

## BAB V

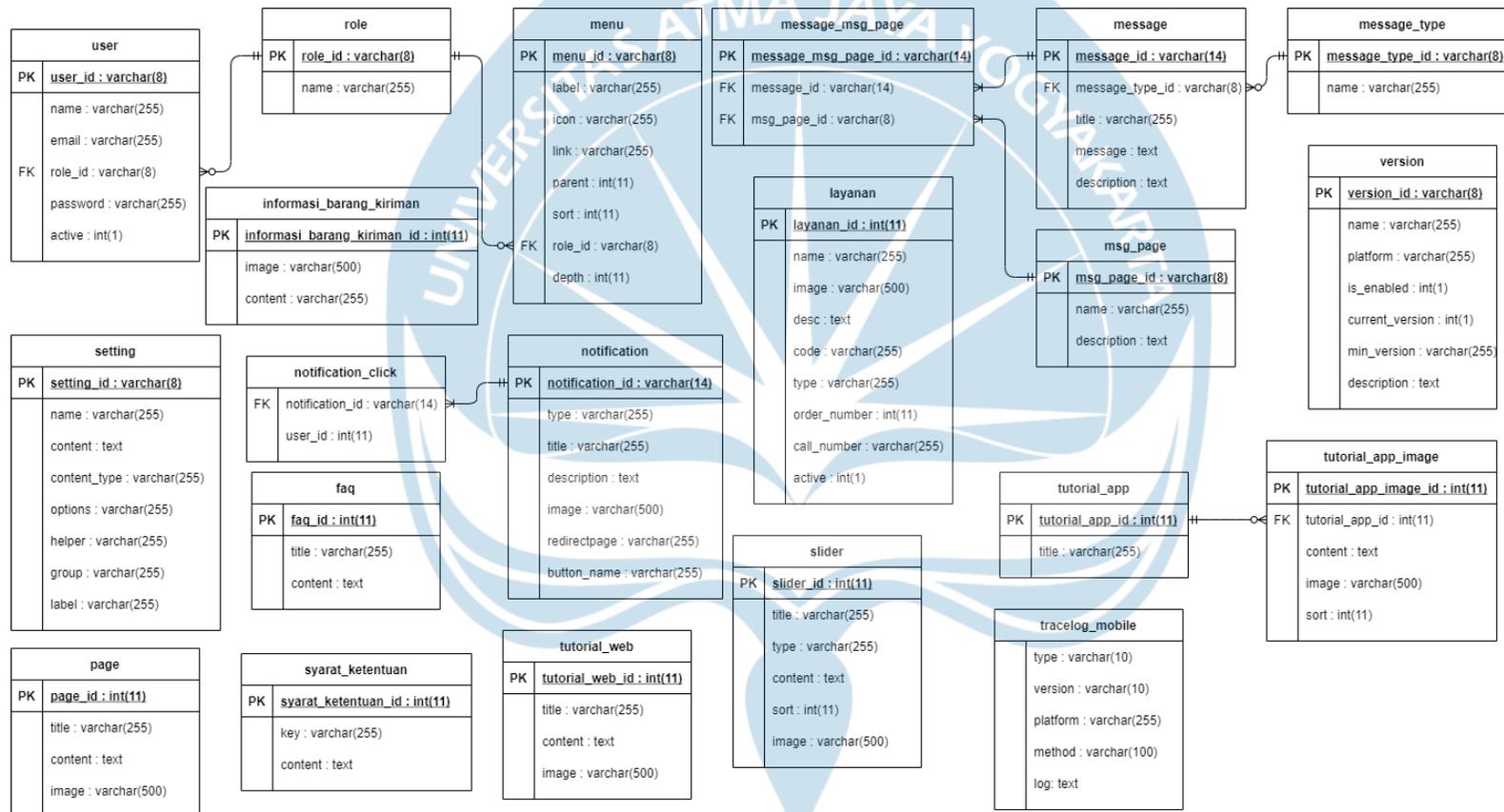
### HASIL DAN PEMBAHASAN

Berdasarkan hasil kebutuhan yang didapatkan dari tahapan analisis pada Bab IV, kebutuhan yang didapatkan akan diimplementasikan dengan menggunakan sebuah perancangan yang akan diterapkan kepada sistem. Perancangan hasil implementasi kebutuhan dibagi menjadi 3, yaitu:

1. *Entity Relationship Diagram* (ERD)
2. *Class Diagram*
3. Perancangan Antarmuka

Hasil dari tahap ini akan digunakan untuk memberikan gambaran kepada *developer* mengenai struktur data serta relasi antar tabel pada *database*, hubungan antar objek dalam sistem, dan tampilan antarmuka yang dilihat secara langsung oleh *user*.

### 5.1. Entity Relationship Diagram (ERD)



Gambar 25. Entity Relationship Diagram

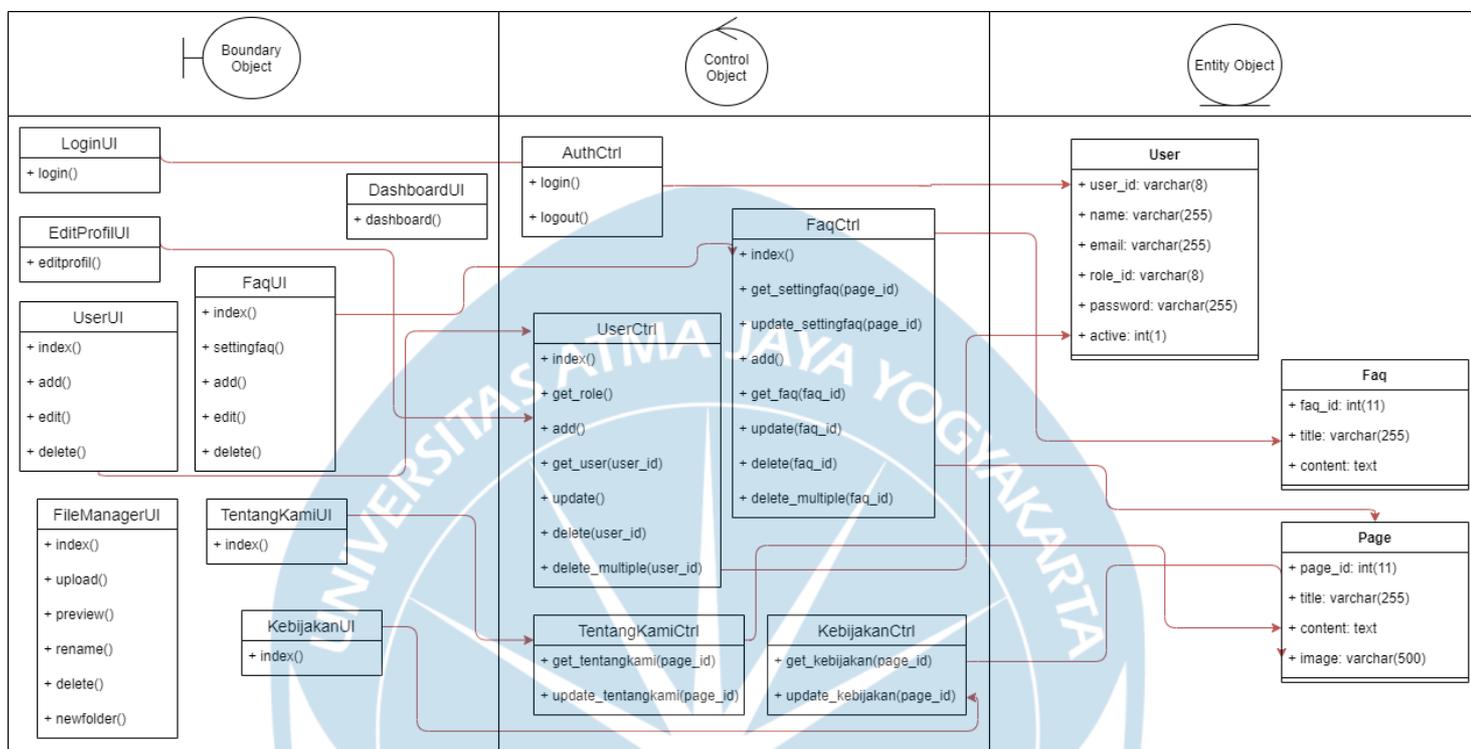
*Entity Relationship Diagram* (ERD) yang terdapat pada Gambar 25 digunakan pada penelitian ini untuk menunjukkan struktur penyimpanan data dari sistem pengelolaan web & app. ERD juga digunakan untuk menunjukkan hubungan-hubungan antar tabel sehingga dapat terlihat tabel-tabel yang digunakan pada tiap fungsi.

Sistem manajemen konten yang dibangun pada penelitian ini mengandung 21 tabel, dimana masing-masing menyimpan data-data yang terkandung pada tiap bagian di situs *web* dan aplikasi *mobile*. Dari 21 tabel tersebut, terdapat 15 data master yang merupakan tabel yang dapat berdiri sendiri tanpa memerlukan data dari tabel lainnya, dan 6 tabel transaksional yang memerlukan *foreign key* dari tabel lainnya. Terdapat 1 tabel yang tercipta dari hubungan *many-to-many* antar tabel, yaitu tabel “*messages\_msg\_pages*”, yang merupakan tabel transaksi yang menghubungkan tabel *message*, dan *msg page*. Tabel tersebut berfungsi untuk menunjukkan halaman apa saja yang akan menampilkan *message* yang di-*input*. Tabel “*page*” digunakan untuk menyimpan isi tampilan dari beberapa halaman statis seperti halaman Tentang Kami, FAQ, dan Kebijakan yang sudah berbentuk HTML sehingga administrator dapat mengubah gaya tampilan tulisan tanpa perlu melakukan *hard code*. Tabel “*setting*” digunakan untuk menyimpan pengaturan-pengaturan untuk keperluan CMS, misalnya pengaturan *default image* yang digunakan saat administrator menambahkan konten baru tanpa menginputkan *image*, dan pengaturan identitas perusahaan yang ditampilkan di *web* dan *app* seperti alamat email, no. telepon dan sebagainya. Masing-masing pengaturan disimpan sebagai baris baru dan bukan kolom, untuk mengantisipasi bertambahnya pengaturan yang ingin dikustomisasi oleh admin di masa mendatang. Tabel “*notification\_click*” merupakan tabel yang digunakan untuk menyimpan jumlah sebuah notifikasi di klik oleh *user*, dimana penginputan ke dalam tabel dilakukan secara otomatis melalui situs *web* dan aplikasi *mobile*.

Tabel-tabel lainnya melambangkan masing-masing entitas dari situs *web* dan *app mobile* dan setiap baris merupakan isi dari entitas tersebut, sebagai contoh tabel “*slider*” menyimpan data *slider* apa saja yang ditampilkan pada *web* dan *app mobile*, tabel “*layanan*” menyimpan data layanan apa saja yang ditampilkan pada *web* dan *app mobile*, dan lain sebagainya.

