

BAB III

METODOLOGI PENELITIAN

3.1 Subyek dan Objek Penelitian

Subjek pada penelitian ini adalah sistem pendataan *status* warga mengenai *covid-19* pada RT02/RW02, RT01/RW03, dan RT02/RW03. Sedangkan objek pada penelitian ini yaitu membuat sistem yang dapat digunakan oleh ketua RT untuk memantau warga di RT-nya, mengenai *status* vaksinasi dan *status covid-19* secara *online*.

3.2 Alat dan Bahan Penelitian

Dalam pembuatan sistem informasi pendataan *status covid-19* warga desa berbasis *web* ini diperlukan perangkat keras dan perangkat lunak yang digunakan sebagai *instrument* pembangunan sistem ini:

3.2.1 Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini adalah laptop dengan spesifikasi seperti pada tabel 3.1.

Tabel 3.1 Spesifikasi Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Laptop Asus Tuf F15	OS : Windows 10 Processor : Intel CoreI i5-10300H Memory : 8 GB
2	<i>Smartphone</i> Redmi 9T	OS : MIUI Global 12.0.7 CPU : Octa-core Max 2.0Ghz Memory : 6 GB
3	<i>Smartphone</i> Samsung Galaxy J2 Prime	OS : Android 6 CPU : Quad Core 1.4 GHz Memory : 1,5 GB

3.2.2 Perangkat Lunak

Berikut ini adalah perangkat lunak yang digunakan peneliti dalam penelitian ini seperti pada tabel 3.2.

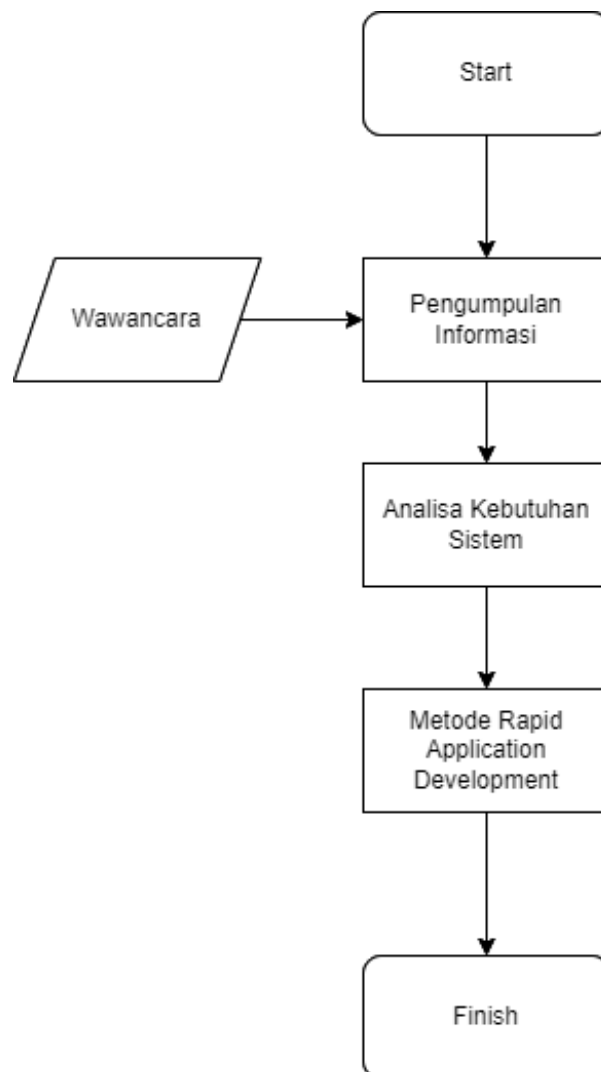
Tabel 3.2 Spesifikasi Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Nama
1	<i>Integrated Development Environment (IDE)</i>	Visual Studio Code
2	<i>Database</i>	MySQL
3	<i>Framework</i>	Laravel
4	<i>Web Browser</i>	Google Chrome dan Mozilla

3.3 Proses Penelitian

3.3.1 Diagram Alir dan *Rapid Application Development*

Pada perancangan sistem informasi ini telah dibuat diagram alir yang berawal pengumpulan informasi dan data hingga *finish* pembuatan sistem *website*-nya. Dimana setelah melakukan pengumpulan informasi dilakukan analisa mengenai kebutuhan akan pembuatan sistem informasi tersebut dan juga melakukan proses *development* menggunakan pendekatan metode *rapid application development*. Diagram alir ditunjukkan pada gambar 3. 1.

