**BAB III**

**METODE PENELITIAN**

* 1. **Metode Pengumpulan Data**

Metode Penelitian adalah penulisan yang difokuskan pada bagaimana penelitian dilaksanakan agar tujuan atau masalah penelitian dapat dijawab. Metode yang digunakan dalam penyusunan penelitian ini adalah metode penulisan dan metode perancangan program yang dapat diuraikan sebagai berikut:

1. Studi Pustaka

Pengumpulan data dengan melakukan pengamatan secara langsung terhadap objek penelitian. Dengan mencatat hal-hal penting yang berhubugan dengan judul laporan sehingga memperoleh data yang lengkap dan lebih akurat.

1. Observasi

Observasi yang akan dilakukan dengan cara menganalisis dan mengadakan pengamatan secara langsung terhadap masyarakat yang mau memberikan informasi terkait pandemik *Covid-19* kepada penulis untuk di lakukan pendataan dan memberikan titik lokasi masyarakat yang terjangkit *Covid-19* dan tidak terjangkit *Covid-19*.

1. Wawancara

Metode yang digunakan untuk mendapatkan data bagi keperluan penulisan ini dengan melakukan wawancara secara langsung dengan masyarakat pada beberapa tempat di Kota Kupang.

**3.2. Bahan dan Alat Penelitian**

**3.2.1. Bahan Penelitian**

Dalam melakukan penelitian ini ada beberapa bahan yang digunakan penulis untuk mendukung terjadinya penelitian yaitu, data-data mengenai masyarakat yang telah di vaksin dan belum di vaksin serta lokasi masyarakat yang terjangkit covid-19 di Kota Kupang.

**3.2.2. Alat Penelitian**

Peralatan yang digunakan dalam penelitian ini yaitu perangkat keras (*Hardware*) dan perangkat lunak (*Software*) sebagai berikut :

1. Perangkat Keras

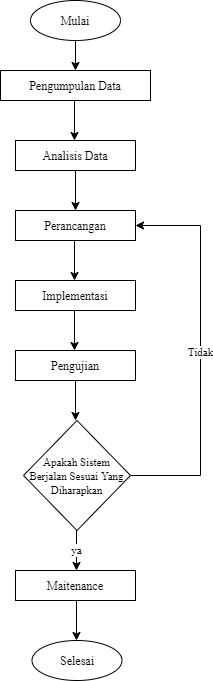
1. Laptop Lenovo Thinkpad S340
2. Procesor AMD Ryzen 3 3200U
3. Ram 12 GB DDR4
4. Harddisk SSD 512 GB

2. Perangkat Lunak

1. *Windows* 10 64-bit
2. *Microsoft Word* 2019
3. *StarUnifled Modeling Language* (UML) 3.2
4. Visual Studio 2010
5. Xampp *Control panel*
6. *Google Chrome*

**3.3. Prosedur Penelitian**

Didalam prosedur penelitian ini terdapat beberapa tahap yang dilakukan peneliti dalam melakukan pengumpulan data hingga selesai. Penilitian ini digunakan sebagai pedoman dalam pelaksanaan penelitian ini agar hasil yang dicapai tujuan yang telah ditentukan sebelumnya. Berikut *Flowchart* tahap penelitian :



Gambar 1. Flowchart Penelitian

Berdasarkan alur *flowchart* di atas, maka dapat dijelaskan sebagai berikut:

1. Pada tahap ini dengan melakukan pengumpulan semua data-data atau informasi yang berkaitan dengan penelitian. Pengumpulan data ini mengenai data masyarakat di Kota Kupang dan informasi yang di ambil dari lokasi tempat penelitian.

2. Pada tahap ini dengan melakukan pengumpulan data menjadi informasi terkait data masyarakat terjangkit *Covid-19* dan bebas *Covid-19*, sehingga karakteristik data tersebut bisa dipahami dan bermanfaat untuk solusi permasalahan terutama masalah yang berkaitan dengan penelitian mengenai permasalahan membangun aplikasi *Health Treacking* perjalanan dimasa pandemik *Covid-19* berbasis *website mobile*.

3. Perancangan

Pada tahap ini dibuat berdasarkan hasil analisa kebutuhan yang akan dilakukan dan digunakan untuk membangun aplikasi *Health Treacking* perjalanan dimasa pandemik *Covid-19* berbasis *website mobile* sebagai media implementasinya.

4. Implementasi

Pada tahap ini perancangan yang telah dibuat diimplementasikan kedalam program sehingga sesuai dengan perancangan.

5. Pengujian

Pada tahap ini dilakukan pengujian kembali kepada aplikasi yang telah dijalankan. Jika pengujian ini hasilnya benar maka akan lanjut, jika hasilnya salah maka akan kembali ke perancangan dan implementasi.

6. Hasil

Hasil dari penelitian ini adalah sebuah aplikasi *Health Treacking* perjalanan dimasa pandemik *Covid-19* berbasis *website mobile* ini mampu memberikan informasi lokasi tempat masyarakat yang terjangkit *Covid-19* dan bebas *Covid-19* di Kota Kupang, memberikan arahan kepada masyarakat, dan mampu memberikan informasi berkaitan dengan apa saja yang dapat dilihat dalam lokasi traking *Covid-19* di Kota Kupang.

# 3.4. Analisis Data

**3.4.1. Perspektif Produk**

Aplikasi *Health Treacking* perjalanan dimasa pandemik *Covid-19* berbasis *website mobile* adalah sebuah *website* yang dapat mempermudah pengguna atau masyarakat dalam pelacakan masyarakat yang terpapar *Covid-19*. Sistem ini menyajikan informasi yang berkaitan dengan jumlah masyarakat yang terpapar *Covid-19* dan yang tidak terpapar *Covid-19* di Kota Kupang beserta petanya untuk menunjukkan lokasi tempat tinggal masyarakat yang terpapar. Sistem ini dibuat mengunakan bahasa pemrograman PHP, Javascript dan database SQL.