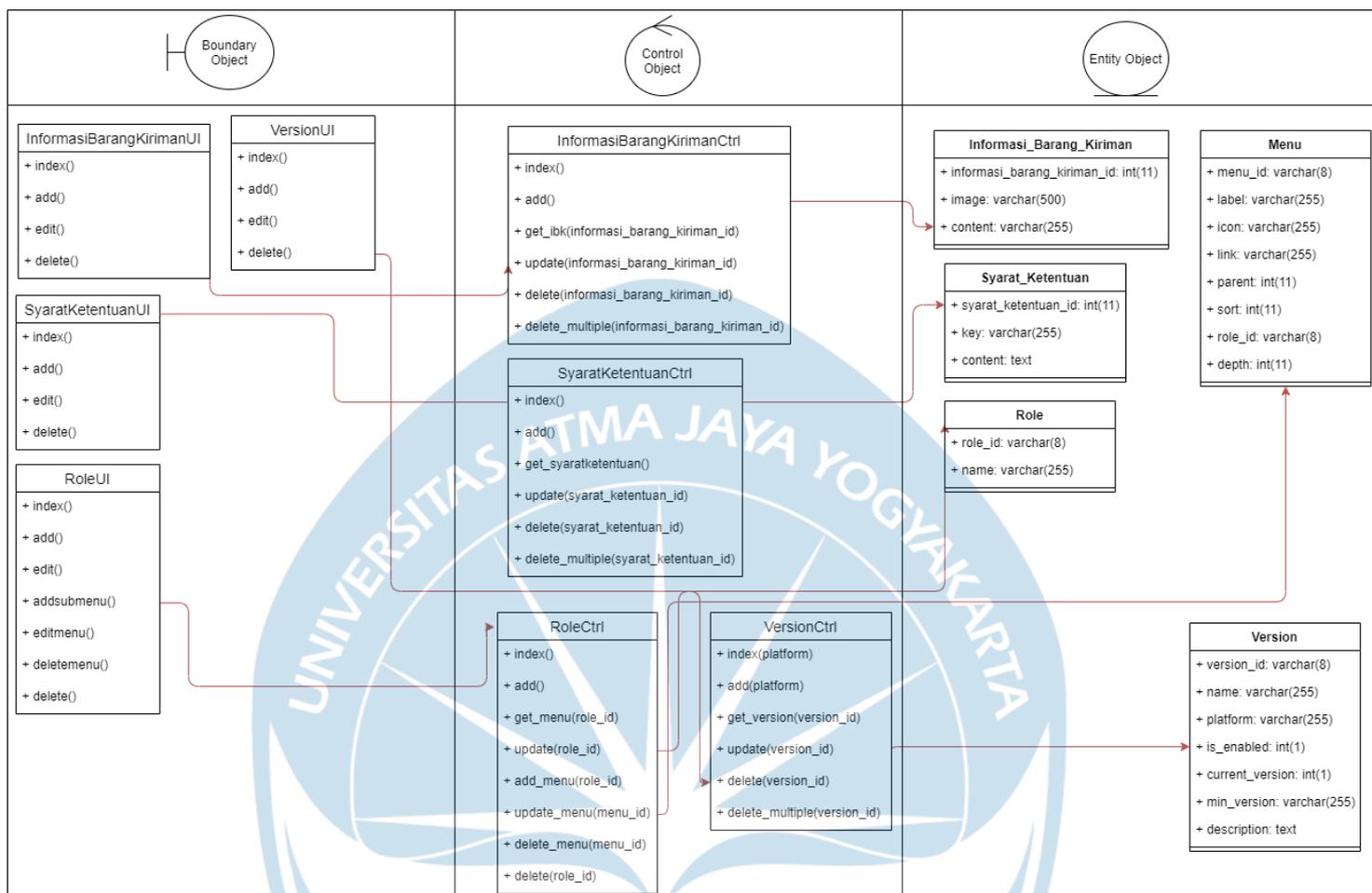


## 5.2. Class Diagram



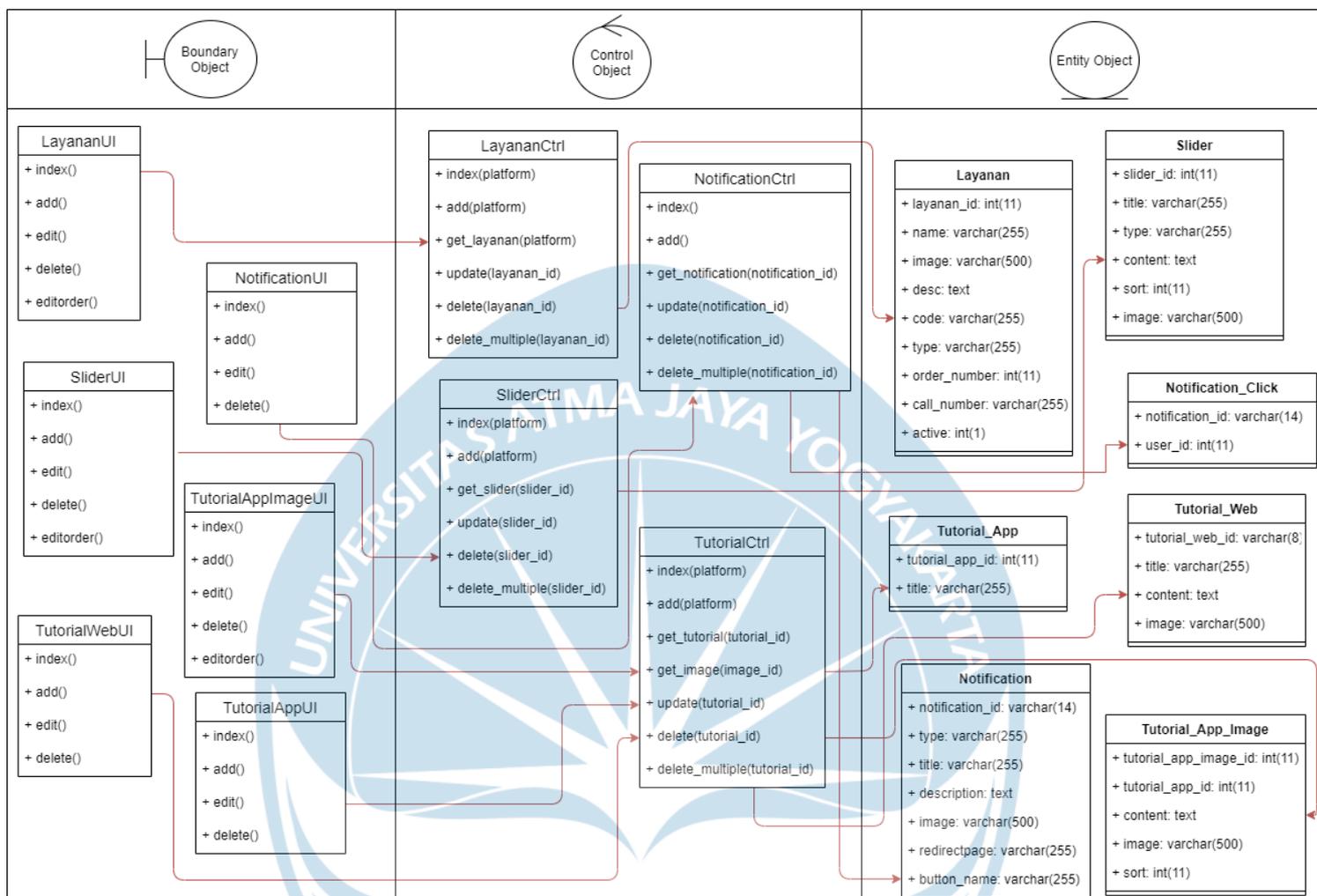
Gambar 26. Class Diagram 1

*Class diagram* pertama pada gambar 26 menunjukkan hubungan antara objek tampilan, objek *controller*, dan objek entitas atau model untuk fungsi *log in*, dashboard, edit profil, *user*, *faq*, kebijakan, tentang kami, dan *file manager*. Tampilan *log in* menggunakan *controller Auth* dan mengambil data dari model *User*. Tampilan edit profil menggunakan *controller User* dan juga mengambil data dari model *User*. Tampilan *faq*, tentang kami, dan kebijakan menggunakan *controller* nya masing-masing dan terhubung dengan model *page*. Untuk fungsi *faq*, *controller* juga terhubung dengan model *faq* untuk menyimpan daftar *faq* yang ditampilkan.



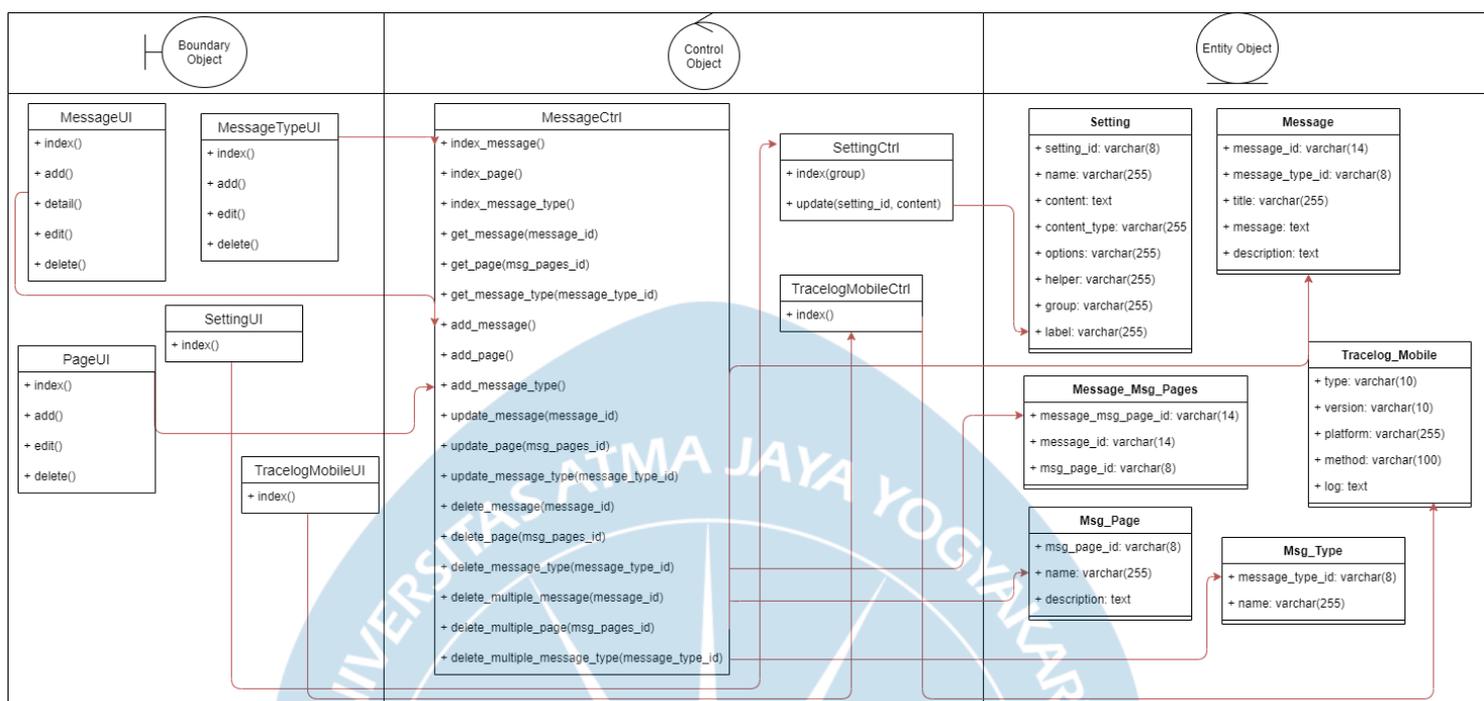
Gambar 27 Class Diagram 2

*Class diagram* kedua pada Gambar 27 menunjukkan hubungan antara objek tampilan, objek *controller*, dan objek entitas atau model untuk fungsi informasi barang kiriman, versi, syarat ketentuan dan *role*. Tampilan versi, informasi barang kiriman, dan syarat ketentuan terhubung pada *controller* dan model masing-masing objek. Tampilan *role* terhubung pada *controller role* dan pada model *role* untuk menyimpan data semua *role*, serta model menu untuk menyimpan data semua menu yang terkait pada masing-masing *role*.



Gambar 28 Class Diagram 3

Class diagram ketiga pada Gambar 28 menunjukkan hubungan antara objek tampilan, objek controller, dan objek entitas atau model untuk fungsi layanan, slider, tutorial web, tutorial app, tutorial app image, dan notifikasi. Tampilan layanan, slider, tutorial web, dan tutorial app terhubung pada controller serta model masing-masing objek.



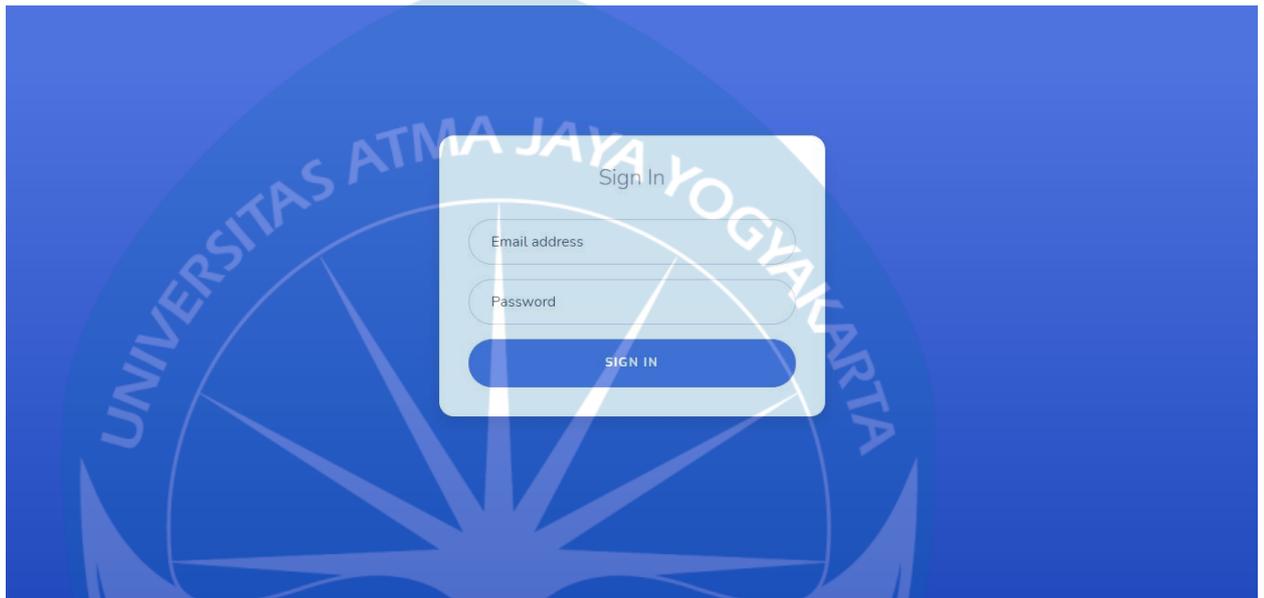
Gambar 29 Class Diagram 4

*Class diagram* keempat pada Gambar 29 menunjukkan hubungan antara objek tampilan, objek *controller*, dan objek entitas atau model untuk fungsi *message*, tipe *message*, halaman, pengaturan aplikasi, dan *tracelog mobile*. Tampilan *message*, tipe *message* dan halaman terhubung pada *controller message* dan pada model masing-masing objek. Model *Message\_Msg\_Pages* digunakan untuk fungsi *message* untuk menyimpan data halaman masing-masing halaman pada fungsi *message*. Tampilan *tracelog mobile* dan pengaturan terhubung pada *controller* dan model masing-masing objek.

### 5.3. Deskripsi Perancangan Antarmuka

#### 5.3.1 Antarmuka LoginUI

##### 5.3.1.1. Antarmuka *login()*

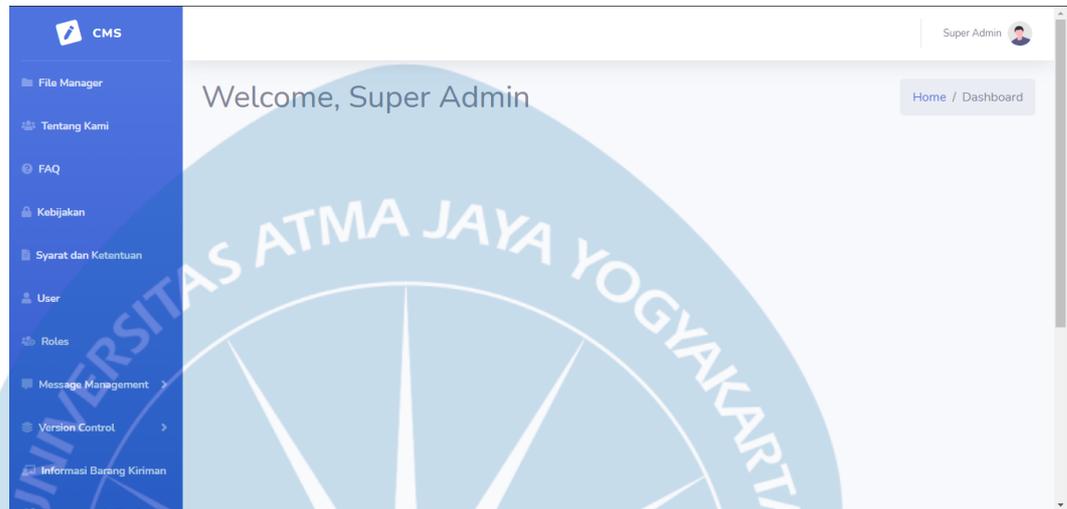


Gambar 30. Tampilan Antarmuka LoginUI Halaman *login()*

Halaman *login()* yang ditampilkan pada Gambar 30 digunakan untuk melakukan validasi pengguna yang dapat mengakses sistem pengelolaan web & app. Pengguna yang dapat masuk ke dalam sistem merupakan akun yang sudah terdaftar pada menu *User*. Pengguna yang berhasil melakukan *log in* akan diarahkan menuju halaman *dashboard* dari CMS.

### 5.3.2 Antarmuka DashboardUI

#### 5.3.2.1 Antarmuka *dashboard()*

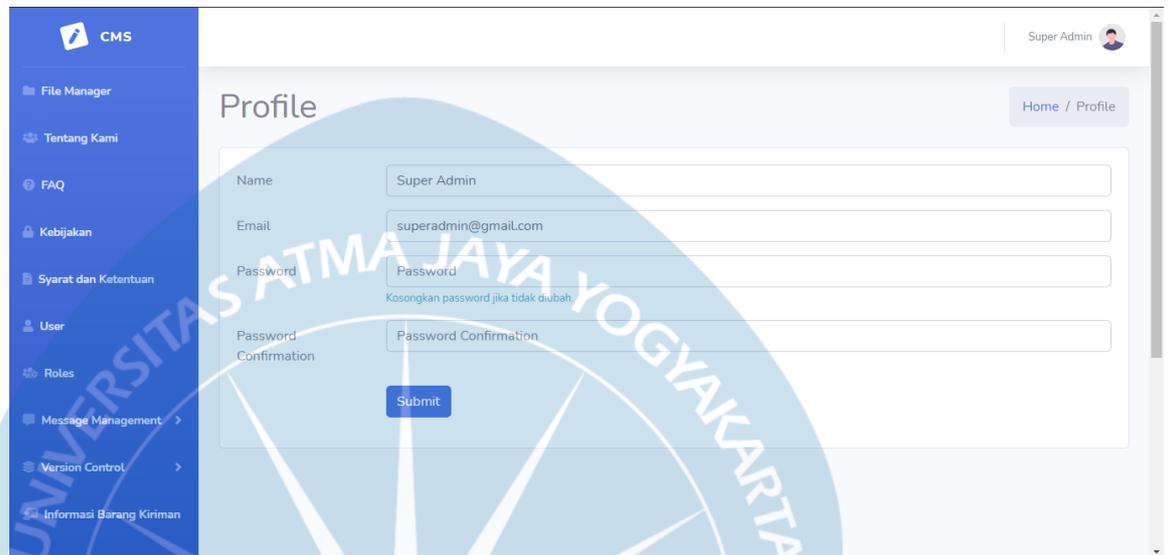


**Gambar 31.** Tampilan Antarmuka DashboardUI Halaman *dashboard()*

Halaman *dashboard()* yang ditampilkan pada Gambar 31. Tampilan Antarmuka DashboardUI Halaman *dashboard()* berfungsi sebagai halaman *landing* setelah *user* berhasil melakukan *log in*. Halaman ini tidak mempunyai fungsi spesifik, melainkan hanya menampilkan nama *user* yang sudah berhasil melakukan *log in*.