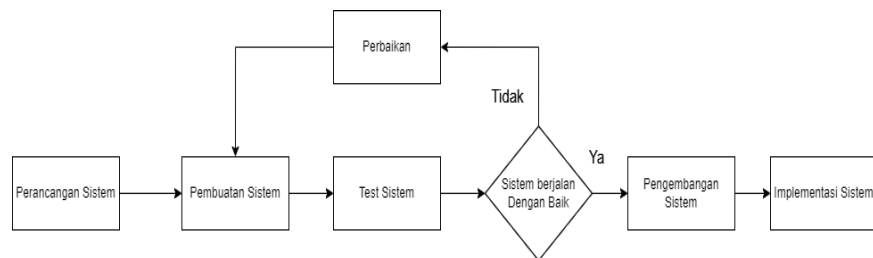


Gambar 3. 1 Diagram Alir Penelitian

Pada tahap metode *rapid application development* dilakukan tahap awal yaitu perancangan sistem dimana melakukan desain cepat mengenai sistem tersebut, mulai dari jalannya sistem hingga aktor dalam sistem tersebut. Dilanjutkan pada pembangunan sistem, *test* sistem, dan masukan yang menentukan apakah dibutuhkan iterasi dalam pengembangan sistem tersebut atau dapat langsung masuk pada tahap *final* dan implementasi. Iterasi yang telah terjadi sebanyak 2 kali. Tampilan proses *rapid application development* dapat dilihat pada gambar 3.2.

Iterasi 1 : Fokus pada pembuatan fondasi/dasar dari tabel hingga *user website*

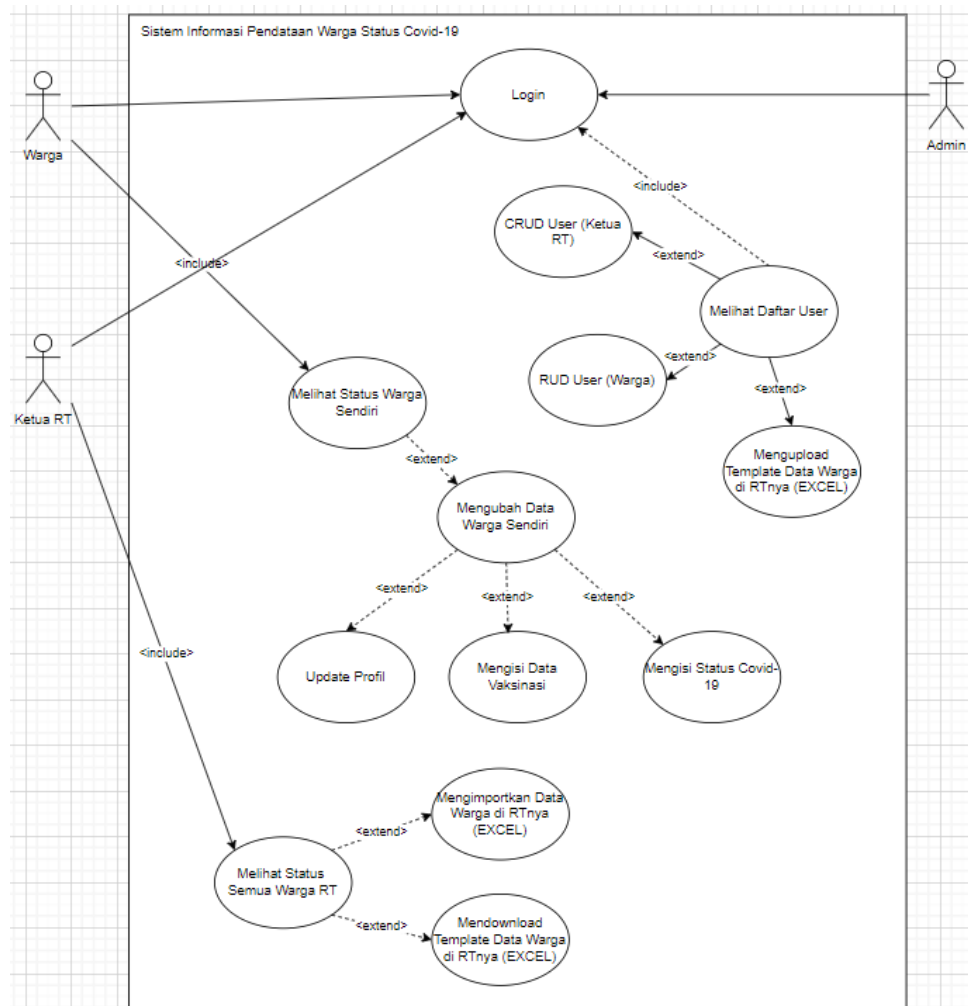
Iterasi 2 : Fokus pada perubahan detail dan kelengkapan dari kode *website* tersebut