**3.4.2. Fungsi Sistem**

Adapun fungsi produk yang terdapat pada *website Health Treacking* perjalanan dimasa pandemik *Covid-19* adalah sebagai berikut:

1. Fungsi Pengguna
2. Melihat titik lokasi masyarakat yang terpapar di Kota Kupang
3. Mencari tempat yang bebas dari masyarakat yang terpapar di Kota Kupang
4. Melihat jumlah penduduk yang terpapar di Kota Kupang
5. Mendaftarkan data diri untuk memenuhi syarat-syarat bebas *Covid-19*
6. Fungsi Admin
7. Melakukan Login
8. Melihat data pengguna
9. Mengelolah tempat terpapar *Covid-19*
10. Mengelolah data penduduk berdasarkan kelurahan

**3.4.3. Karakteristik Pengguna**

Pengguna sistem informasi ini adalah semua orang, khususnya orang-orang yang ingin mencari lokasi bebas *Covid-19* dan terpapar *Covid-19* di Kota Kupang. Pengguna juga dapat berupa pengguna dengan tingkat otorisasi khusus *(administrator)* yang mempunyai wewenang untuk melakukan modifikasi pada informasi dalam database yang akan ditampilkan.

**3.4.4. Kebutuhan Antarmuka Eksternal**

1. Antarmuka pengguna

Pengguna dapat mengoperasikan sistem ini menggunakan *keyboard, mouse, touchpad,* dan *touchscreen* dengan sistem operasi windows, android, dan lain sebagainya.

1. Antarmuka perangkat lunak

Kebutuhan perangkat lunak yang digunakan untuk membuat sistem ini diantaranya adalah:

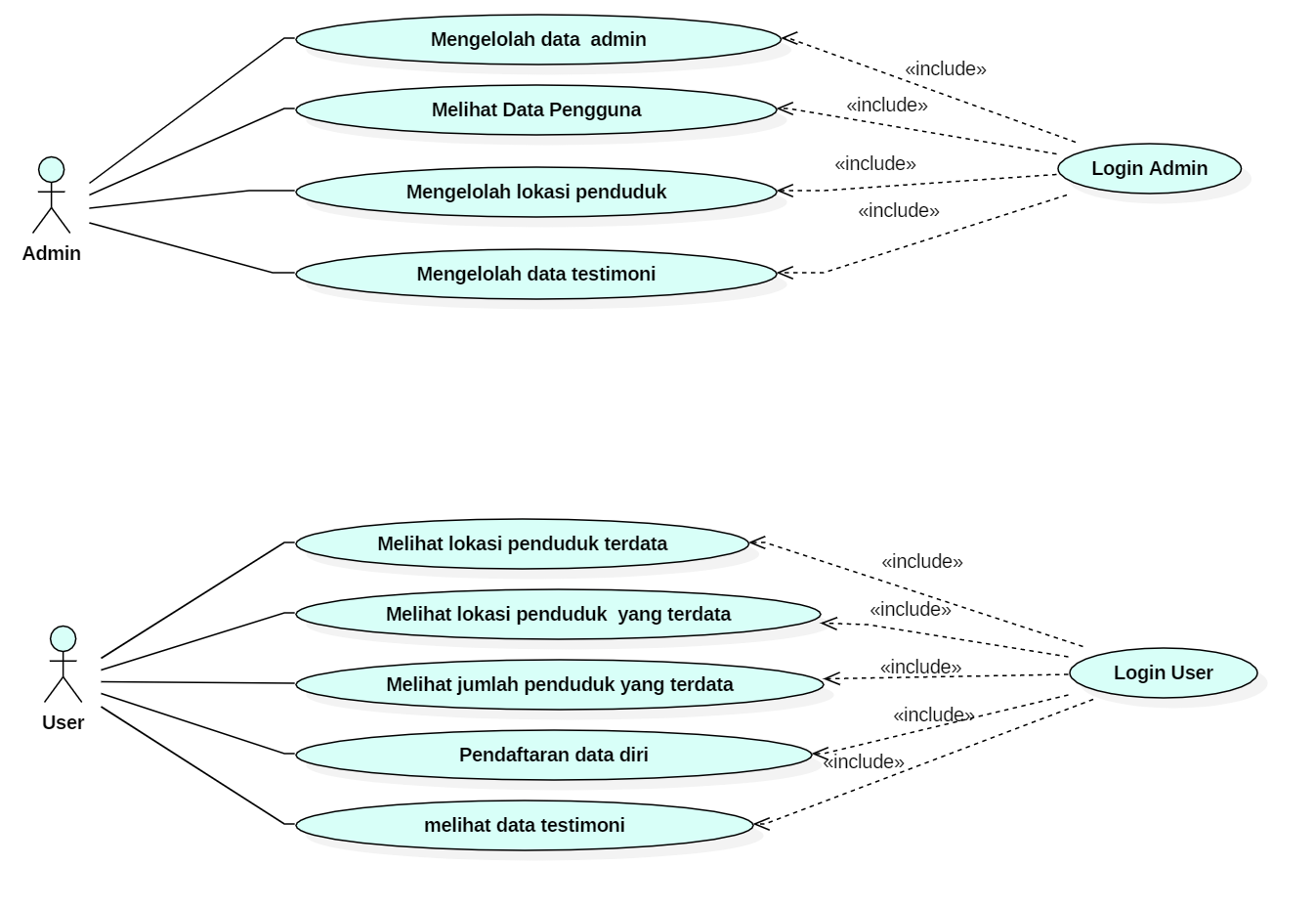
1. *Microsoft Windows* 10 64 bit sebagai operating *system*
2. *Xampp* sebagai aplikasi web server dan database
3. *Visual Studio Code* sebagai aplikasi untuk membangun *website*
4. *Chrome* sebagai aplikasi untuk menjalankan *website*
5. DrawIO sebagai aplikasi untuk mendesain perancangan sistem
6. Antarmuka perangkat keras

Adapun perangkat keras yang digunakan untuk membangun sistem ini adalah laptop tipe AMD Ryzen 3 3200U dengan RAM 12GB DDR4 serta SSD 512GB.