### 5.3.3. Antarmuka EditProfilUI

#### 5.3.3.1. Antarmuka editprofil()



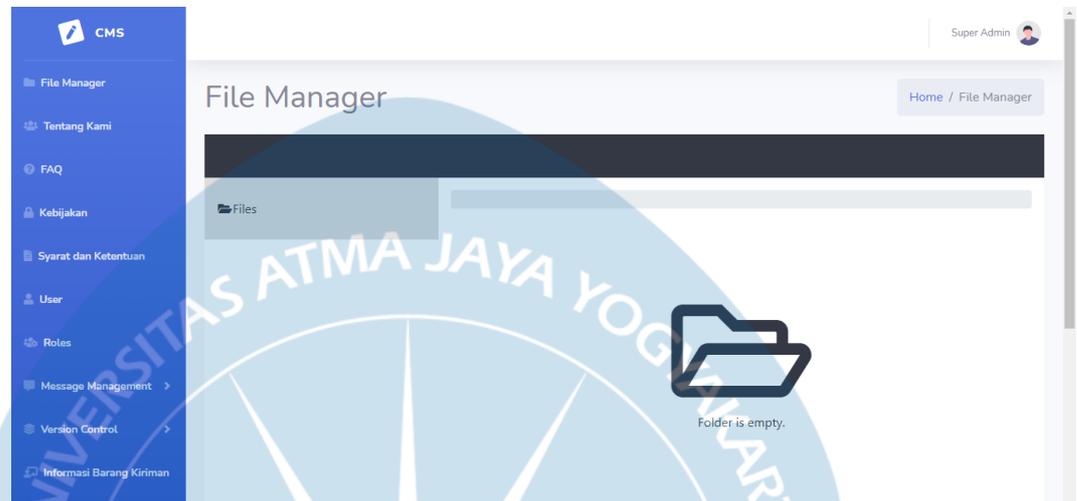
The screenshot displays a web interface for editing a user profile. On the left is a blue sidebar with a 'CMS' logo and a list of menu items: File Manager, Tentang Kami, FAQ, Kebijakan, Syarat dan Ketentuan, User, Roles, Message Management, Version Control, and Informasi Barang Kiriman. The main area is titled 'Profile' and features a form with the following fields: Name (filled with 'Super Admin'), Email (filled with 'superadmin@gmail.com'), Password (filled with 'Password'), and Password Confirmation (filled with 'Password Confirmation'). A blue 'Submit' button is positioned below the form. In the top right corner, the user 'Super Admin' is identified with a profile icon. A breadcrumb trail shows 'Home / Profile'.

**Gambar 32.** Tampilan Antarmuka EditProfilUI Halaman editprofil()

Halaman editprofil() yang ditampilkan pada Gambar 32 merupakan halaman yang berfungsi untuk menampilkan dan mengubah data *user* yang sudah berhasil melakukan *log in*. Tombol “*Submit*” digunakan untuk menyimpan semua data yang terdapat pada *form edit* profil.

### 5.3.4 Antarmuka FileManagerUI

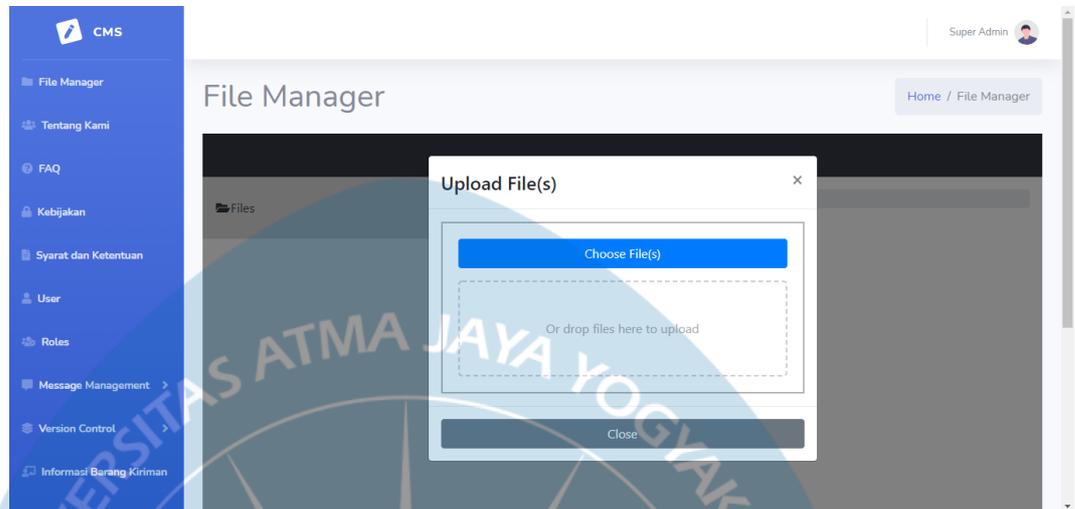
#### 5.3.4.1. Antarmuka *index()*



**Gambar 33.** Tampilan Antarmuka FileManagerUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 33 berfungsi untuk menampilkan daftar *file* dan *folder* yang tersimpan pada sistem penyimpanan *cloud*. Lokasi *folder* yang ditampilkan terletak pada *folder* root dari sistem penyimpanan *cloud*. Fungsi *file manager* menggunakan *plugin* yaitu *Laravel File Manager* dari Unisharp.

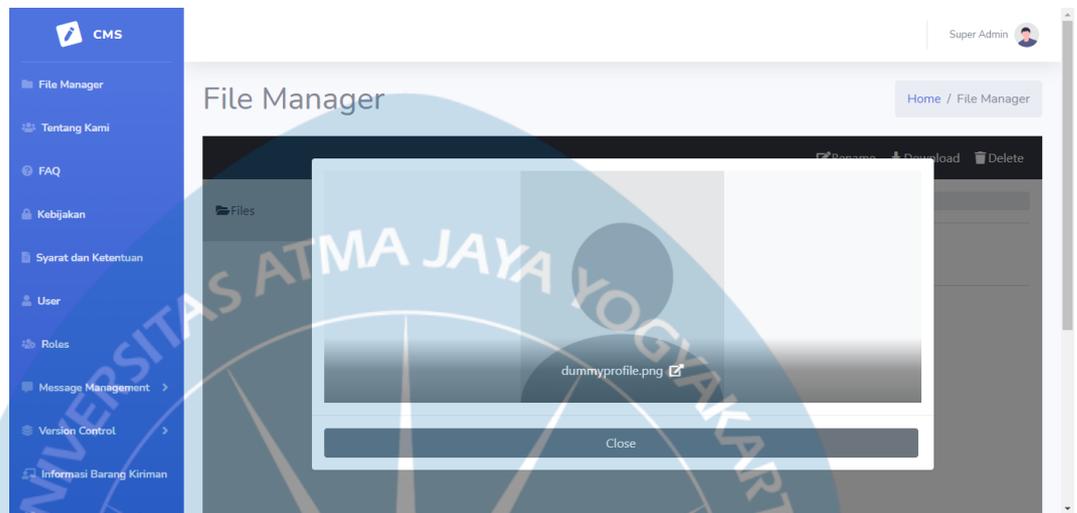
### 5.3.4.2. Antarmuka *upload()*



**Gambar 34.** Tampilan Antarmuka FileManagerUI Halaman *upload()*

Halaman *upload()* yang ditampilkan pada Gambar 34 merupakan kotak *dialog* yang berfungsi untuk menampilkan tombol pilih *file*, serta menampilkan *file* apa saja yang sudah di *upload* oleh *user*. Tombol “Choose *File(s)*” digunakan untuk membuka *file manager* agar *user* dapat memilih *file* yang ingin di *upload*. *File-file* terpilih kemudian akan di *upload* dan akan ditampilkan pada kotak di bawah tombol ”Choose *File(s)*”. Tombol “Close” digunakan untuk menutup halaman *upload*. *File* akan ter *upload* pada lokasi *folder* yang dipilih *user* saat meng klik tombol *upload*.

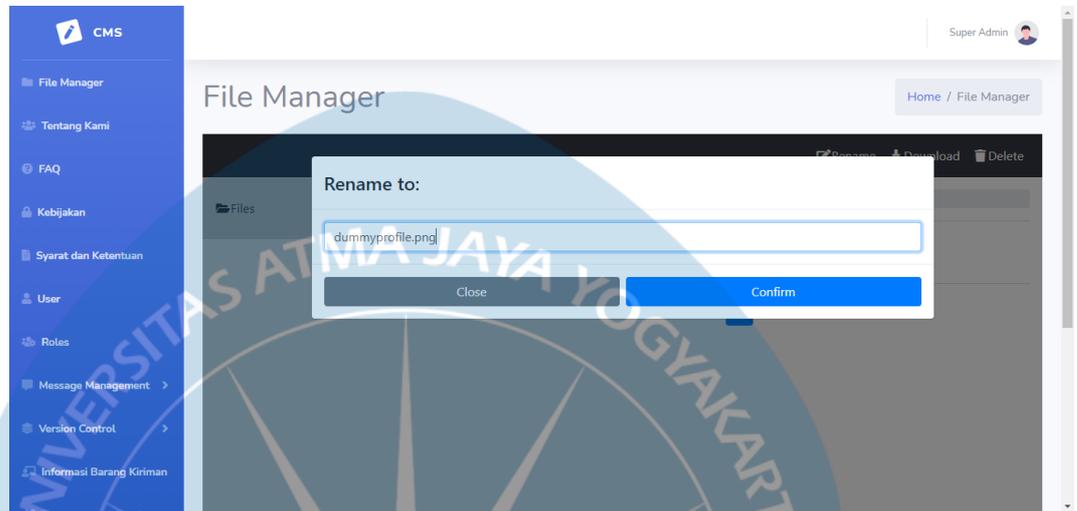
### 5.3.4.3. Antarmuka *preview()*



**Gambar 35.** Tampilan Antarmuka FileManagerUI Halaman *preview()*

Halaman *preview()* yang ditampilkan pada Gambar 35 merupakan kotak *dialog* yang berfungsi untuk menampilkan *file* gambar dengan ukuran yang lebih besar. Tombol “Close” digunakan untuk menutup halaman *preview* dan kembali ke halaman *index()*.

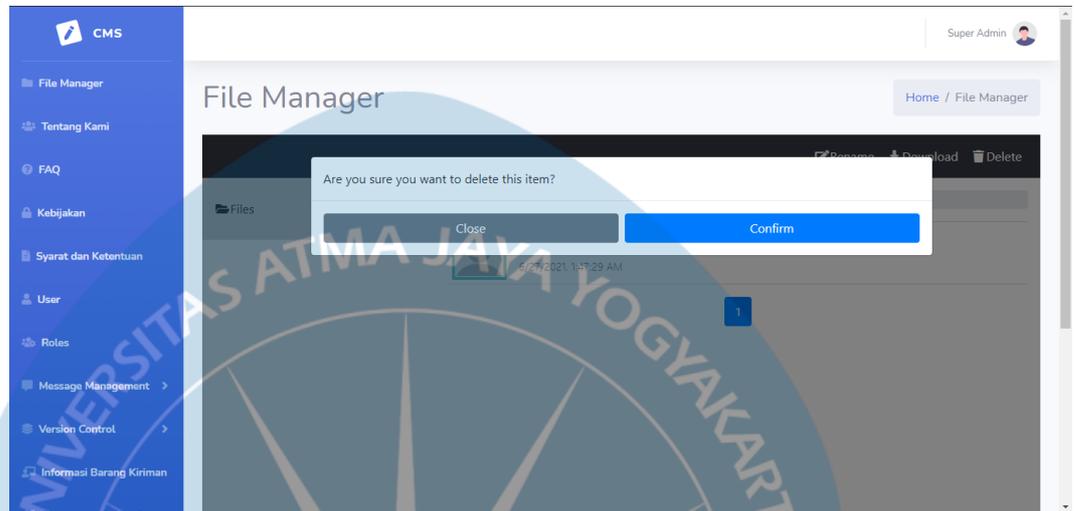
#### 5.3.4.4. Antarmuka rename()



**Gambar 36.** Tampilan Antarmuka FileManagerUI Halaman rename()

Halaman rename() yang ditampilkan pada Gambar 36 merupakan kotak *dialog* yang berfungsi untuk menampilkan serta merubah nama dari *file* terpilih. Nama *file* akan tersimpan setelah *user* meng-klik tombol “Confirm”.

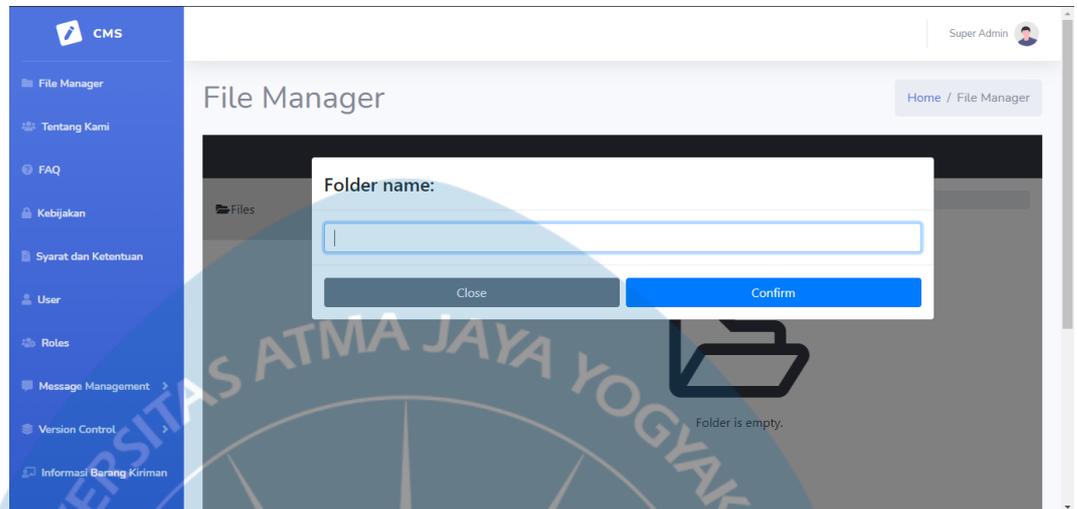
### 5.3.4.5. Antarmuka *delete()*



**Gambar 37.** Tampilan Antarmuka FileManagerUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 37 berfungsi untuk menampilkan kotak *dialog* yang digunakan sebagai konfirmasi hapus *file/folder*. Tombol “Confirm” digunakan sebagai tombol konfirmasi, sehingga *file/folder* akan dihapus setelah *user* meng klik tombol tersebut. Tombol “Close” digunakan untuk menutup halaman *delete()* dan membatalkan penghapusan *file/folder*.