Gambar 3. 2 *Rapid Application Development* Sistem

3.3.2 Use Case Diagram

Pada sistem informasi ini terdapat 2 macam *user* dan juga *admin*. 2 *user* tersebut adalah warga dan ketua RT. Warga memiliki akses seperti *login* menggunakan akun bawaan dengan *username* (NIK) dan *password* (tanggal lahir). Warga dapat melakukan perintah *input*, *edit*, dan *delete* pada *status* vaksinasi, serta dapat mengubah/mengganti data diri pada akun dan *status covid-19* sendiri. Sementara ketua RT dapat melakukan *login* dengan akun yang telah dibuatkan oleh *admin*. Ketua RT juga dapat melihat data yang telah dimasukan oleh warga pada RT-nya, serta mengunduh dan mengimpor *template* data sebagai cara mendaftarkan warganya pada sistem. Dimana *admin* memiliki akses mengenai *input*, *edit*, dan *delete list* akun *user*, hanya saja tidak dapat *input* akun *user* warga. Selain itu, admin juga dapat mengimpor *template* untuk pendaftaran warga. Diagram *use case* ditunjukkan pada Gambar 3. 3 dibawah ini.



Gambar 3. 3 *Use Case* Diagram Sistem

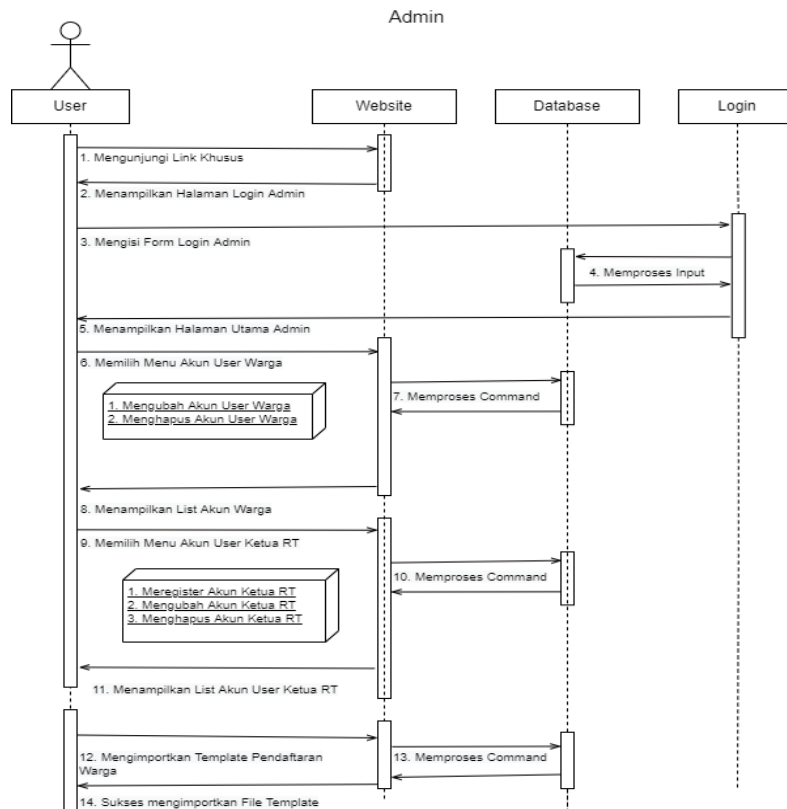
3.3.3 *Sequence* Diagram

Fungsi dari *sequence* diagram adalah untuk menunjukkan alur dari sistem yang akan dirancang, pada sistem ini memiliki 3 alur berjalannya sistem dan alur pengguna tersebut dibagi menjadi *Admin*, Ketua RT, dan Warga.

a. *Sequence* Diagram Admin

Proses *sequence* untuk *admin* dimulai dari *admin* menuju ke *link* khusus untuk melakukan proses gabung ke dalam sistem informasi, setelah selesai maka *admin* dapat mengimportir *template* pendaftaran warganya yang nantinya diunduh oleh ketua RT-nya, dan memiliki akses terhadap daftar akun untuk

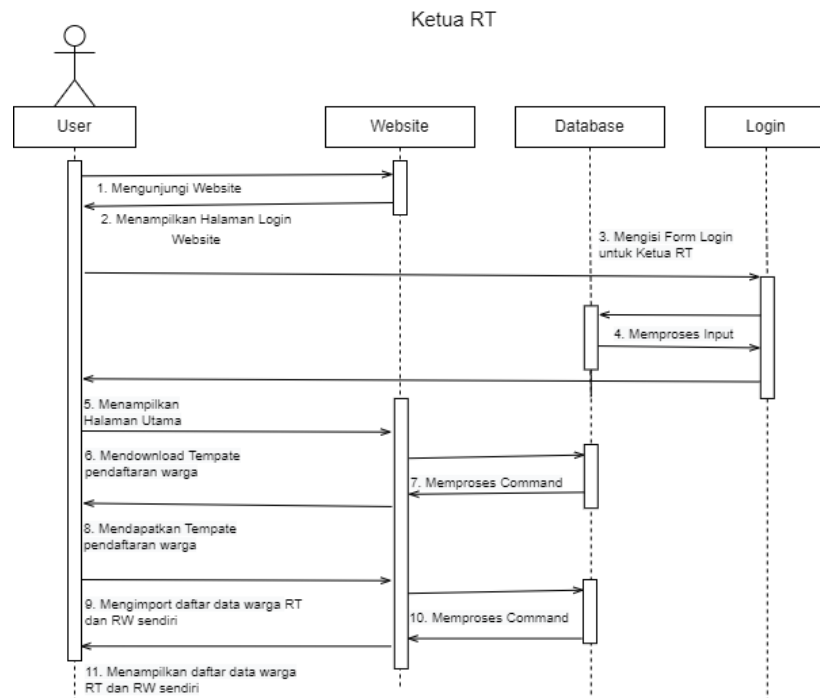
user warga dan ketua RT. Diagram *Sequence* untuk *admin* ditunjukkan pada Gambar 3. 4 dibawah ini.