**3.4.5. Kebutuhan Fungsional**

**1. *Use Case Diagram***

*Use case diagram* adalah gambaran fungsionalitas dari suatu sistem. Pada *use case* aplikasi ini, pengguna dapat berinteraksi dengan sistem secara langsung sedangkan administrator harus melakukan login sebelum berinteraksi atau mengelola data yang ada pada sistem.



Gambar 2. *Use Case Diagram*

1. Spesifikasi *Use Case* Admin
2. Spesifikasi *use case:* Mengelolah data admin

Tabel 5. Spesifikasi data admin

|  |  |
| --- | --- |
| *Use case name* | Mengelolah data admin |
| *Brief description* | *Use case* ini digunakan admin untuk mengelola data admin |
| *Actor* | Admin |
| *Basic flow* | 1. Admin meminta sistem menampilkan halaman mengelola data admin   A-1: Admin memilih mengubah data admin   1. Sistem memperbaharui basis data admin 2. *Use case* selesai |
| *Alternative flow* | A-1: Admin memilih mengubah data kategori data admin   1. Admin meminta sistem menampilkan antarmuka ubah data admin 2. Sistem menampilkan antarmuka ubah data admin 3. Admin memasukan data yang akan diubah 4. Admin meminta sistem menyimpan data   E-1 : Data masih ada yang kosong   1. Kembali ke basic flow langkah ke tiga |
| *Error flow* | E-1 : Data masih ada yang kosong   1. Sistem memunculkan pesan bahwa masih ada data yang kosong 2. Kembali ke basic flow langkah ke satu |
| *Pre Condition* | Admin telah memasuki sistem |
| *Post Condition* | Data kategori wisata pada basis data telah diperbaharui |
| *Include* | Login |
| *Extend* | None |

1. Spesifikasi *use case:* Melihat data pengguna

Tabel 6. Spesifikasi Melihat data pengguna

|  |  |
| --- | --- |
| *Use case name* | Melihat data pengguna |
| *Brief description* | *Use case* ini digunakan untuk melihat data pengguna |
| *Actor* | Admin |
| *Basic flow* | 1. Admin meminta sistem menampilkan halaman data pengguna 2. *Use case* selesai |
| *Pre Condition* | Admin telah memasuki system |
| *Include* | Login |
| *Extend* | None |

1. Spesifikasi *use case:* Mengelolah lokasi penduduk

Tabel 7. Spesifikasi Mengelolah lokasi penduduk

|  |  |
| --- | --- |
| *Use case name* | Mengelolah lokasi penduduk |
| *Brief description* | Use case ini digunakan untuk mengelola lokasi penduduk |
| *Actor* | Admin |
| *Basic flow* | 1. Admin meminta sistem menampilkan halaman mengelola lokasi penduduk   A-1: Admin memilih mengubah lokasi penduduk   1. Sistem memperbaharui basis data 2. Use case selesai |
| *Error flow* | E-1 : Data masih ada yang kosong   1. Sistem memunculkan pesan bahwa masih ada data yang kosong 2. Kembali ke basic flow langkah ke satu |
| *Pre Condition* | Admin telah memasuki sistem |
| *Post Condition* | data lokasi telah diperbaharui |
| *Include* | Login |
| *Extend* | None |

1. Spesifikasi *Use Case* User
2. Spesifikasi *use case:* Melihat lokasi penduduk terdata

Tabel 8. Spesifikasi Melihat lokasi penduduk terdata

|  |  |
| --- | --- |
| *Use case name* | Melihat lokasi penduduk terdata |
| *Brief description* | *Use case* ini digunakan untuk melihat lokasi penduduk terdata |
| *Actor* | User |
| *Basic flow* | 1. User meminta sistem menampilkan halaman lokasi penduduk terdata 2. *Use case* selesai |
| *Pre Condition* | User telah memasuki system |
| *Include* | Login |
| *Extend* | None |

1. Spesifikasi *use case:* Melihat jumlah penduduk yang terdata

Tabel 9. Spesifikasi Melihat jumlah penduduk yang terdata

|  |  |
| --- | --- |
| *Use case name* | Melihat jumlah penduduk yang terdata |
| *Brief description* | *Use case* ini digunakan untuk melihat jumlah penduduk yang terdata |
| *Actor* | User |
| *Basic flow* | 1. User meminta sistem menampilkan halaman jumlah penduduk yang terdata 2. *Use case* selesai |
| *Pre Condition* | User telah memasuki system |
| *Include* | Login |
| *Extend* | None |

1. Spesifikasi *use case:* Pendaftaran dara diri

Tabel 10. Spesifikasi Pendaftaran dara diri

|  |  |
| --- | --- |
| *Use case name* | Pendaftaran dara diri |
| *Brief description* | *Use case* ini digunakan user untuk pendaftaran dara diri |
| *Actor* | User |
| *Basic flow* | 1. User meminta sistem menampilkan halaman pendaftaran dara diri   A-1: User memilih menambah data user   1. Sistem memperbaharui basis data user 2. *Use case* selesai |
| *Alternative flow* | A-1: User memilih mengubah data diri   1. User meminta sistem menampilkan antarmuka ubah data diri user 2. Sistem menampilkan antarmuka ubah data diri user 3. User memasukan data yang akan diubah 4. User meminta sistem menyimpan data   E-1 : Data masih ada yang kosong   1. Kembali ke basic flow langkah ke tiga |
| *Error flow* | E-1 : Data masih ada yang kosong   1. Sistem memunculkan pesan bahwa masih ada data yang kosong 2. Kembali ke basic flow langkah ke satu |
| *Pre Condition* | User telah memasuki sistem |
| *Post Condition* | Data pendaftara diri pada basis data telah diperbaharui |
| *Include* | Login |
| *Extend* | None |