#### 5.3.4.5. Antarmuka *newfolder()*

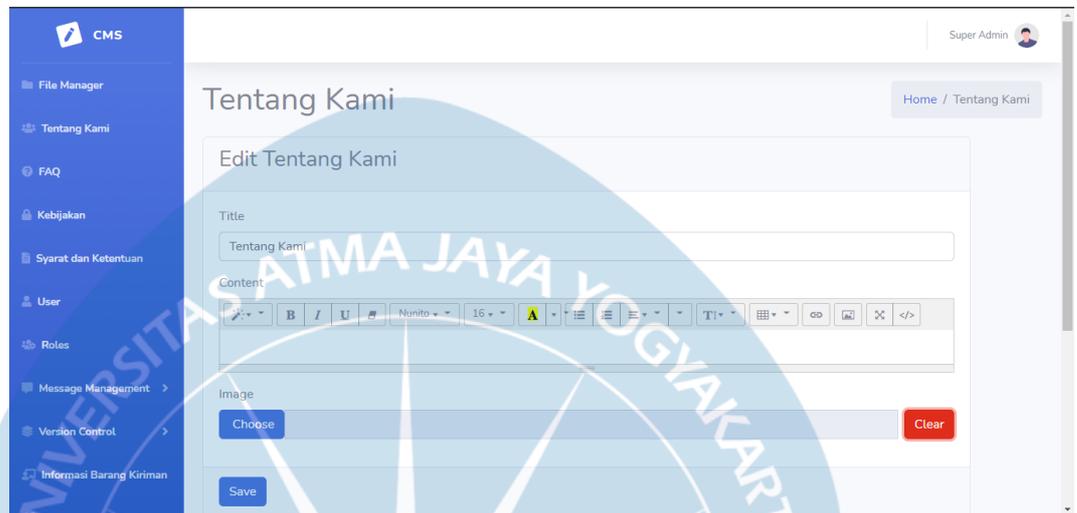


**Gambar 38.** Tampilan Antarmuka FileManagerUI Halaman *newfolder()*

Halaman *newfolder()* yang ditampilkan pada Gambar 38 merupakan kotak *dialog* yang berfungsi untuk menambahkan *folder* baru ke dalam sistem penyimpanan pada lokasi yang terpilih. Tombol “Confirm” berfungsi untuk menyimpan nama *folder* baru dan menambah *folder* baru ke dalam sistem penyimpanan. Tombol “Close” digunakan untuk menutup halaman *newfolder()* dan kembali ke halaman *index()*.

### 5.3.5. Antarmuka TentangKamiUI

#### 5.3.5.1. Antarmuka *index()*

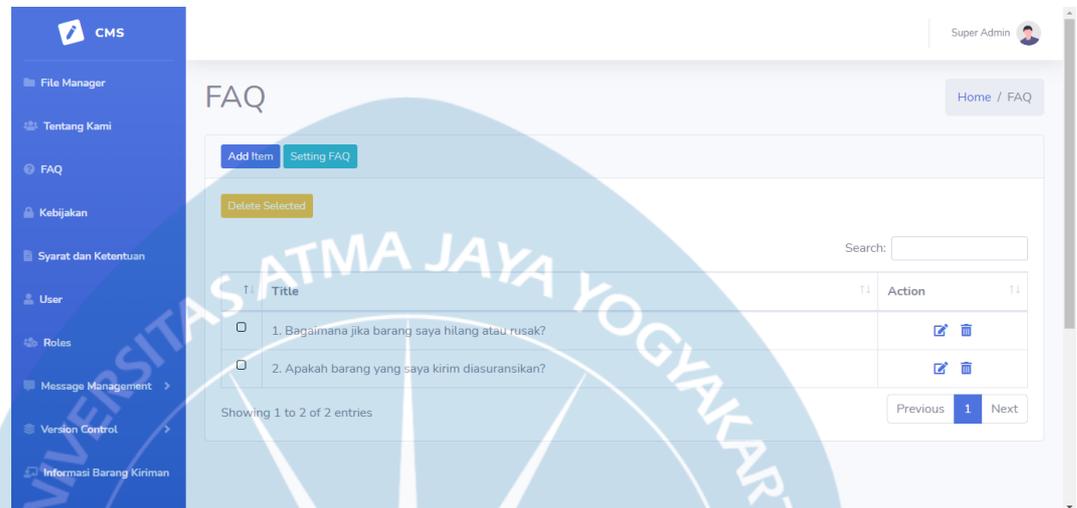


**Gambar 39.** Tampilan Antarmuka TentangKamiUI Halaman *index()*

Halaman *index()* Gambar 39 merupakan halaman yang berfungsi untuk menampilkan dan mengubah isi konten dari bagian “Tentang Kami”. Bagian *content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol *save* digunakan untuk menyimpan isi dari *form* ke dalam *database*.

## 5.3.6. Antarmuka FaqUI

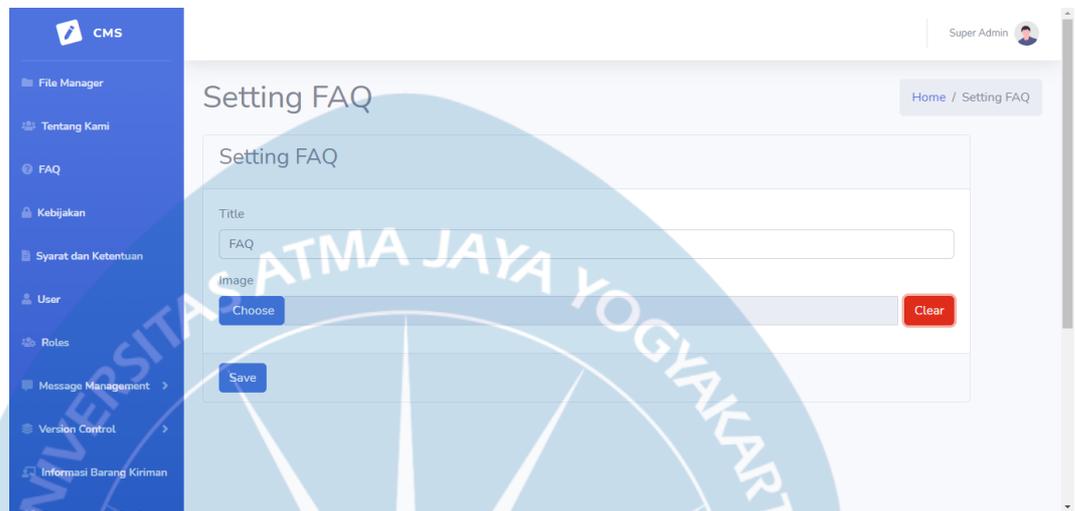
### 5.3.6.1. Antarmuka *index()*



**Gambar 40.** Tampilan Antarmuka FaqUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 40 berfungsi untuk menampilkan daftar pertanyaan yang ditampilkan pada bagian FAQ. Terdapat tombol “Add Item” yang berfungsi untuk masuk ke halaman *add()* dan menambahkan FAQ baru. Tombol “Setting FAQ” berfungsi untuk menampilkan halaman FAQ dan mengubah keterangan FAQ. Tombol dengan *icon* pensil digunakan untuk masuk ke halaman *edit* FAQ terpilih. Tombol dengan *icon* tempat sampah, dan “Delete Selected” digunakan untuk menghapus FAQ terpilih.

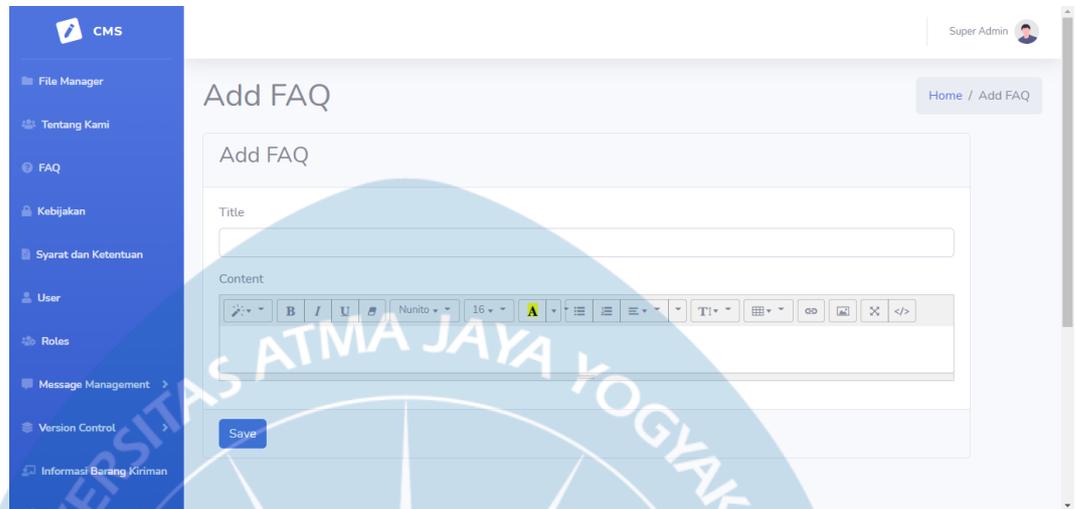
### 5.3.6.2. Antarmuka *settingfaq()*



**Gambar 41.** Tampilan Antarmuka FaqUI Halaman *settingfaq()*

Halaman *settingfaq()* yang ditampilkan pada Gambar 41 digunakan untuk meng-*edit* bagian judul, serta gambar yang ditampilkan pada halaman FAQ. Tombol *save* digunakan untuk menyimpan isi dari inputan *user* tersebut.

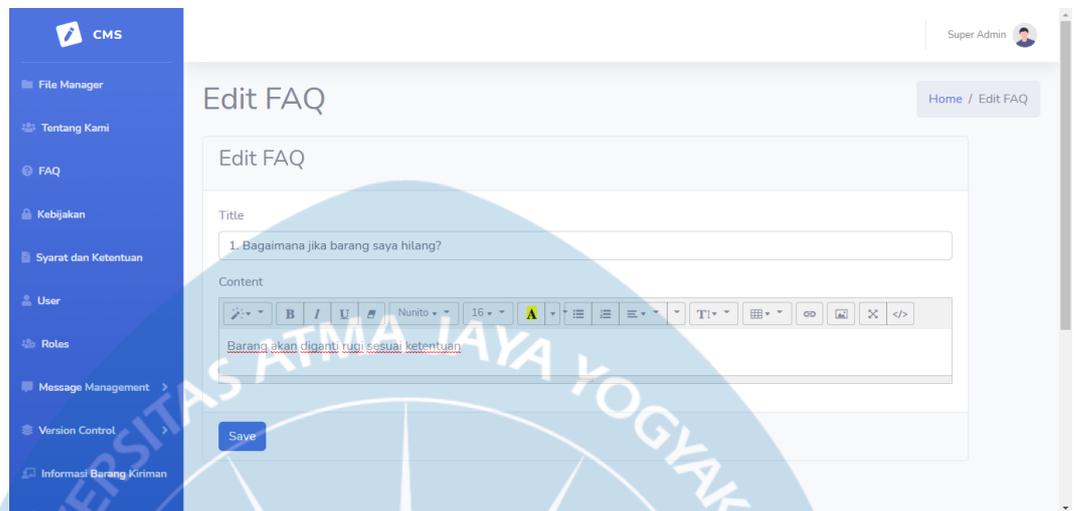
### 5.3.6.3. Antarmuka *add()*



**Gambar 42.** Tampilan Antarmuka FaqUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 42 berfungsi untuk menambahkan data FAQ baru. Bagian *content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol *save* digunakan untuk menambahkan FAQ yang diinputkan pada *form* ke dalam *database*.

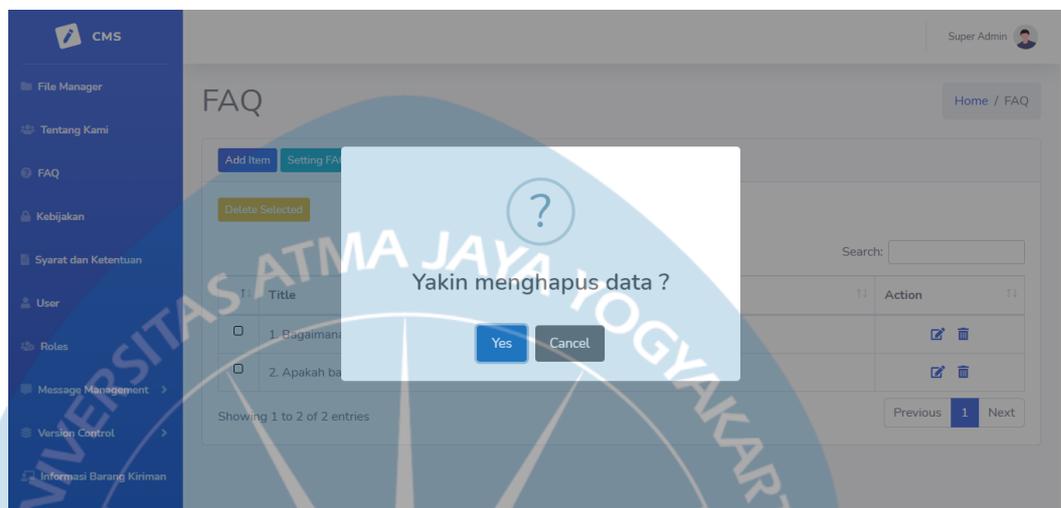
#### 5.3.6.4. Antarmuka *edit()*



**Gambar 43.** Tampilan Antarmuka FaqUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 43 digunakan untuk menampilkan dan mengubah data FAQ terpilih. Bagian *content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “Save” digunakan untuk menyimpan perubahan data FAQ ke dalam *database*.

### 5.3.6.5. Antarmuka *delete()*

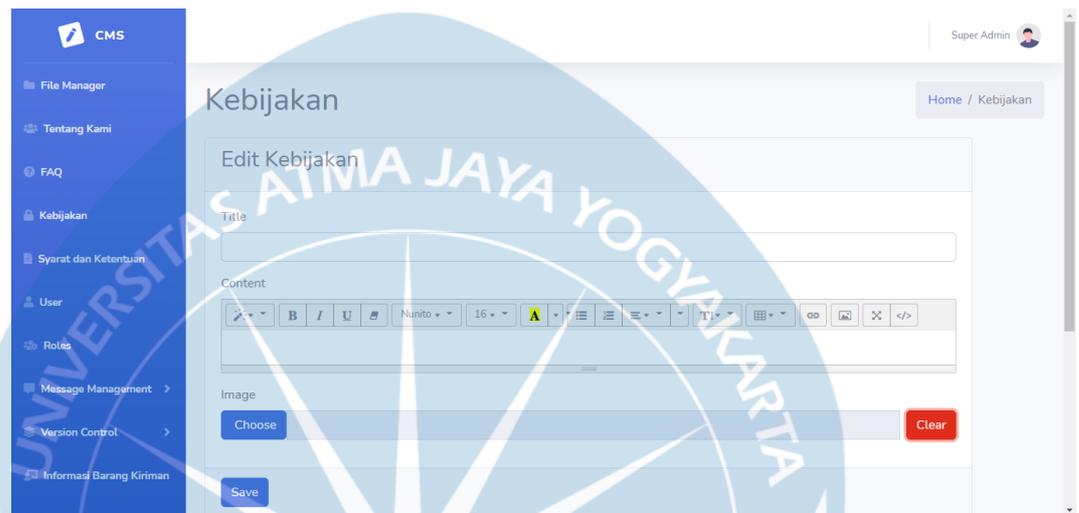


**Gambar 44.** Tampilan Antarmuka FaqUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 44 merupakan sebuah kotak *dialog* yang berfungsi sebagai halaman konfirmasi untuk menghapus FAQ terpilih. Tombol “*Yes*” digunakan sebagai tombol konfirmasi untuk menghapus FAQ terpilih. Tombol “*Cancel*” berfungsi untuk menutup halaman *delete()* dan kembali ke halaman *index()* FAQ.