Gambar 3. 4 *Sequence Diagram Admin*

b. *Sequence Diagram User Ketua RT*

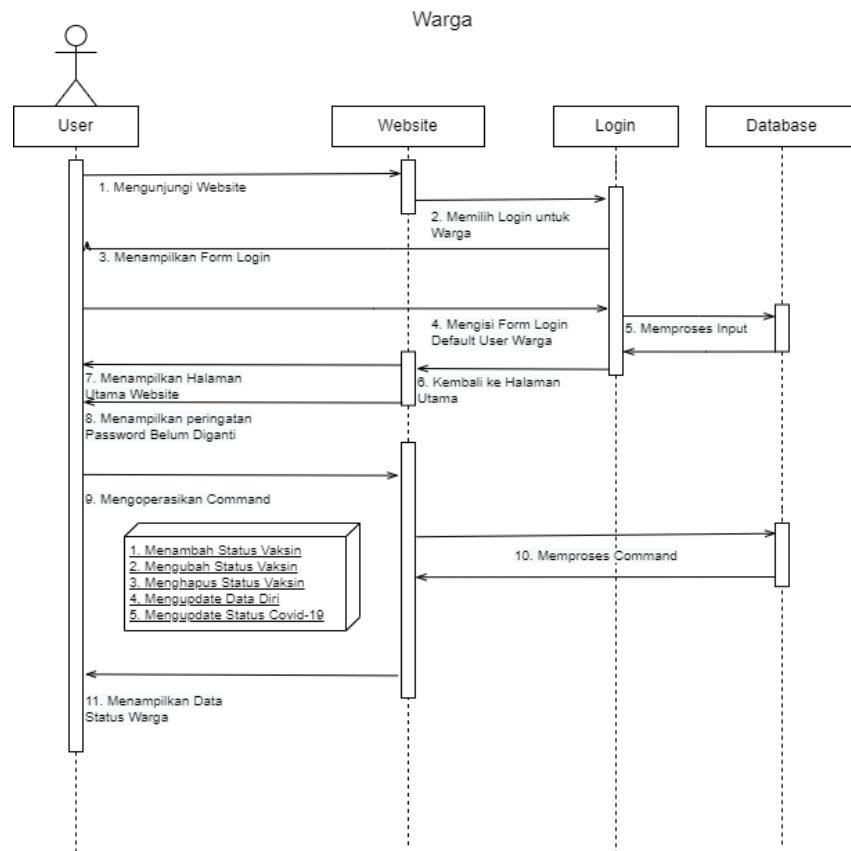
Proses *sequence* untuk ketua RT dimulai dari *user* mengunjungi halaman utama untuk melakukan proses gabung ke dalam sistem dengan menggunakan akun yang telah dibuat oleh *admin*, setelah selesai maka ketua RT dapat mengunduh *template* untuk pengumpulan data dan mengimportkan *template* tersebut ketika sudah diisi. Kemudian Ketua RT memiliki akses untuk melihat daftar akun serta *status* vaksinasinya yang telah dimasukan oleh pengguna warga tetapi ketua RT hanya dapat melihat daftar akun pada RT dan RW nya sendiri. Diagram *sequence* untuk ketua RT ditunjukkan pada Gambar 3. 5.



Gambar 3. 5 *Sequence Diagram* Ketua RT

c. *Sequence Diagram* User Warga

Proses *sequence* untuk warga dimulai dari *user* mengunjungi halaman utama untuk melakukan proses gabung ke dalam sistem. Dimana pengguna warga dapat langsung gabung menggunakan NIK dan tanggal lahir sebagai akun awal. Setelah selesai melakukan proses gabung maka warga diharuskan mengganti password bawaan sebelum memasukan data diri, status vaksinasi, serta *status covid-19*. Warga dapat menambah, mengubah dan menghapus *status* vaksinasi juga dapat mengubah *status covid-19* pengguna tersebut. *Sequence diagram* dapat dilihat pada Gambar 3. 6 *Sequence Diagram* Warga.