1. Spesifikasi *use case:* Melihat testimoni

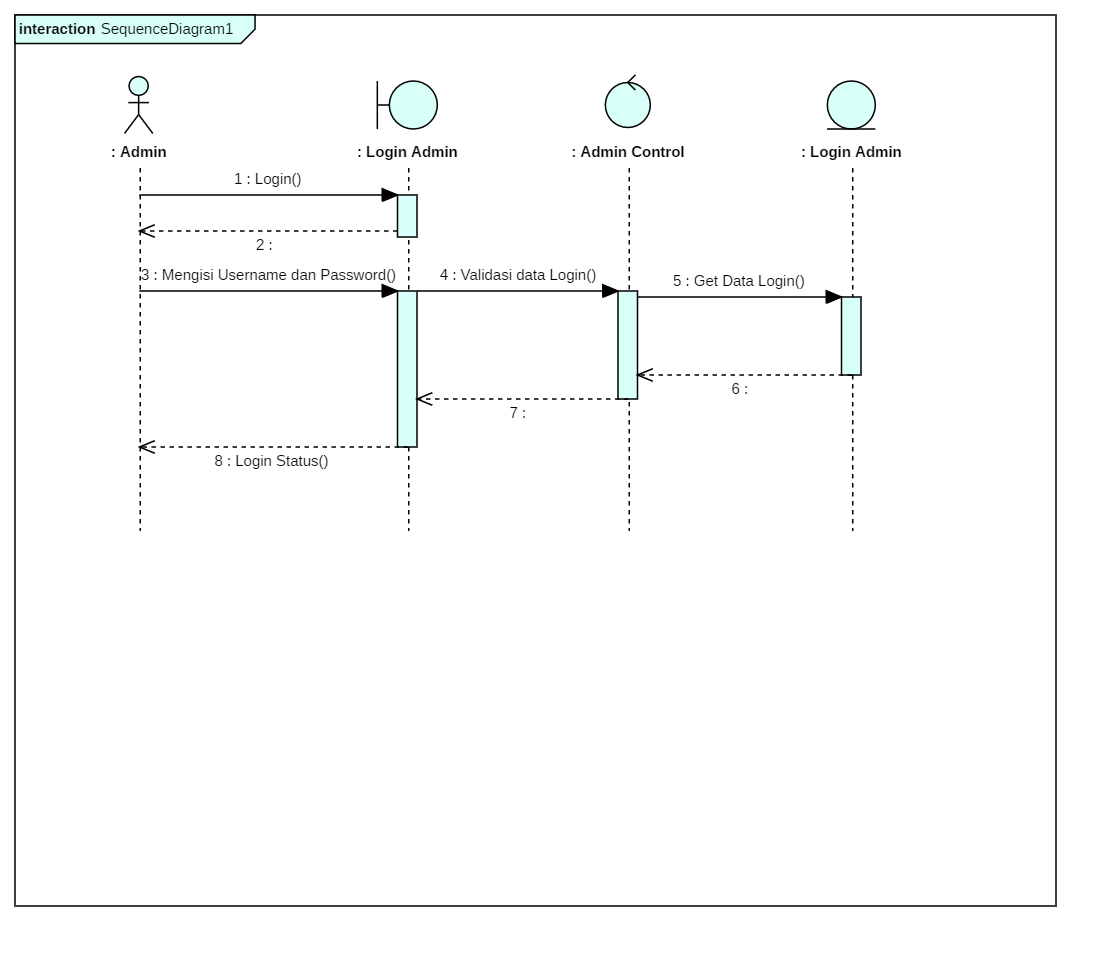
Tabel 11. Spesifikasi Melihat testimoni

|  |  |
| --- | --- |
| *Use case name* | Melihat testimoni |
| *Brief description* | *Use case* ini digunakan untuk melihat testimoni |
| *Actor* | User |
| *Basic flow* | 1. User meminta sistem menampilkan halaman testimoni 2. *Use case* selesai |
| *Pre Condition* | User telah memasuki system |
| *Include* | Login |
| *Extend* | None |

**2. *Sequence Diagram***

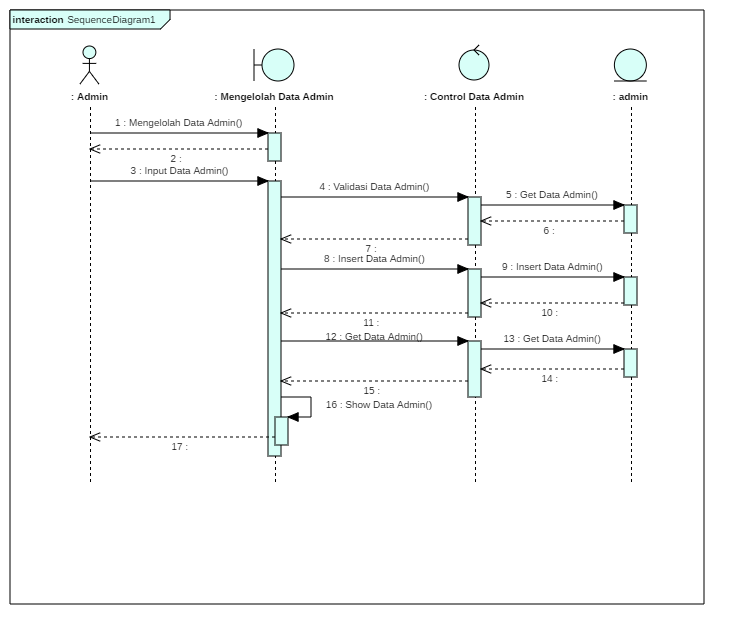
*Sequence diagram* adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah *object*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object.* Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