### 5.3.7. Antarmuka KebijakanUI

#### 5.3.7.1. Antarmuka *index()*

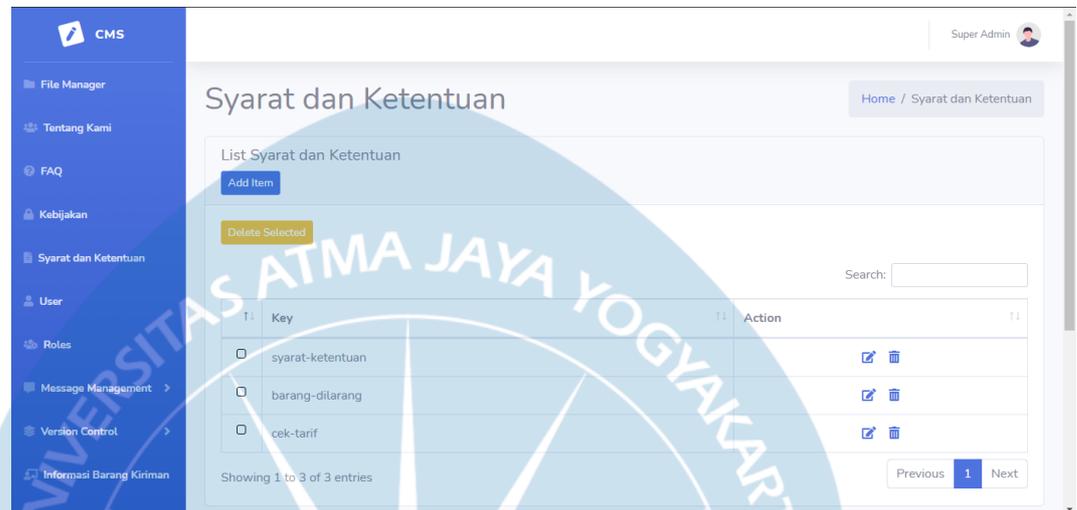


**Gambar 45.** Tampilan Antarmuka KebijakanUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 45 berfungsi untuk menampilkan dan meng-*edit* isi konten dari bagian Kebijakan. Bagian *content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “*Save*” digunakan untuk menyimpan perubahan data Kebijakan dan menyimpan ke dalam *database*.

### 5.3.8. Antarmuka SyaratKetentuanUI

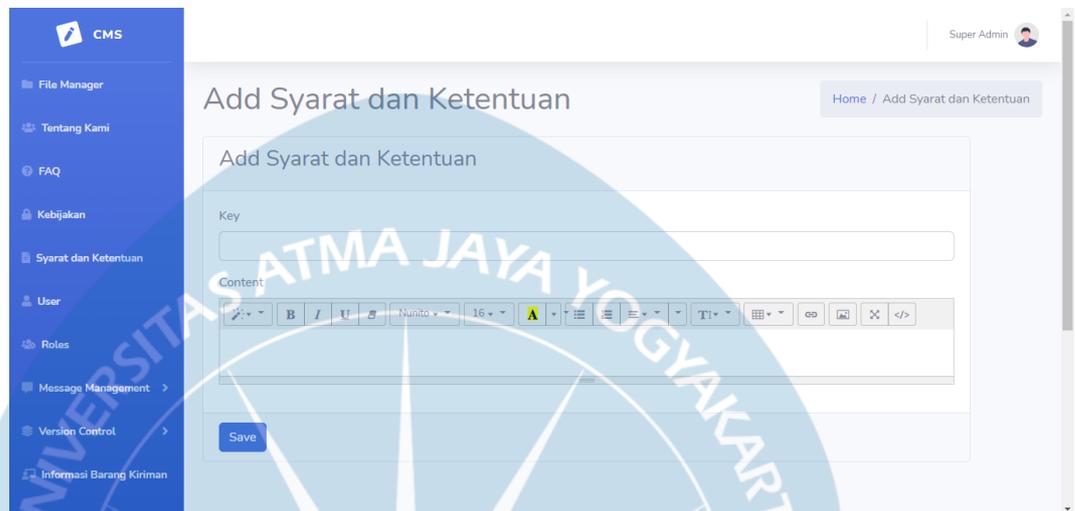
#### 5.3.8.1. Antarmuka *index()*



**Gambar 46.** Tampilan Antarmuka SyaratKetentuanUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 46 berfungsi untuk menampilkan daftar syarat ketentuan dan *tooltip-tooltip* yang ditampilkan pada *web* dan *app*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()* untuk menambah syarat ketentuan atau *tooltip* baru. Tombol dengan *icon* pensil berfungsi untuk masuk ke halaman *edit()* untuk mengubah data syarat ketentuan terpilih. Tombol “Delete Selected” dan tombol dengan *icon* tempat sampah digunakan untuk memilih dan menghapus syarat ketentuan terpilih.

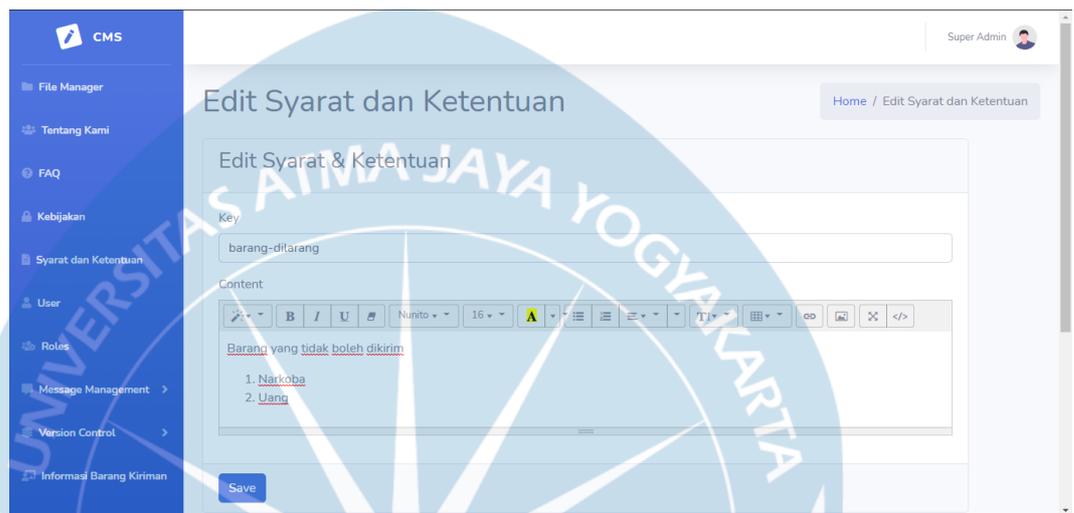
### 5.3.8.2. Antarmuka *add()*



**Gambar 47.** Tampilan Antarmuka SyaratKetentuanUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 47 berfungsi untuk menambahkan data syarat ketentuan ataupun *tooltip* baru. Bagian *content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “*Save*” berfungsi untuk menyimpan syarat ketentuan baru ke dalam *database*.

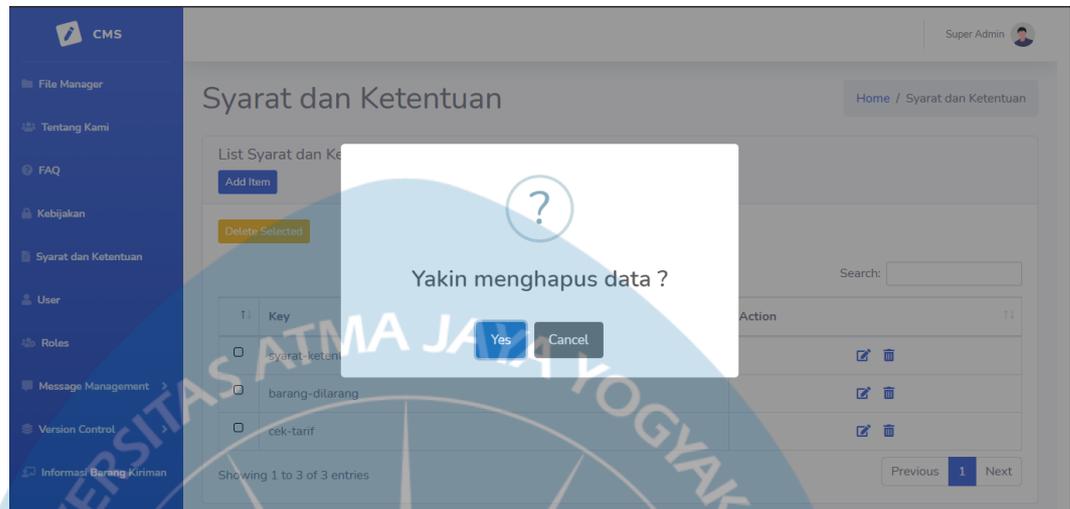
### 5.3.8.3. Antarmuka *edit()*



**Gambar 48.** Tampilan Antarmuka SyaratKetentuanUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 48 berfungsi untuk menampilkan dan mengubah data syarat ketentuan terpilih. Bagian *content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “Save” digunakan untuk menyimpan perubahan data syarat ketentuan ke *database*.

#### 5.3.8.4. Antarmuka *delete()*

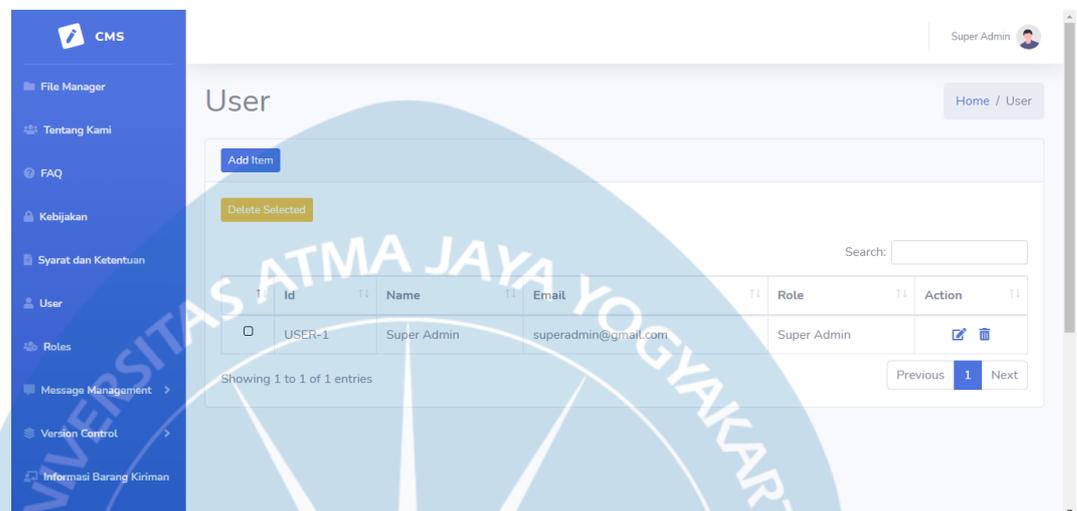


**Gambar 49.** Tampilan Antarmuka SyaratKetentuanUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 49 merupakan kotak *dialog* yang berfungsi sebagai halaman konfirmasi untuk menghapus syarat ketentuan terpilih. Tombol “*Yes*” berfungsi untuk mengkonfirmasi penghapusan syarat ketentuan terpilih. Tombol “*Cancel*” berfungsi untuk menutup dan membatalkan penghapusan syarat ketentuan terpilih dan kembali ke halaman *index()* syarat ketentuan.

### 5.3.9. Antarmuka UserUI

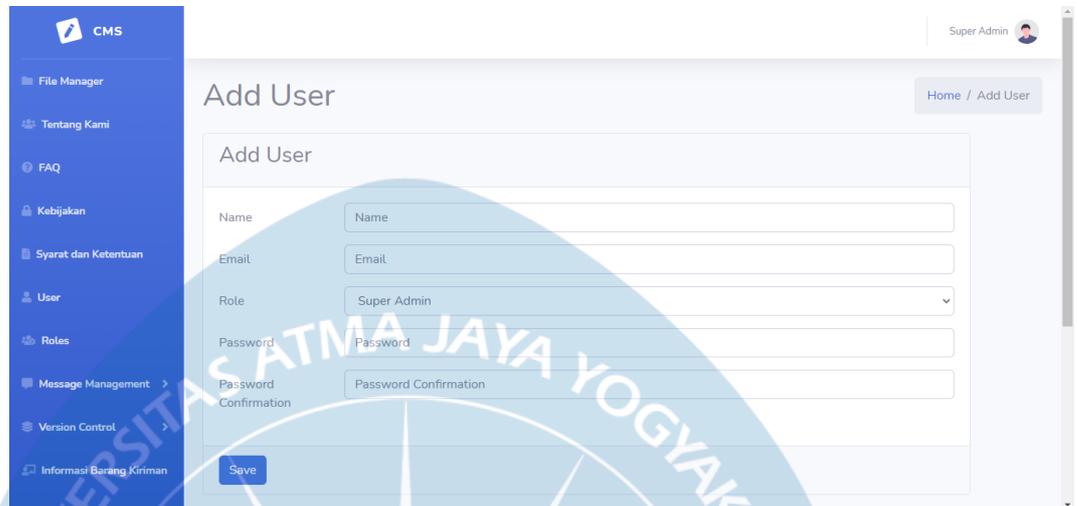
#### 5.3.9.1. Antarmuka *index()*



**Gambar 50.** Tampilan Antarmuka UserUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 50 berfungsi untuk menampilkan data *user* yang tersimpan pada *database*. Tombol “*Add Item*” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data *user* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *user* terpilih. Tombol “*Delete Selected*” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *user* terpilih.

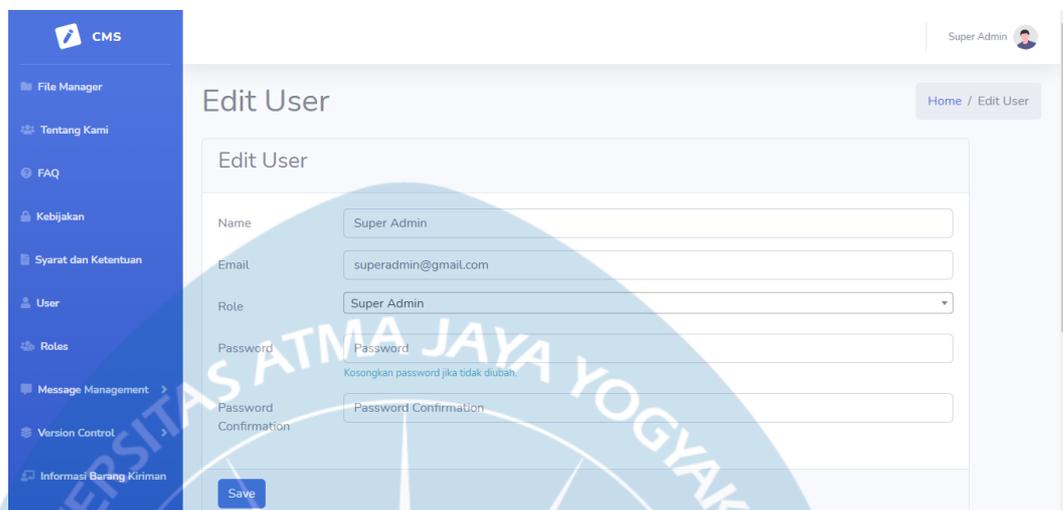
### 5.3.9.2. Antarmuka *add()*



**Gambar 51.** Tampilan Antarmuka UserUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 51 berfungsi untuk menambahkan *user* CMS yang baru. Bagian *role* pada halaman ini berupa *dropdown* yang berisi *role-role* yang tersimpan pada tabel *role*. Bagian *password confirmation* merupakan *form* yang berfungsi untuk mengkonfirmasi bahwa *password* yang diinput tepat.

