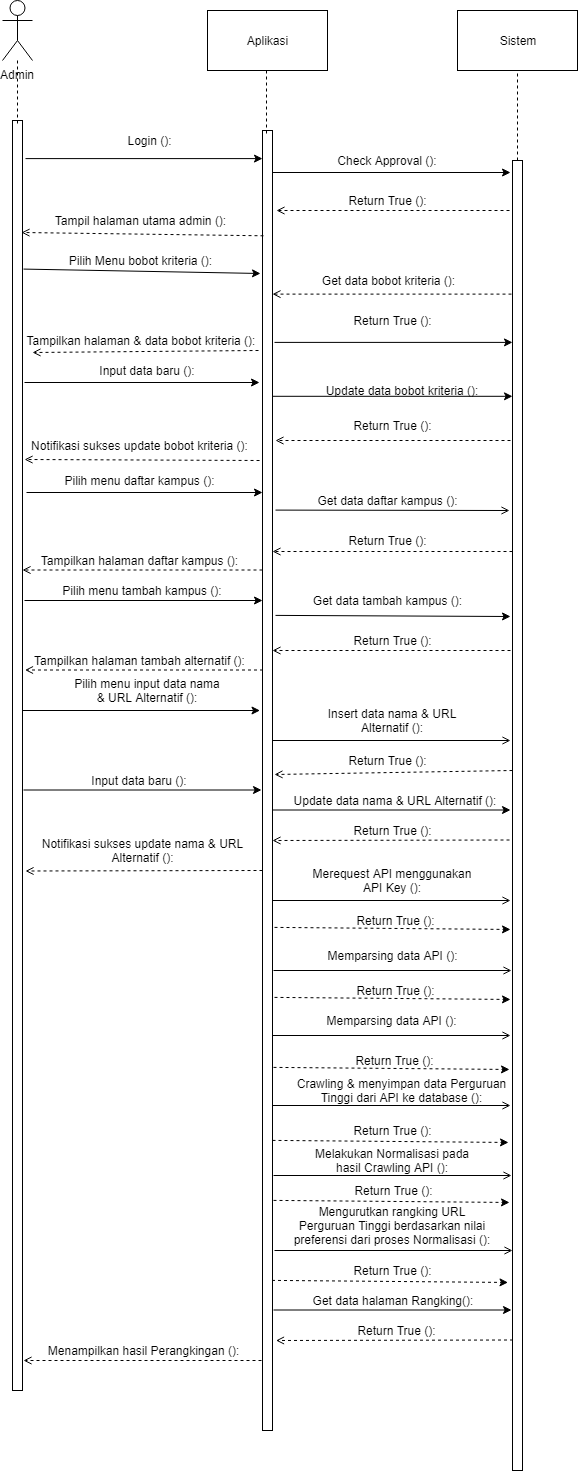


**Gambar 3.20 Diagram *Class***

* + 1. **Diagram *Sequence***

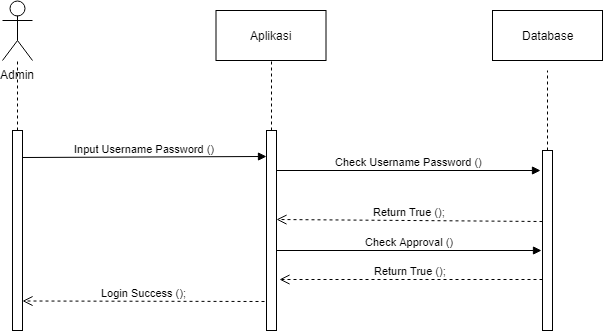
Diagram *Sequence* adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah objek kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek pada sistem.