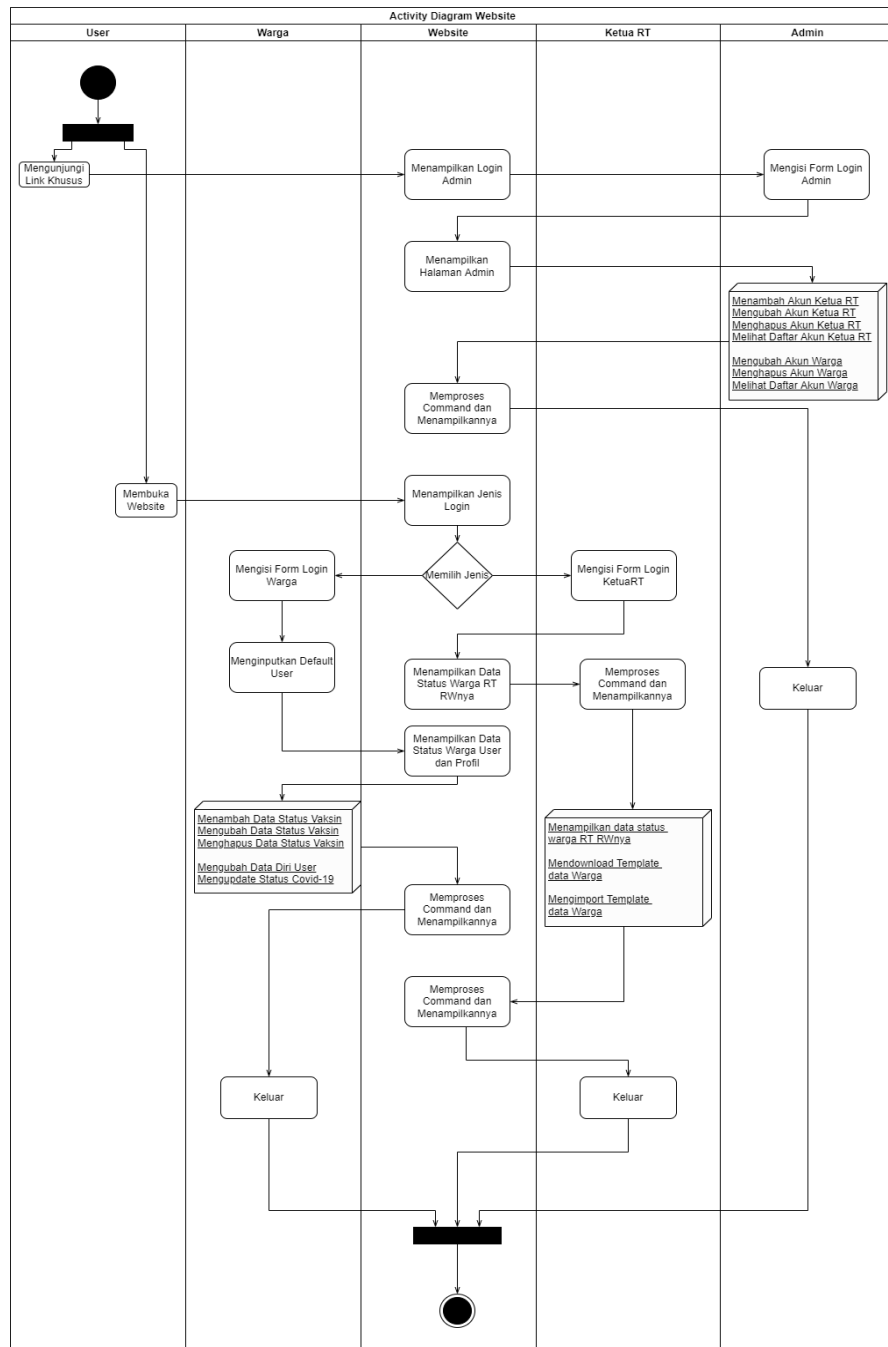


Gambar 3. 6 *Sequence Diagram Warga*

3.3.4 *Activity Diagram*

Pada *activity diagram* ini dijelaskan bahwa pengguna dibagi menjadi *admin* dan pengguna biasa. Dimana jika ingin masuk sebagai *admin* diharuskan mengunjungi *link* khusus untuk melakukan proses gabung. *Admin* memiliki akses terhadap penambahan, perubahan, dan penghapusan akun ketua RT. *Admin* juga dapat melakukan perubahan akun warga, penghapusan akun warga, serta *admin* dapat mengimporkan *file template* yang nantinya diisi oleh ketua RT sebagai data warga RT dan RW-nya. Untuk pengguna biasa diharuskan mengunjungi *link website* utama yang nantinya mendapat akses sesuai dengan tipe akunnya. Dimana ketua RT dapat mengunduh *template* data yang ingin dimasukkan pada database dan mengimpor *template* tadi agar data yang diisi pada *template* terdaftar pada database sistem. Ketua RT