### 5.3.9.3. Antarmuka *edit()*

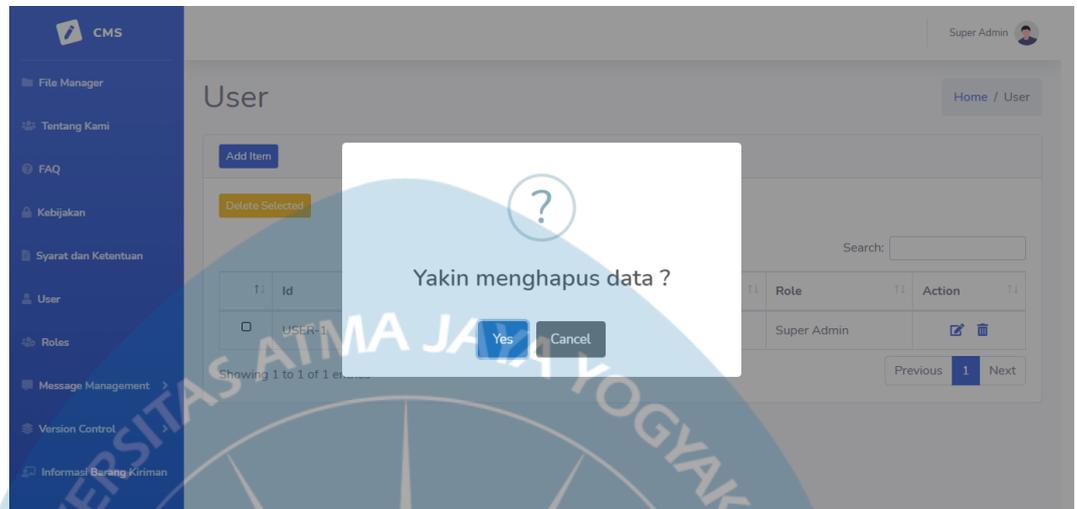


The screenshot displays the 'Edit User' page within a CMS. On the left, a blue sidebar lists various system functions. The main area features a form for editing user details. The form includes input fields for Name, Email, and Password, and a dropdown menu for Role. A 'Save' button is positioned at the bottom left of the form. The user 'Super Admin' is currently logged in, as shown in the top right corner.

**Gambar 52.** Tampilan Antarmuka UserUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 52 berfungsi untuk menampilkan dan mengubah data *user* terpilih. Bagian *password* dan *password confirmation* pada halaman ini dapat dikosongkan jika *user* tidak ingin mengubah *password user* terpilih.

#### 5.3.9.4. Antarmuka *delete()*

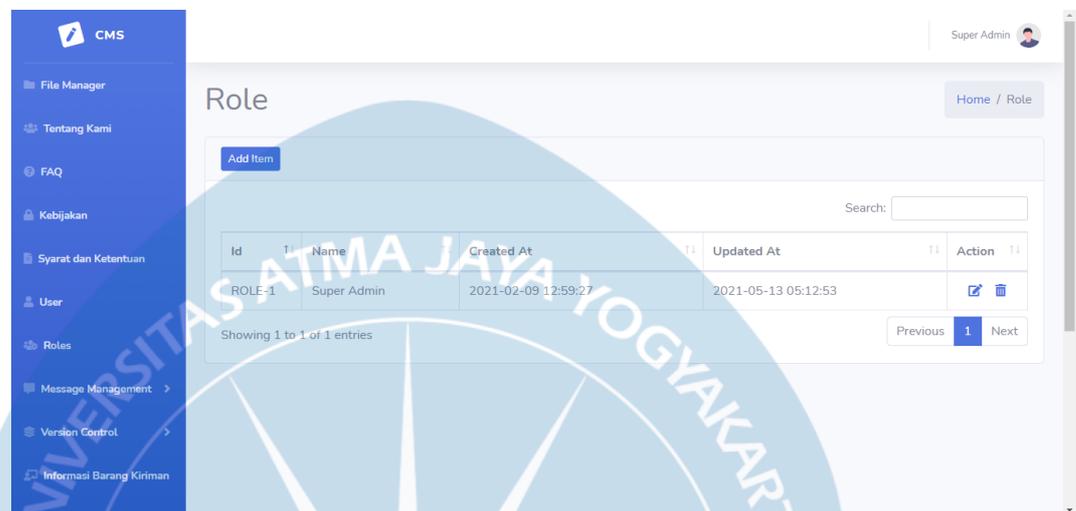


**Gambar 53.** Tampilan Antarmuka UserUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 53 berfungsi sebagai halaman konfirmasi penghapusan data *user*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data *user* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan *user* terpilih.

### 5.3.10 Antarmuka RoleUI

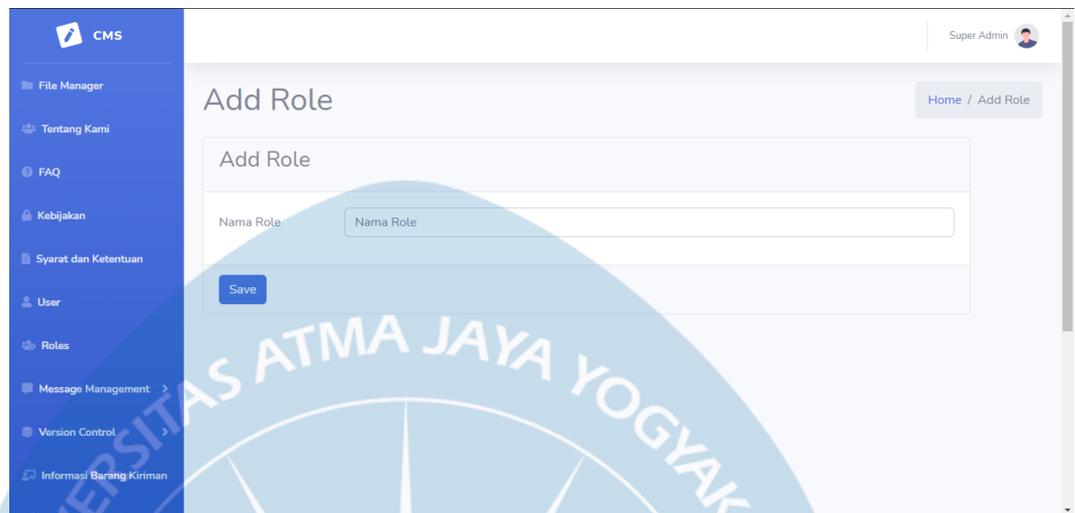
#### 5.3.10.1. Antarmuka *index()*



**Gambar 54.** Tampilan Antarmuka RoleUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 54 berfungsi untuk menampilkan semua *role* yang terdapat di *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()* untuk menambah *role* baru. Tombol berlogo pensil berfungsi untuk masuk ke halaman *edit role* untuk meng-*edit* detail dari *role* terpilih. Tombol berlogo tempat sampah berfungsi untuk menghapus *role* terpilih.

### 5.3.10.2. Antarmuka *add()*

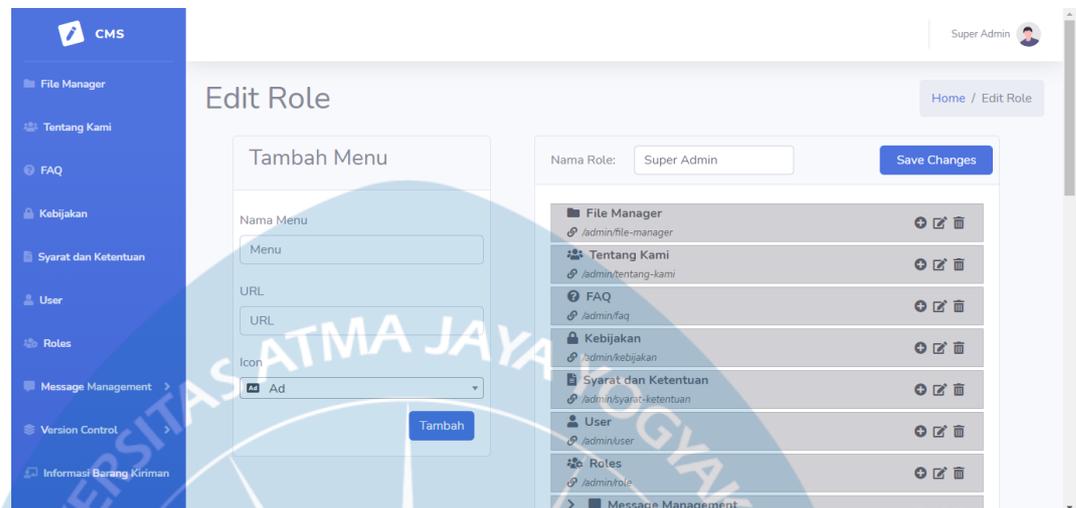


The screenshot displays a web application interface for adding a new role. On the left, a blue sidebar contains a 'CMS' logo and a list of navigation items: File Manager, Tentang Kami, FAQ, Kebijakan, Syarat dan Ketentuan, User, Roles, Message Management, Version Control, and Informasi Barang Kiriman. The main content area is titled 'Add Role' and includes a breadcrumb trail 'Home / Add Role'. The form contains a 'Nama Role' label and a corresponding text input field. Below the input field is a blue 'Save' button.

**Gambar 55.** Tampilan Antarmuka RoleUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 55 berfungsi untuk menambahkan data *role* baru. Tombol “*Save*” digunakan untuk menyimpan data *role* baru ke dalam *database*.

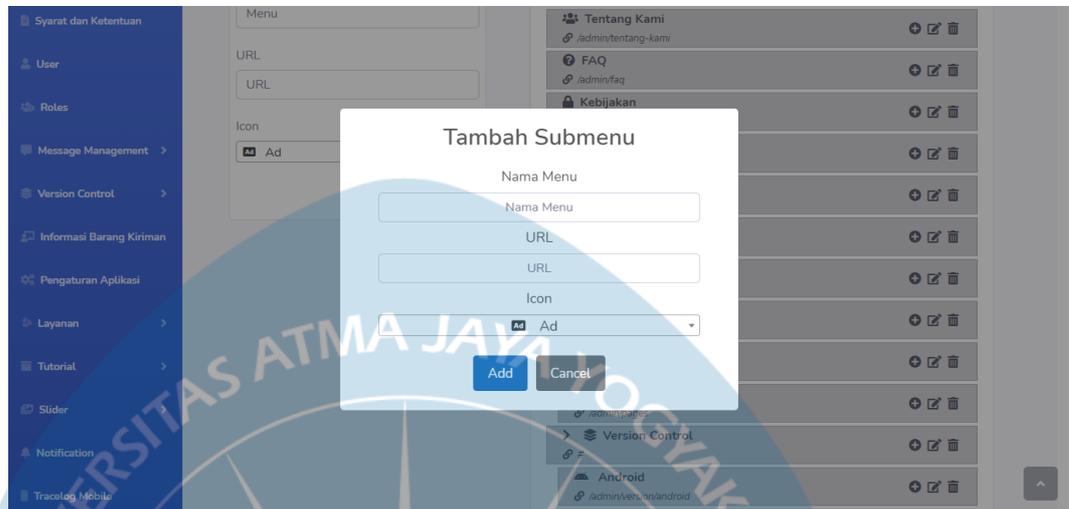
### 5.3.10.3. Antarmuka *edit()*



Gambar 56. Tampilan Antarmuka RoleUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 56 berfungsi untuk menampilkan dan mengubah nama *role*, serta menu yang dapat diakses oleh *role* terpilih. Bagian tambah menu dan tombol “Tambah” berfungsi untuk menambah menu baru yang dapat diakses oleh *role* terpilih. Bagian *Icon* pada bagian tambah menu berbentuk *dropdown* dan berisi daftar *icon* yang dapat dipilih untuk ditampilkan sebagai *icon* masing-masing menu. Bagian nama *role* berfungsi untuk menampilkan dan mengubah nama dari *role* terpilih. Pada bagian di bawah nama *role*, terdapat daftar menu dan submenu yang dapat diakses oleh *role* terpilih. Pada masing-masing menu terdapat tombol berlogo tambah, yang berfungsi untuk menambahkan submenu baru ke dalam menu terpilih, tombol berlogo pensil untuk mengubah detail dari menu/submenu terpilih, dan tombol berlogo tempat sampah yang berfungsi untuk menghapus menu/submenu terpilih. Menu-menu yang terdapat pada bagian bawah nama *role* dapat digeser untuk diubah urutannya. Tombol “Save Changes” berfungsi untuk menyimpan perubahan dari urutan menu, serta nama *role* yang diubah.

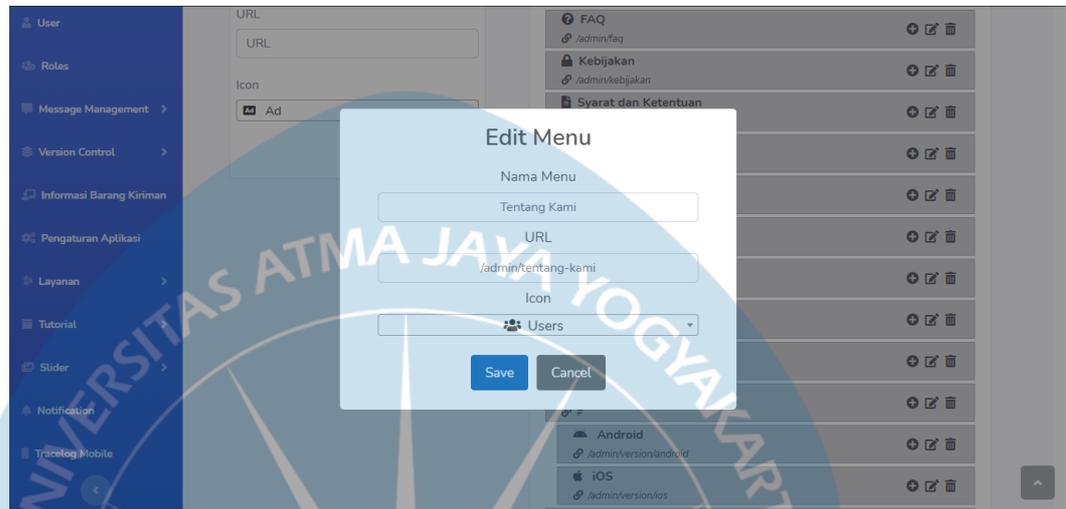
#### 5.3.10.4. Antarmuka addsubmenu()



**Gambar 57.** Tampilan Antarmuka RoleUI Halaman addsubmenu()

Halaman `addsubmenu()` yang ditampilkan pada Gambar 57 merupakan kotak *dialog* yang berfungsi untuk menambahkan submenu baru ke dalam menu terpilih. Bagian *Icon* pada halaman ini berupa *dropdown* dan berisi semua *icon-icon* yang dapat dipilih untuk ditampilkan sebagai *icon* menu. Tombol “Add” berfungsi untuk menambahkan submenu ke dalam *database*. Tombol “Cancel” berfungsi untuk membatalkan proses tambah submenu dan kembali ke halaman *edit()*.