* + - 1. **Diagram *Sequence* Perangkingan**



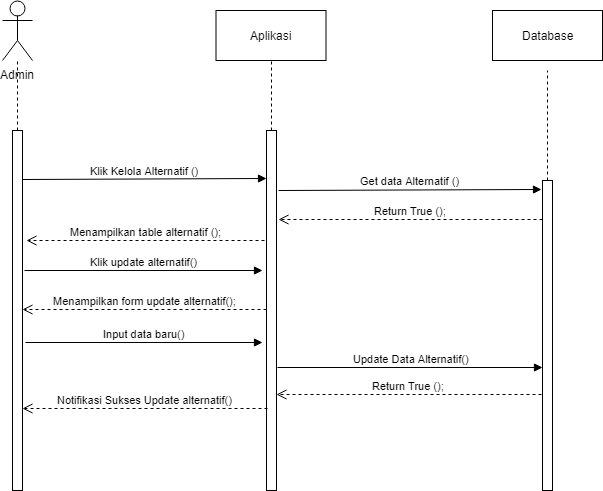
**Gambar 3.21 Diagram *Sequance* Perangkingan**

* + - 1. **Diagram *Sequance Login***



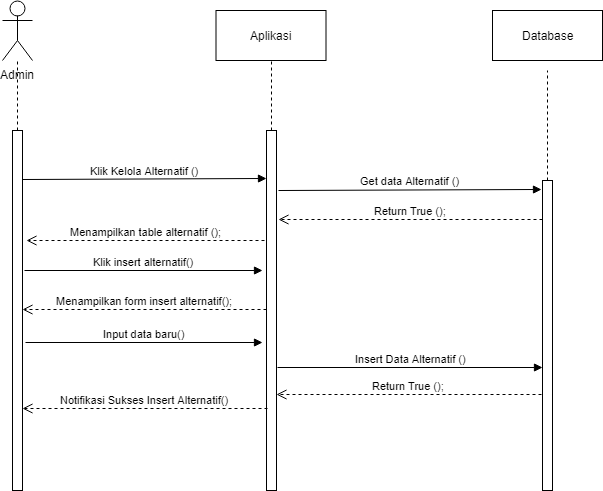
**Gambar 3.22 Diagram *Sequance Login***

## Diagram *Sequance* Edit Nama/URL Perguruan Tinggi



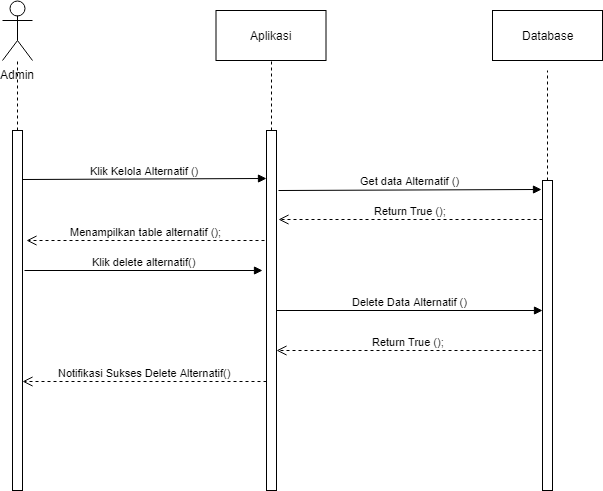
**Gambar 3.23 Diagram *Sequance* Edit Nama/URL Perguruan Tinggi**

## Diagram *Sequance* Tambah Nama/URL Perguruan Tinggi



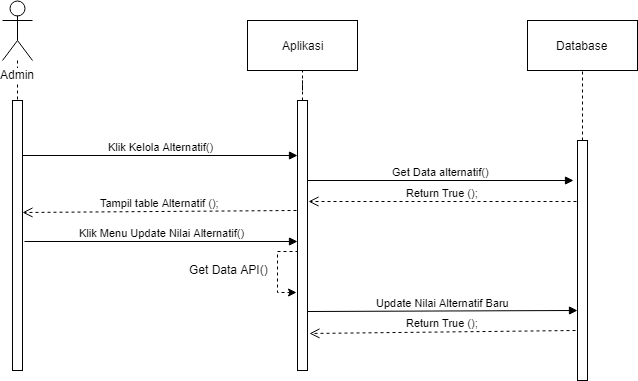
**Gambar 3.24 Diagram *Sequance* Tambah Nama/URL Perguruan Tinggi**