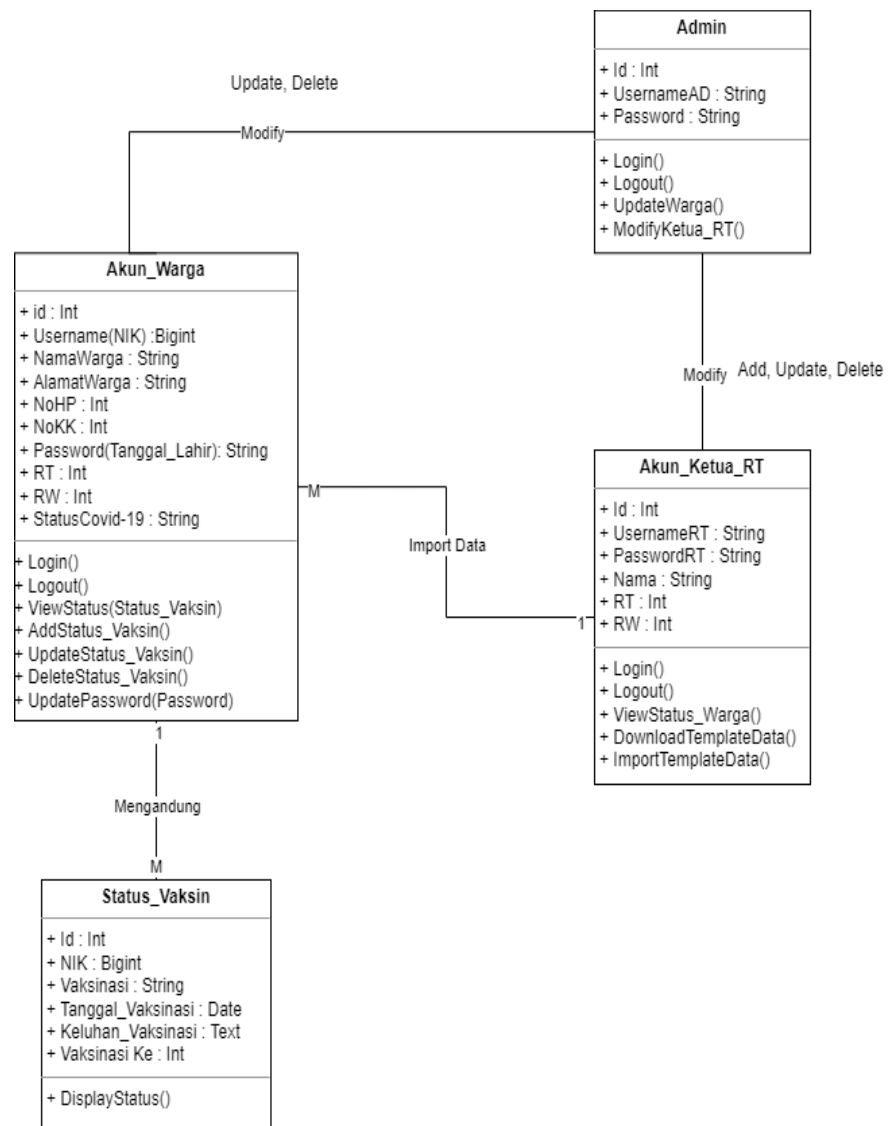
juga dapat melihat data dari warga pada RT dan RW-nya. Sementara warga dapat menambah, mengubah dan menghapus *status* vaksinasinya. Warga dapat memperbarui *status covid-19* dan data dirinya. *Activity* diagram pada sistem informasi *web* ditunjukkan pada Gambar 3. 7.



Gambar 3. 7 Activity Diagram Sistem

3.3.5 Class Diagram

Class diagram dalam *Unified Modeling Language* (UML) adalah jenis diagram struktur statis yang menggambarkan struktur sistem dengan menunjukkan kelas sistem, *attribute*, *method*, dan hubungan antara objeknya [20], pada *class* diagram ini terdapat 5 *class* yang masing masing memiliki atribut dan metodenya, selain itu kelas “Status_Vaksin” merupakan *extention* dari kelas “Akun_Warga”. *Class* diagram sistem informasi ditunjukkan pada Gambar 3. 8.



Gambar 3. 8 Class Diagram Sistem