1) *Sequence diagram* login



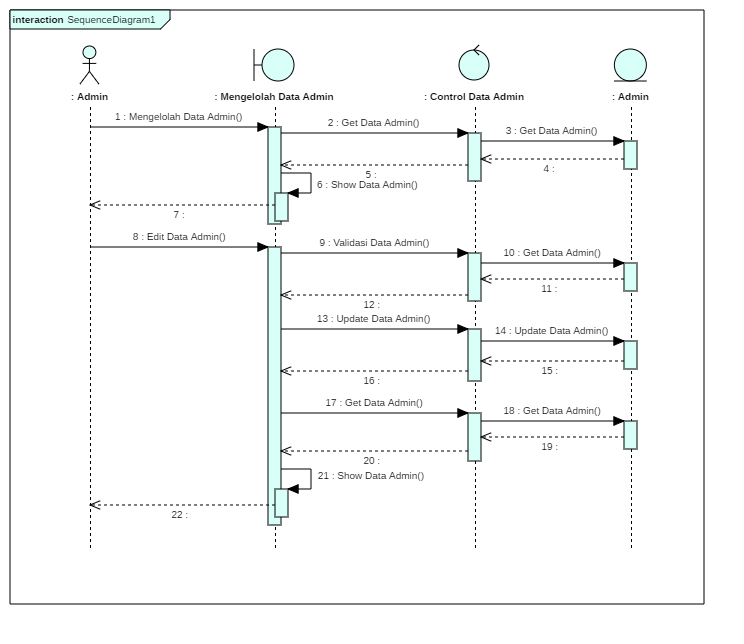
Gambar 3. *Sequence diagram login*

2) *Sequence diagram* tambahdata admin



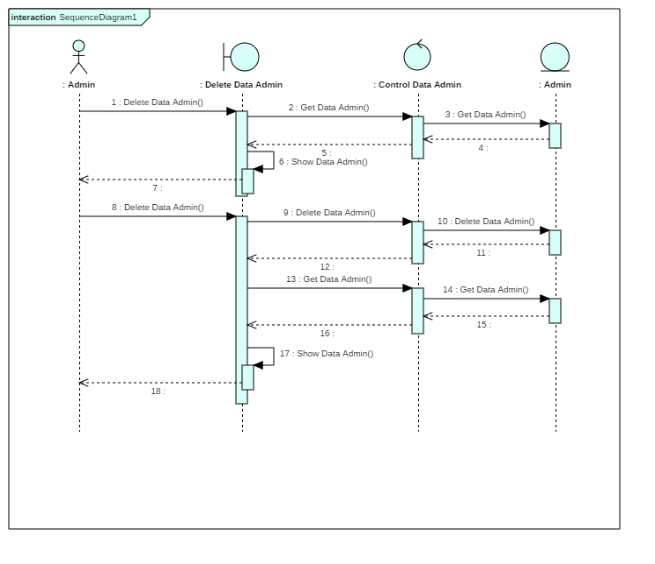
Gambar 4. *Sequence diagram tambah data admin*

3) *Sequence diagram* ubah data admin



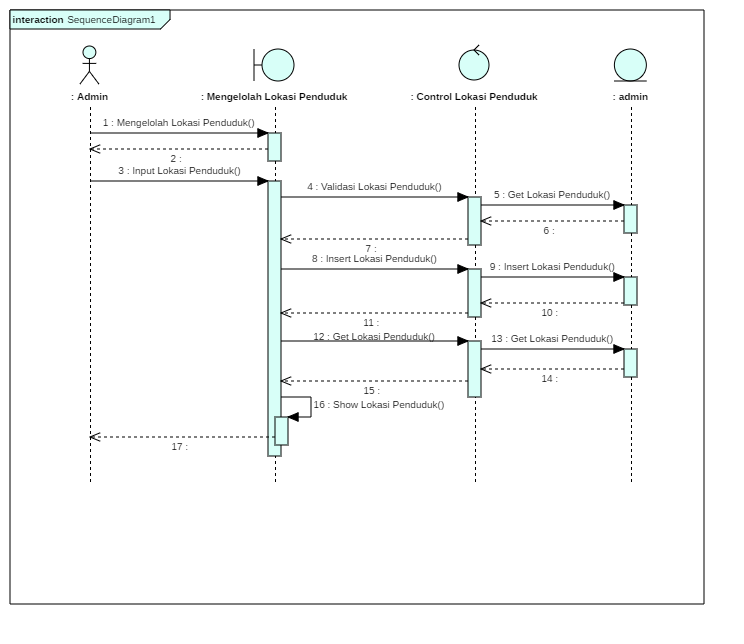
Gambar 5. *Sequence diagram ubah data admin*

4) *Sequence diagram* menghapus data admin



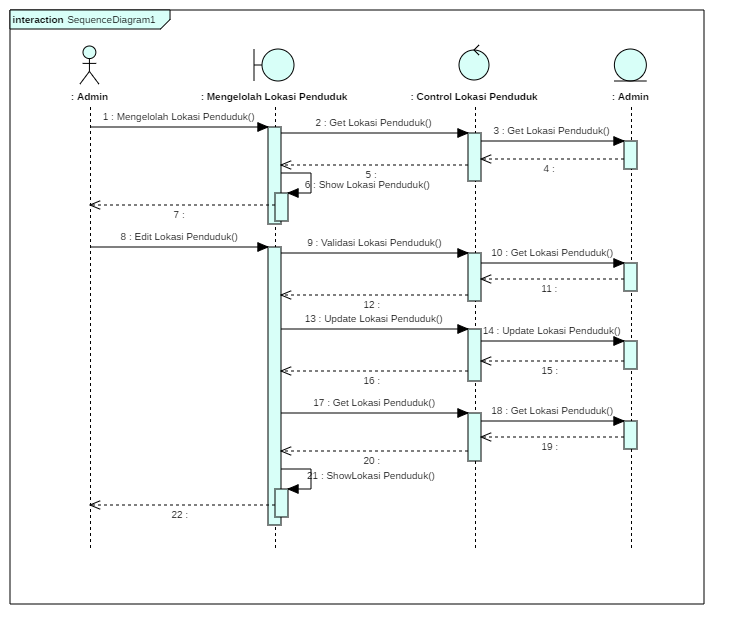
Gambar 6. *Sequence diagram hapus data admin*