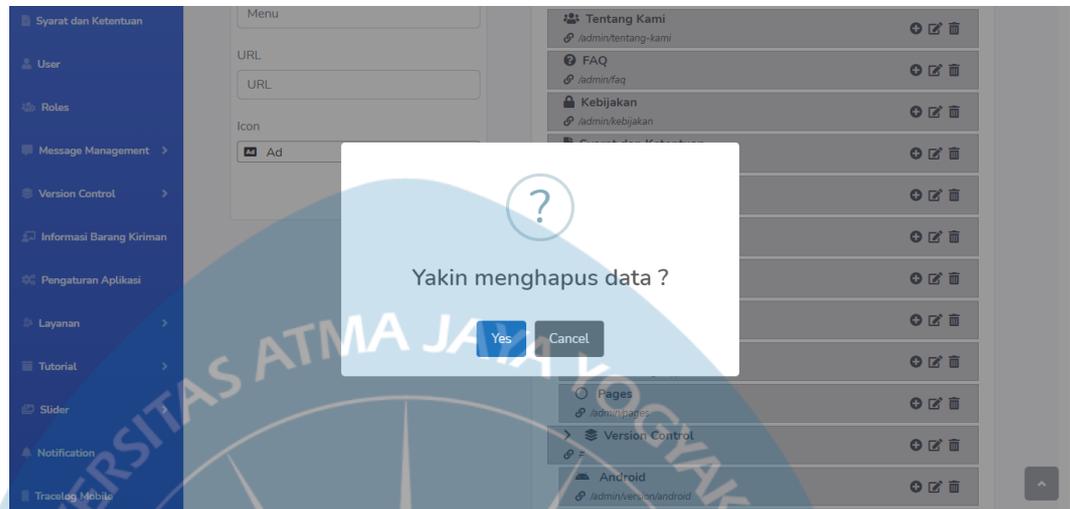
### 5.3.10.5. Antarmuka editmenu()



**Gambar 58.** Tampilan Antarmuka RoleUI Halaman editmenu()

Halaman *editmenu()* yang ditampilkan pada Gambar 58 merupakan sebuah kotak *dialog* yang berfungsi untuk menampilkan dan meng-*edit* data menu/submenu terpilih. Tombol “*Save*” digunakan untuk menyimpan perubahan data menu/submenu ke dalam *database*. Tombol “*Cancel*” berfungsi untuk membatalkan proses *edit* menu dan kembali ke halaman *edit()*.

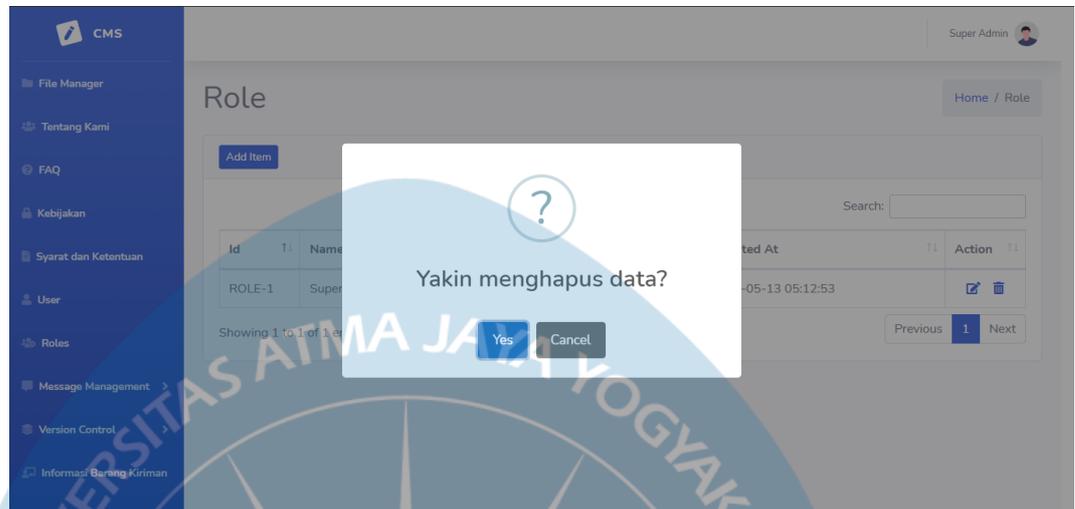
### 5.3.10.6. Antarmuka deletemenu()



**Gambar 59.** Tampilan Antarmuka RoleUI Halaman deletemenu()

Halaman `deletemenu()` yang ditampilkan pada Gambar 59 merupakan kotak *dialog* yang berfungsi sebagai halaman konfirmasi hapus menu/submenu terpilih. Menu yang mempunyai submenu tidak dapat dihapus hingga menu tidak lagi mempunyai submenu. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan menu/submenu dari *database*. Tombol “Cancel” berfungsi untuk membatalkan penghapusan menu/submenu dan kembali ke halaman *edit()*.

### 5.3.10.7. Antarmuka *delete()*

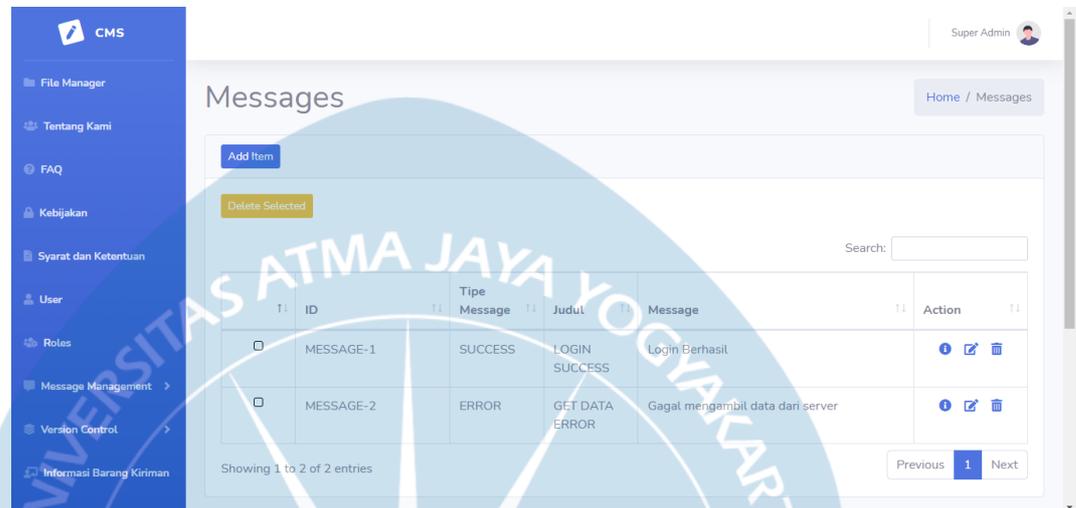


**Gambar 60.** Tampilan Antarmuka RoleUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 60 merupakan kotak *dialog* yang berfungsi sebagai halaman konfirmasi penghapusan *role* terpilih. Tombol “Yes” berfungsi untuk menghapus *role* terpilih dari *database*. Tombol “Cancel” berfungsi untuk membatalkan penghapusan menu terpilih dan kembali ke halaman *index()*.

### 5.3.11. Antarmuka MessageUI

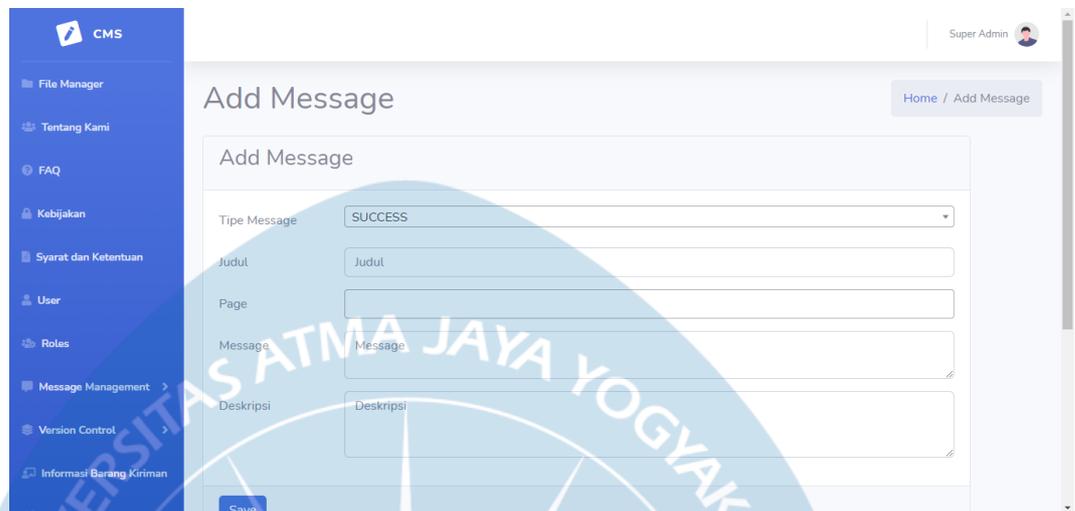
#### 5.3.11.1. Antarmuka *index()*



**Gambar 61.** Tampilan Antarmuka MessageUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 61 berfungsi untuk menampilkan data *message* yang tersimpan pada *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data *message* baru. Tombol dengan logo “i” berfungsi untuk menampilkan halaman *detail()* yang berfungsi untuk menampilkan data detail dari *message* terpilih. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *message* terpilih. Tombol “Delete Selected” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *message* terpilih.

### 5.3.11.2. Antarmuka *add()*

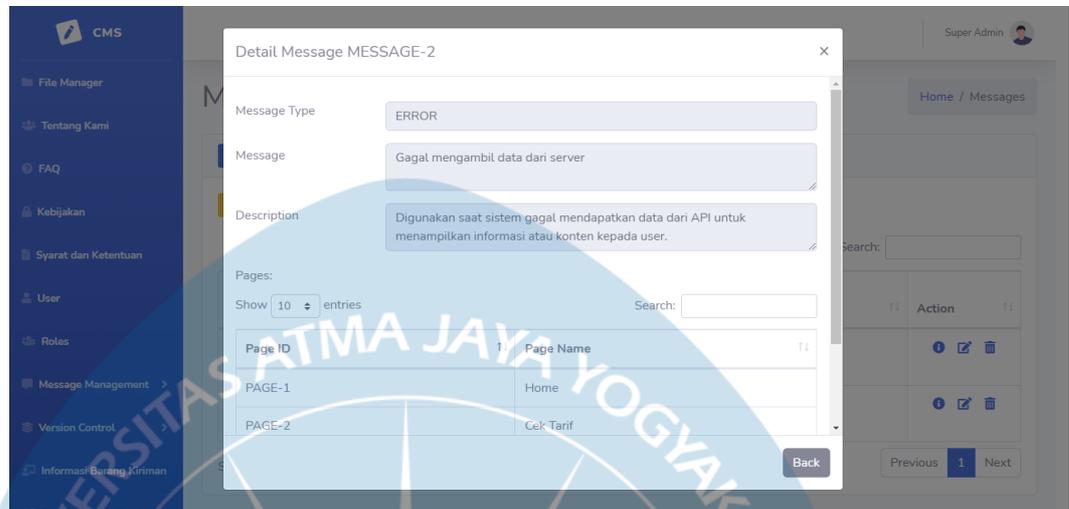


The screenshot shows a web application interface for adding a message. On the left is a blue sidebar menu with options like 'File Manager', 'Tentang Kami', 'FAQ', 'Kebijakan', 'Syarat dan Ketentuan', 'User', 'Roles', 'Message Management', 'Version Control', and 'Informasi Barang Kiriman'. The main content area is titled 'Add Message' and contains a form with the following fields: 'Type Message' (a dropdown menu currently showing 'SUCCESS'), 'Judul' (a text input field), 'Page' (a select multiple dropdown), 'Message' (a text area), and 'Deskripsi' (a text area). A 'Save' button is located at the bottom left of the form. In the top right corner, there is a user profile for 'Super Admin' and a breadcrumb trail 'Home / Add Message'.

**Gambar 62.** Tampilan Antarmuka MessageUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 62 berfungsi untuk menambahkan data *message* baru. Bagian tipe *message* pada halaman ini berbentuk *dropdown* yang berisi tipe-tipe *message* yang terdapat pada tabel *message\_type*. Bagian page pada halaman ini berupa select multiple yang berisi halaman-halaman yang terdapat pada tabel page. Tombol “Save” berfungsi untuk menambahkan data *message* baru ke dalam *database*.

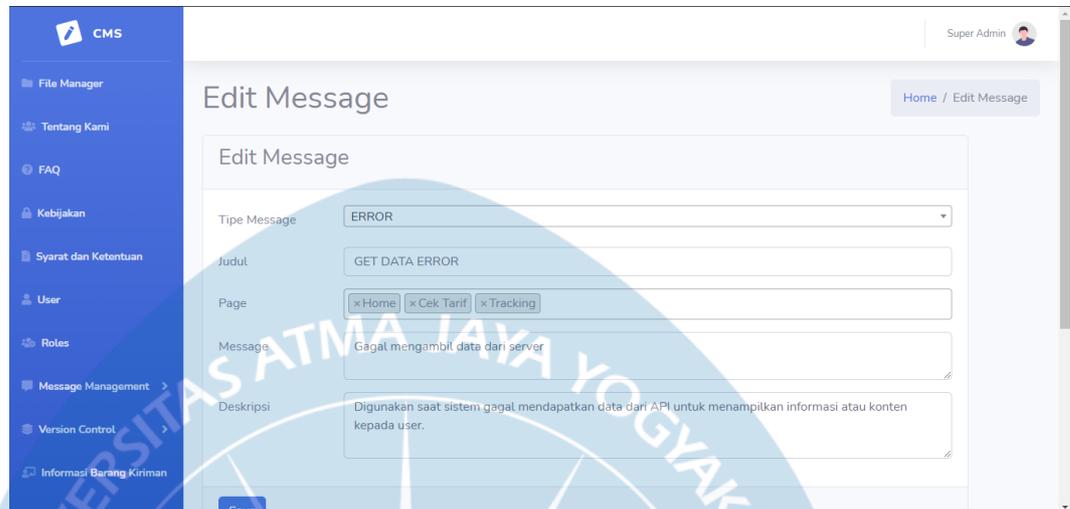
### 5.3.11.3. Antarmuka detail()



**Gambar 63.** Tampilan Antarmuka MessageUI Halaman detail()

Halaman detail() yang ditampilkan pada Gambar 63 merupakan sebuah kotak *dialog* yang berfungsi untuk menampilkan data detail dari *message* terpilih. Tombol “Back” berfungsi untuk menutup halaman detail() dan kembali ke halaman *index*).