## Diagram *Sequance* Hapus Nama/URL Perguruan Tinggi



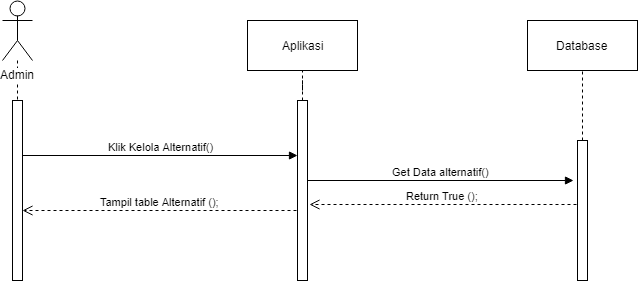
**Gambar 3.25 Diagram *Sequance* Hapus Nama/URL Perguruan Tinggi**

* + - 1. **Diagram *Sequance Update* Nilai Alternatif dari API**



**Gambar 3.26 Diagram *Sequance Update* Nilai Alternatif dari API**

## Diagram *Sequance* Melihat Nama/URL Perguruan Tinggi



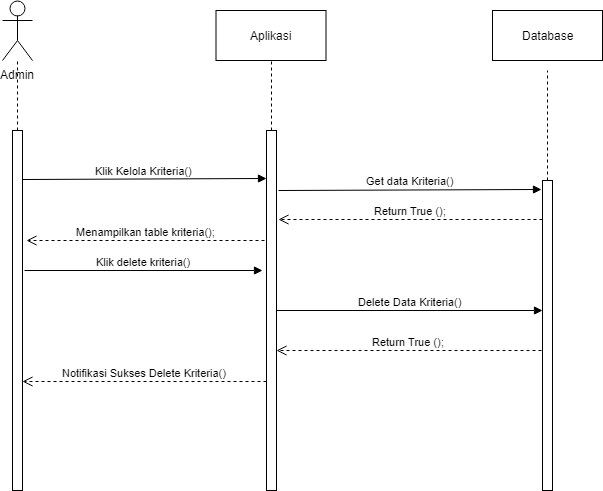
**Gambar 3.27 Diagram *Sequance* Melihat Nama/URL Perguruan Tinggi**

* + - 1. **Diagram *Sequance* Edit Bobot Kriteria**



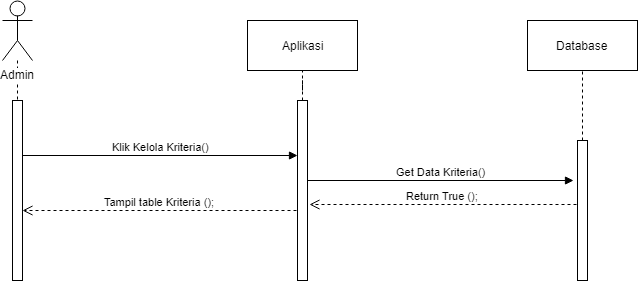
**Gambar 3.28 Diagram Edit Bobot Kriteria**

* + - 1. **Diagram *Sequance* Hapus Kriteria**



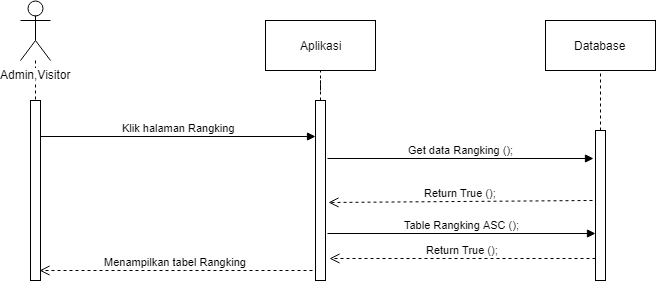
**Gambar 3.29 Diagram *Sequance* Hapus Kriteria**