5) *Sequence diagram* tambahlokasi pennduduk



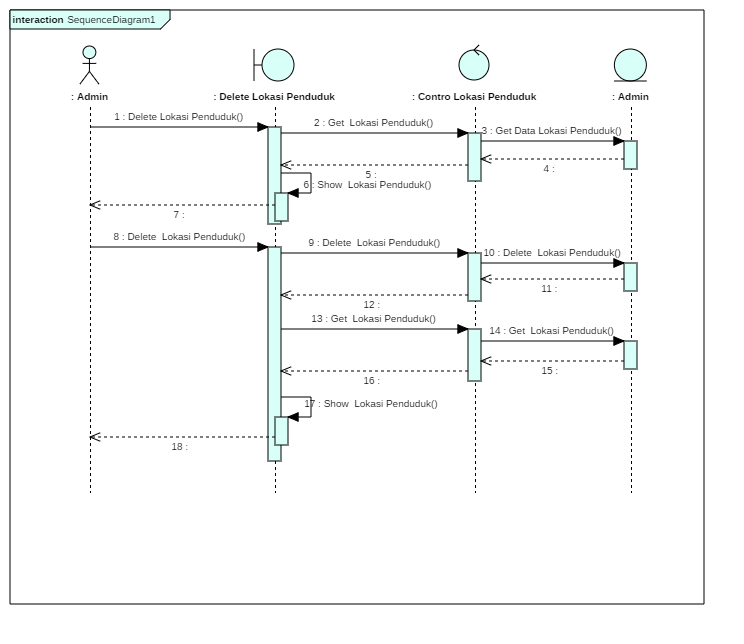
Gambar 7. *Sequence diagram tambah lokasi penduduk*

6) *Sequence diagram* ubahlokasi pennduduk



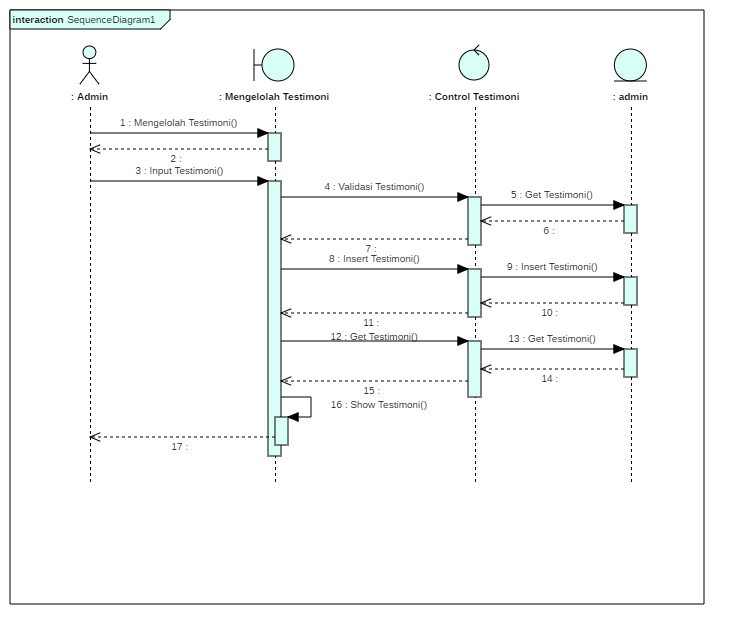
Gambar 8. *Sequence diagram ubah lokasi penduduk*

7) *Sequence diagram* hapuslokasi pennduduk



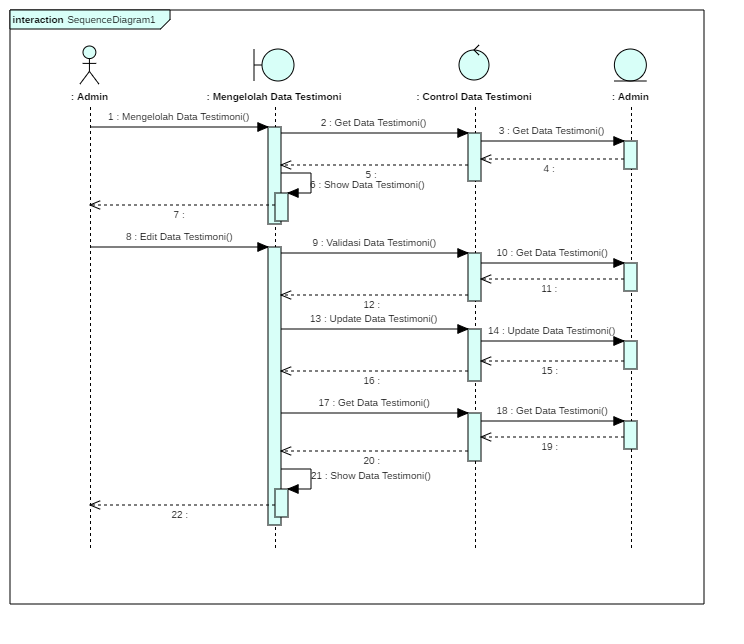
Gambar 9. *Sequence diagram hapus lokasi penduduk*

8) *Sequence diagram* tambahtestimoni



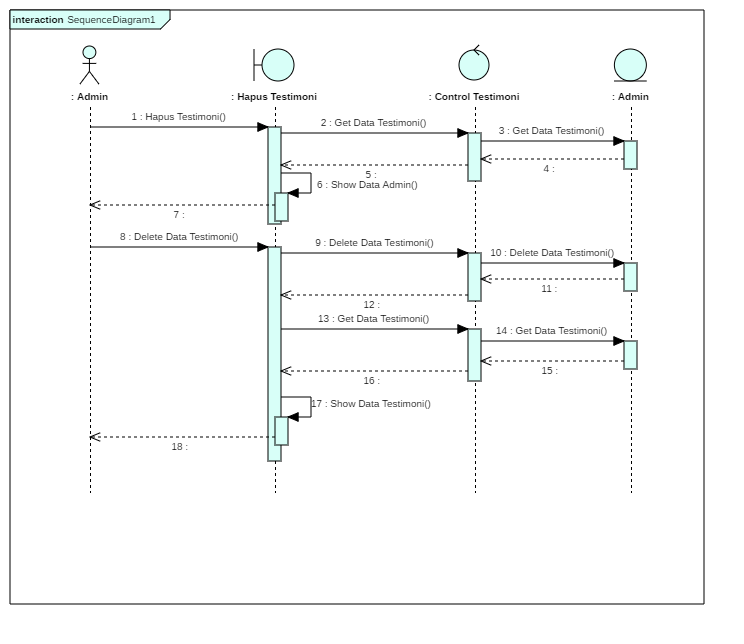
Gambar 10. *Sequence diagram tambah testimoni*

9) *Sequence diagram* ubah testimoni



Gambar 11. *Sequence diagram ubah testimoni*

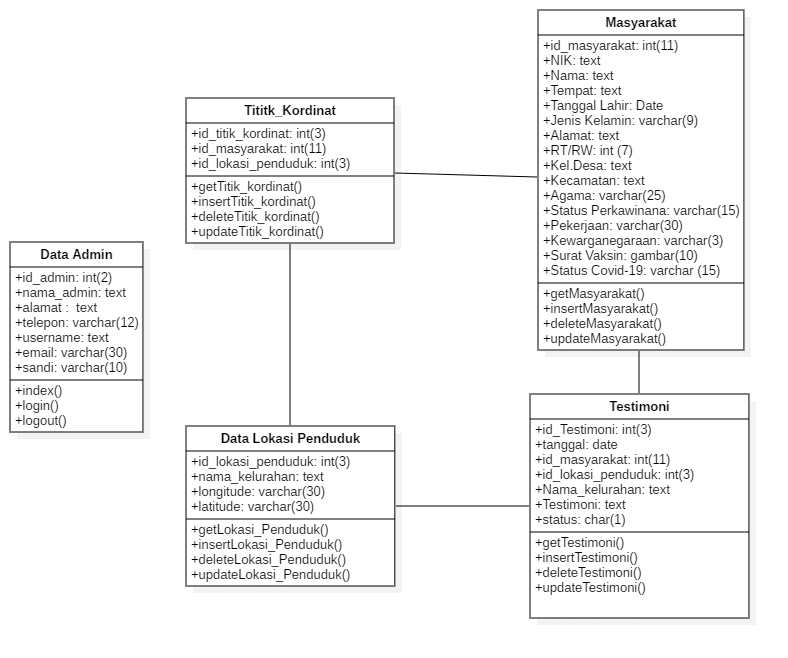
10) *Sequence diagram* hapus testimoni



Gambar 12. *Sequence diagram hapus testimoni*

**3. *Class Diagram***

*Class diagram* adalah model statis yang menggambarkan struktur dan deskripsi class serta hubungannya antara class. *Class diagram* mirip ER-Diagram pada perancangan database, bedanya pada ER-diagram tidak terdapat operasi/methode tapi hanya atribut. Class terdiri dari nama kelas, atribut dan operasi/methode.



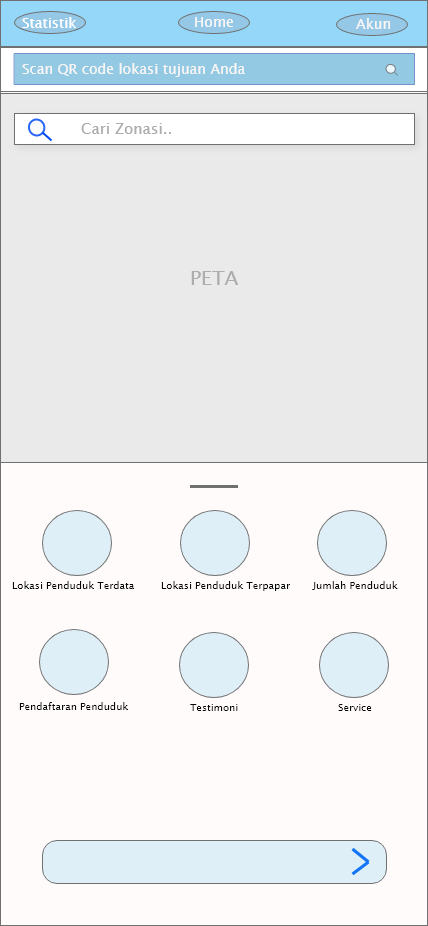
Gambar 13. *Class diagram* admin

**3.5. Perancangan Antar Muka**

Perancangan antar muka menggunakan pendekatan mobile website, website pengguna dirancang dengan menggunakan tampilkan mobile agar kompatibel saat diakses menggunakan perangkat mobile maupun desktop. Website admin memiliki tampilan desktop website agar memper mudah admin dalam memasukan berbagai macam data untuk diolah dan ditampilkan ke pengguna. Berikut merupakan perancangan antar muka website pengguna dan admin.

1. **Perancangan antar muka halaman beranda**

Beranda merupakan tampilan awal Ketika website diakses. Beranda memiliki beberapa menu dan fitur yang tersedia.



Gambar 3.1. Halaman antar muka Beranda

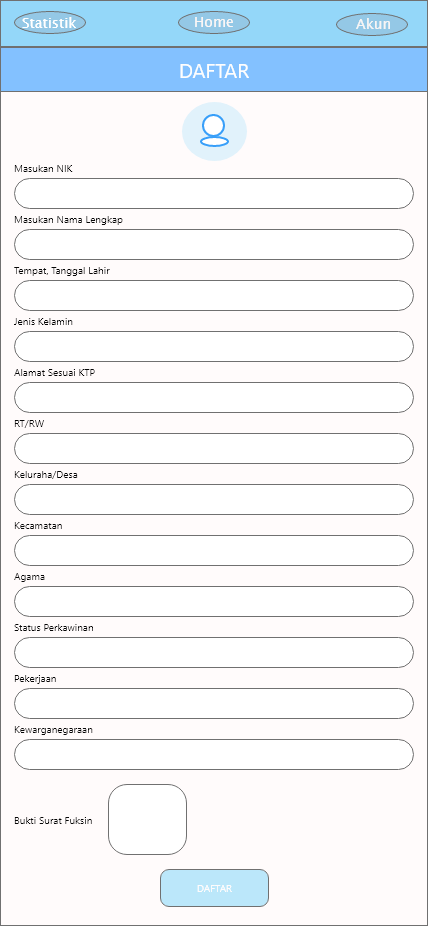
1. **Perancangan antarmuka login**

Antar muka login berfungsi untuk menampilkan halaman login yang akan digunakan admin sebelum menggunakan fitur-fitur web. Halaman login berfungsi sebagai proses masuk kesebuah halaman dengan memasukan identitas pengguna atau pemilik akun minimal terdiri dari username dan password untuk mendapat hak akses dalam mengubah atau mengakses fitur-fitur dari halaman.