* + - 1. **Diagram *Sequance* Melihat Data Kriteria**



**Gambar 3.30 Diagram *Sequance* Melihat Data Kriteria**

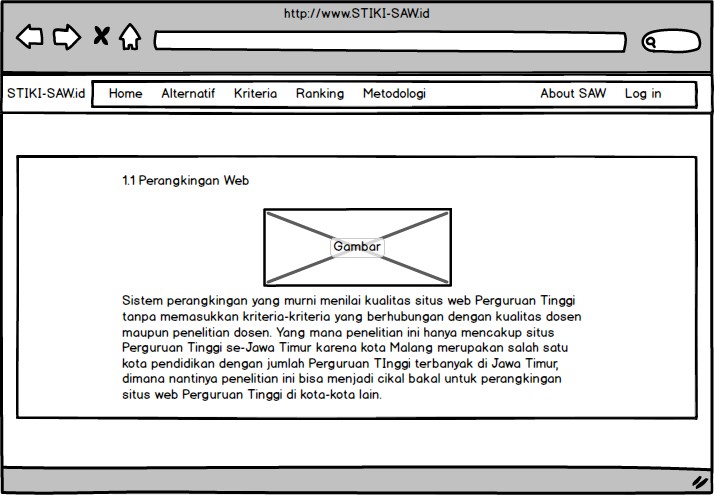
## Diagram *Sequance* Melihat Rangking PerguruanTinggi