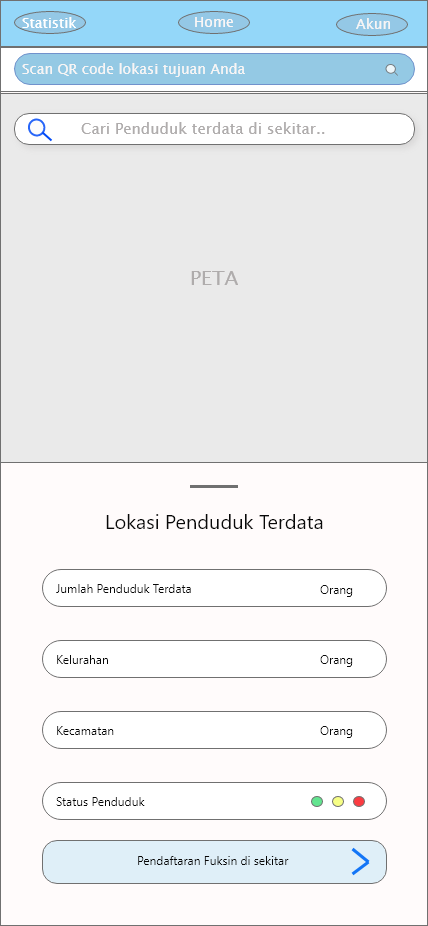


Gambar 3.2. Halaman antar muka Login

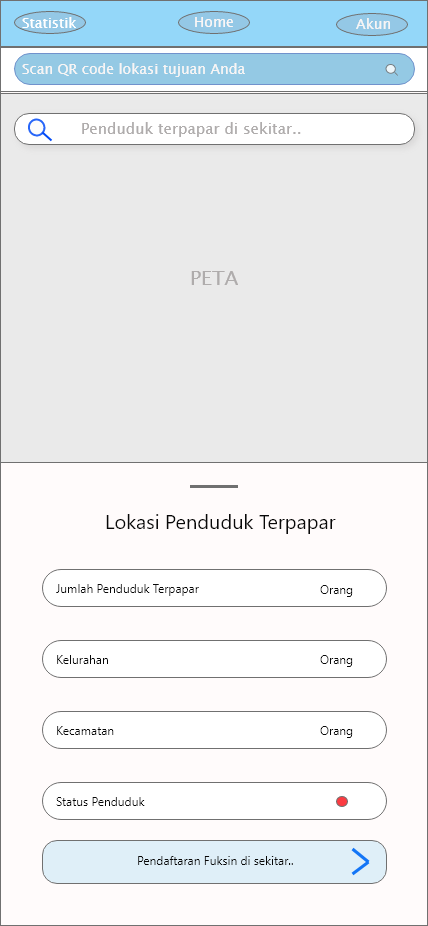
1. **Perancangan antar muka daftar**  
   Halaman ini digunakan untuk mendaftar user dan juga admin data yang diisi sesuai dengan NIK dan bukti vaksin dengan ekstensi IMG

  
  
Gambar 3.3. Halaman antar muka Daftar

1. **Perancangan antar muka penduduk terdata**  
   Halaman ini berfungsi menampilkan lokasi penduduk yang telah terdata dan berapa banyak penduduk, dan lokasi vaksin terdekat dari lokasi penduduk tersebut.

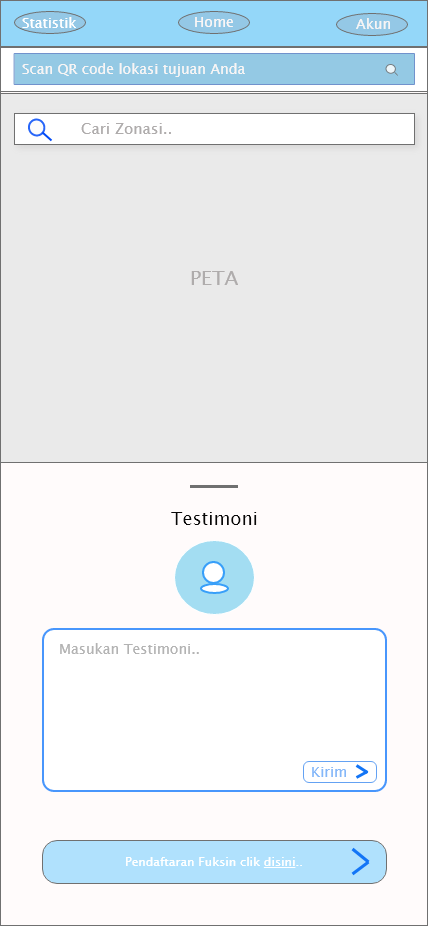


Gambar 3.4. Halaman antar muka Lokasi penduduk terdata

1. **Perancangan antar muka Lokasi penduduk terpapar**  
   Halaman ini berfungsi untuk menampilakn lokasi penduduk yang terpapar covid-19 dengan indicator berwarna merah.

Gambar 3.5. Halaman antar muka Lokasi penduduk terpapar

1. **Perancangan antar muka Testimoni**  
   Halaman ini berfungsi untuk menampilkan kolom komentar tentang website ini, untuk membantu developer dalam pengembangan website.



Gambar 3.6. Halaman antar muka Testimoni