**Gambar 3.31 Diagram *Sequance* Melihat Rangking Perguruan Tinggi**

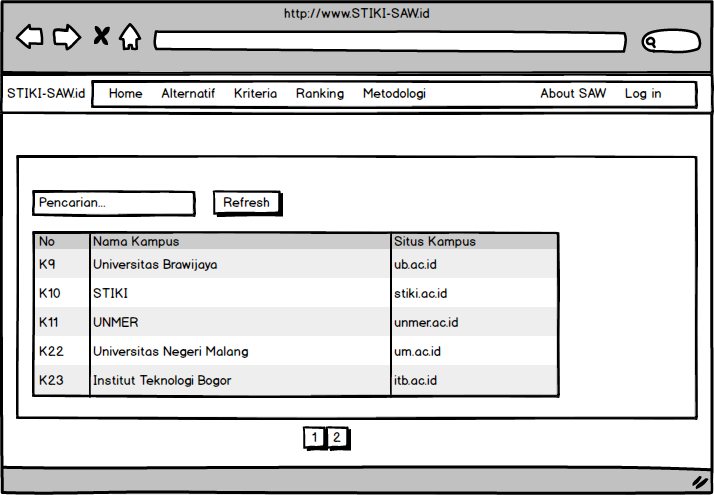
## Desain Interface

* + - 1. **Desain Halaman Home**



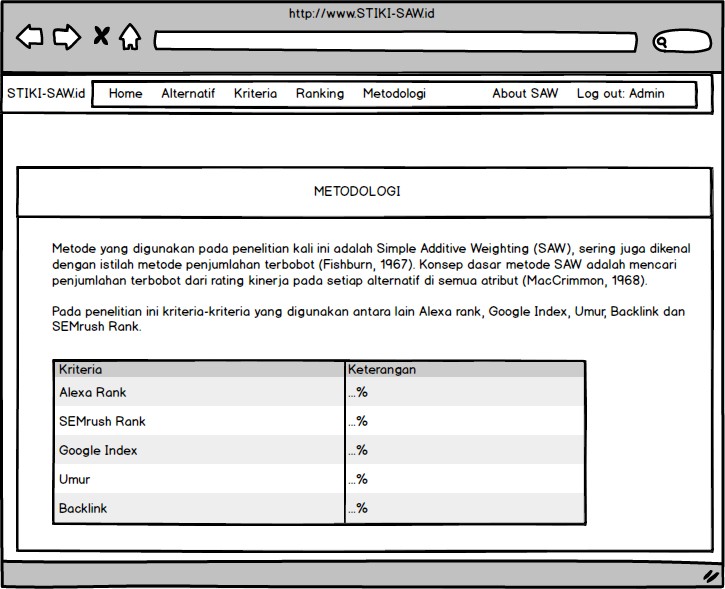
**Gambar 3.32 Desain Halaman Home**

## Desain Halaman Rangking (Visitor)



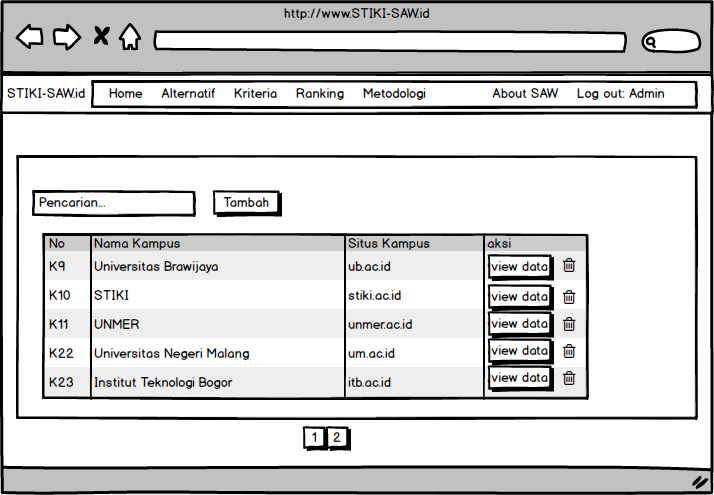
**Gambar 3.33 Desain Halaman Rangking**

## Desain Halaman Metodologi



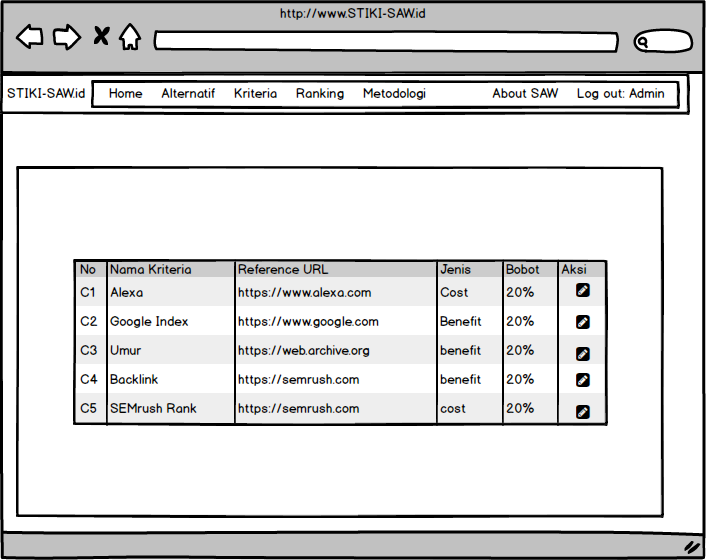
**Gambar 3.34 Desain Halaman Metodologi**

## Desain Halaman Kelola Alternatif (Admin)



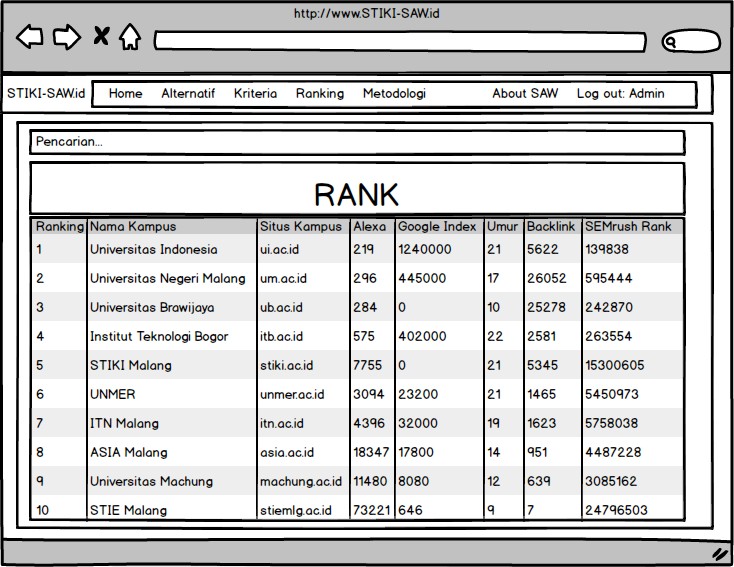
**Gambar 3.35 Desain Halaman Kelola Alternatif (Admin)**

## Desain Halaman Kriteria (Admin)



**Gambar 3.36 Desain Halaman Kelola Kriteria (Admin)**

## Desain Halaman Rangking (Admin)



**Gambar 3.37 Desain Halaman Rangking (Admin)**