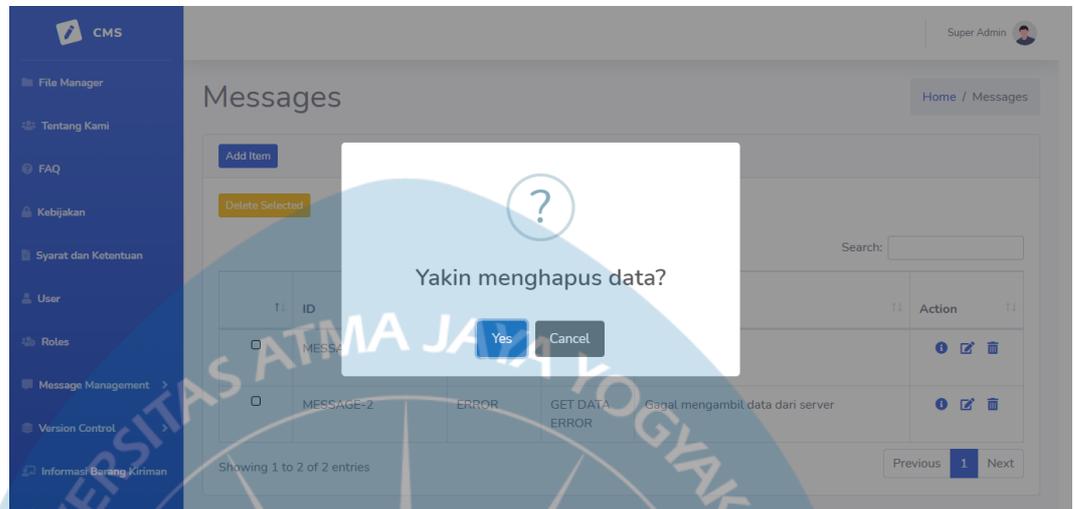
#### 5.3.11.4. Antarmuka *edit()*



**Gambar 64.** Tampilan Antarmuka MessageUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 64 berfungsi untuk menampilkan dan mengubah data *message* terpilih. Bagian tipe *message* pada halaman ini berbentuk *dropdown* yang berisi tipe-tipe *message* yang terdapat dalam tabel *message\_type*. Bagian page pada halaman ini berbentuk *select multiple* yang berisi halaman-halaman yang terdapat dalam tabel page. Tombol “*Save*” berfungsi untuk menyimpan perubahan data *message* terpilih.

### 5.3.11.5. Antarmuka *delete()*

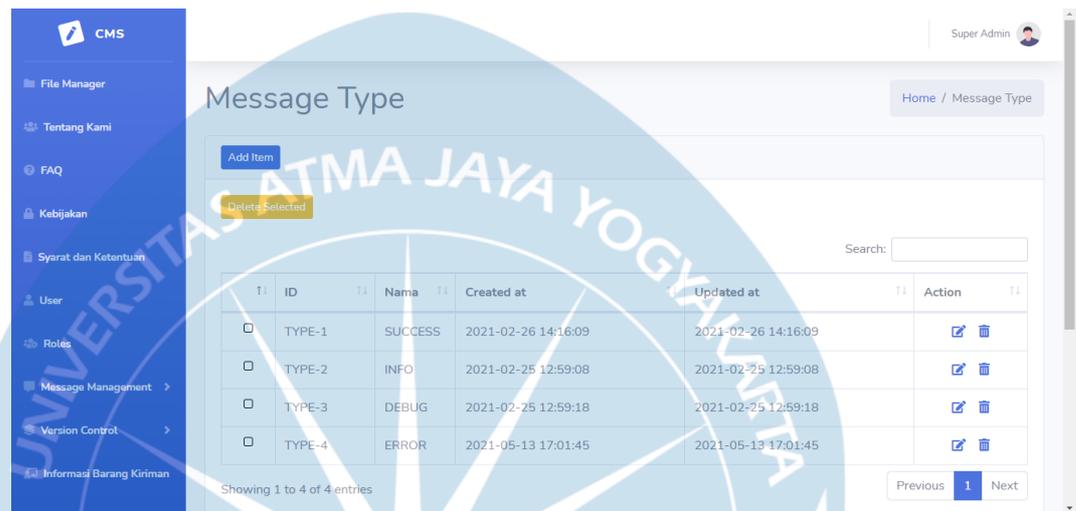


**Gambar 65.** Tampilan Antarmuka MessageUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 65 berfungsi sebagai halaman konfirmasi penghapusan data *user*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data *user* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan *user* terpilih.

### 5.3.12. Antarmuka MessageTypeUI

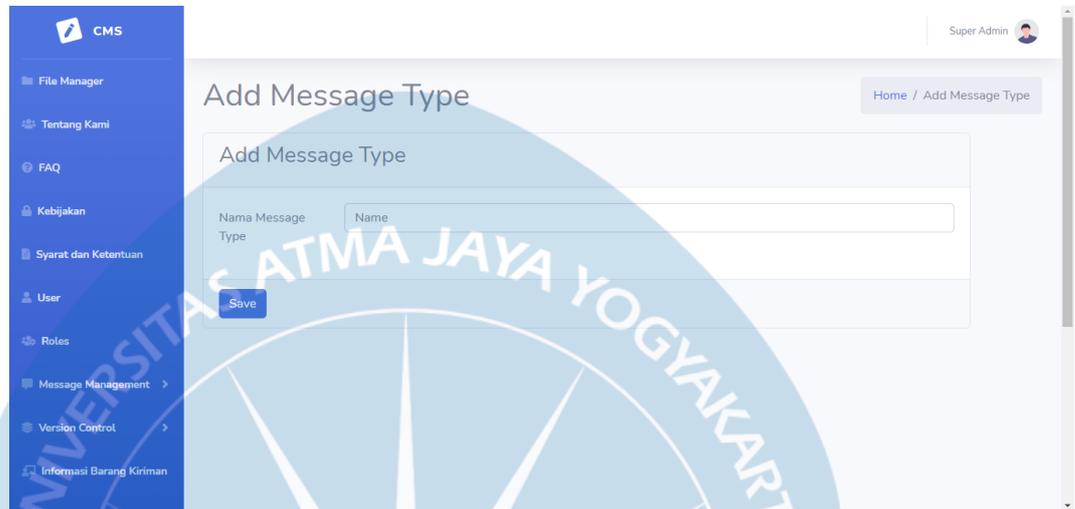
#### 5.3.12.1. Antarmuka *index()*



**Gambar 66.** Tampilan Antarmuka MessageTypeUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 66 berfungsi untuk menampilkan data tipe *message* yang tersimpan pada *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data tipe *message* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data tipe *message* terpilih. Tombol “Delete Selected” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus tipe *message* terpilih.

### 5.3.12.2. Antarmuka *add()*

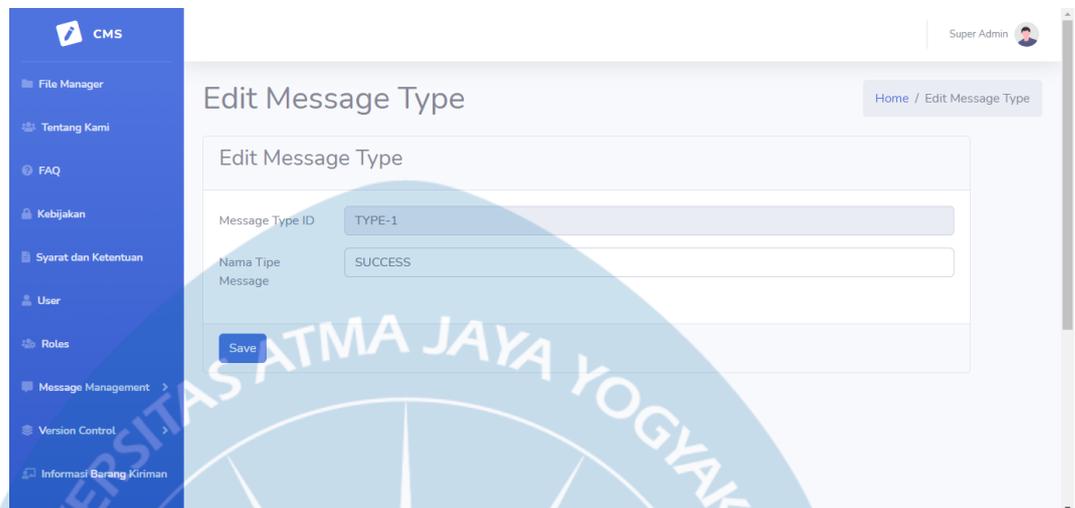


The screenshot shows a web application interface for adding a new message type. On the left is a blue sidebar menu with the CMS logo and various navigation options. The main content area is titled 'Add Message Type' and includes a breadcrumb trail 'Home / Add Message Type'. The form itself has a title 'Add Message Type' and a large text input field. Below this, there is a label 'Name Message Type' and a smaller 'Name' input field. A 'Save' button is positioned at the bottom left of the form area. The user 'Super Admin' is logged in, as indicated in the top right corner.

**Gambar 67.** Tampilan Antarmuka MessageTypeUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 67 berfungsi untuk menambahkan data tipe *message* baru. Tombol “*Save*” berfungsi untuk menambah data tipe *message* baru ke dalam *database*.

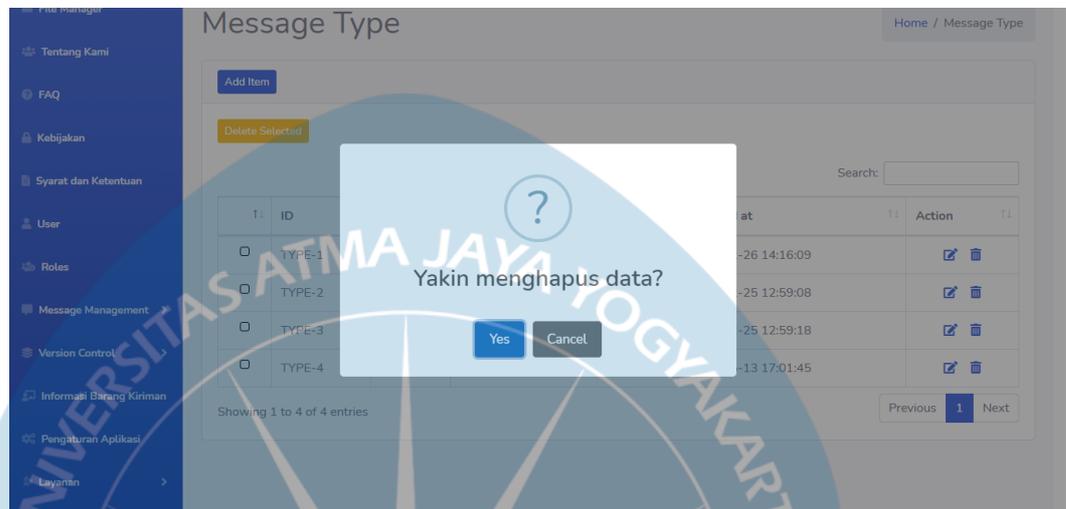
### 5.3.12.3. Antarmuka *edit()*



**Gambar 68.** Tampilan Antarmuka MessageTypeUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 68 berfungsi untuk menampilkan dan mengubah data tipe *message* terpilih. Bagian *Message Type ID* berfungsi hanya untuk menampilkan ID tipe *message* dan tidak bisa *di-edit*. Tombol “*Save*” berfungsi untuk menyimpan perubahan pada tipe *message* terpilih.

#### 5.3.12.4. Antarmuka *delete()*



**Gambar 69.** Tampilan Antarmuka MessageTypeUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 69 berfungsi sebagai halaman konfirmasi penghapusan data tipe *message*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data tipe *message* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan tipe *message* terpilih. Tipe *message* yang masih digunakan pada sebuah *message* tidak dapat dihapus.

### 5.3.13. Antarmuka PageUI

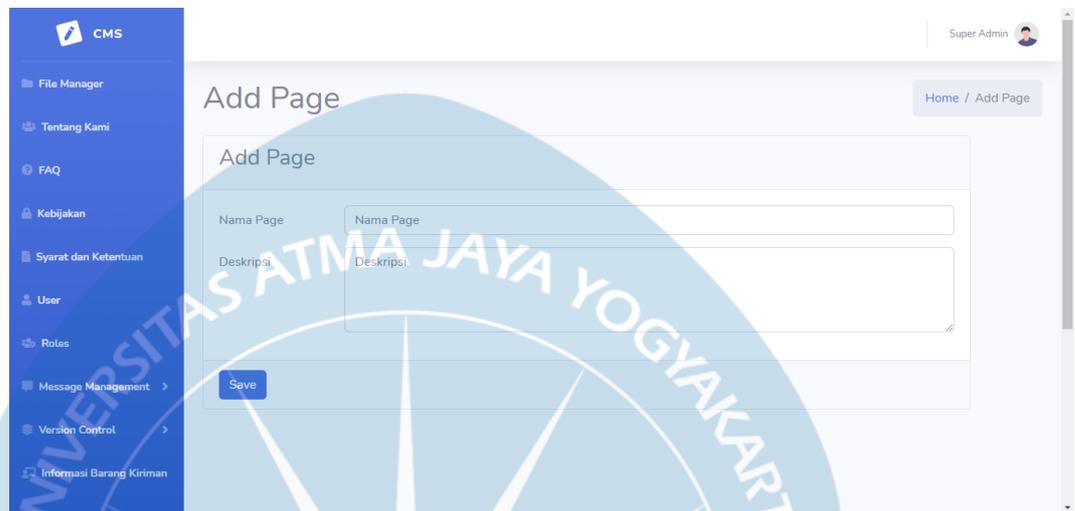
#### 5.3.13.1. Antarmuka *index()*



**Gambar 70.** Tampilan Antarmuka PageUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 70 berfungsi untuk menampilkan data halaman-halaman yang tersimpan pada *database*. Tombol “*Add Item*” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data halaman baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data halaman terpilih. Tombol “*Delete Selected*” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus halaman terpilih.

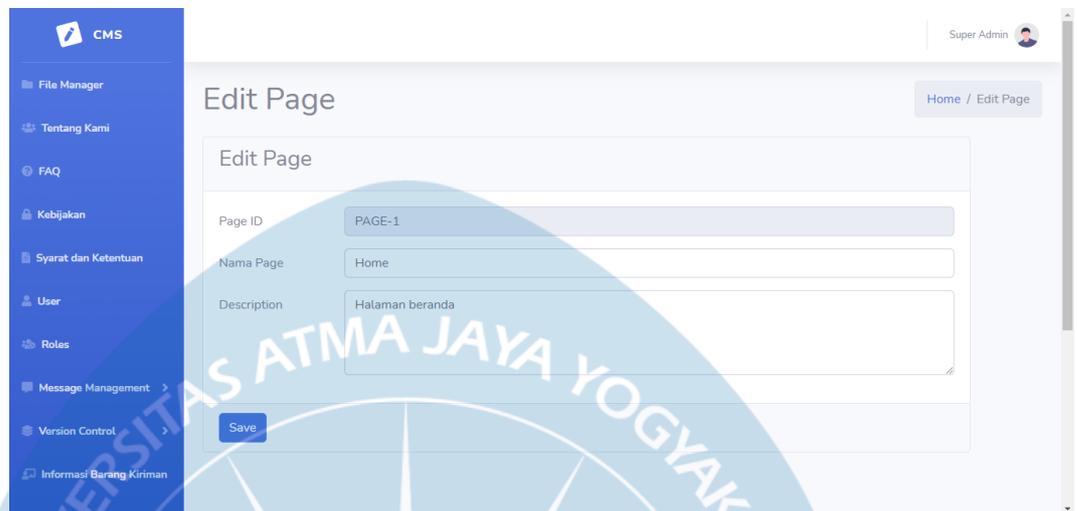
### 5.3.13.2. Antarmuka *add()*



**Gambar 71.** Tampilan Antarmuka PageUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 71 berfungsi untuk menambahkan data halaman baru. Tombol “*Save*” berfungsi untuk menambahkan halaman baru ke dalam *database*.

### 5.3.13.3. Antarmuka *edit()*

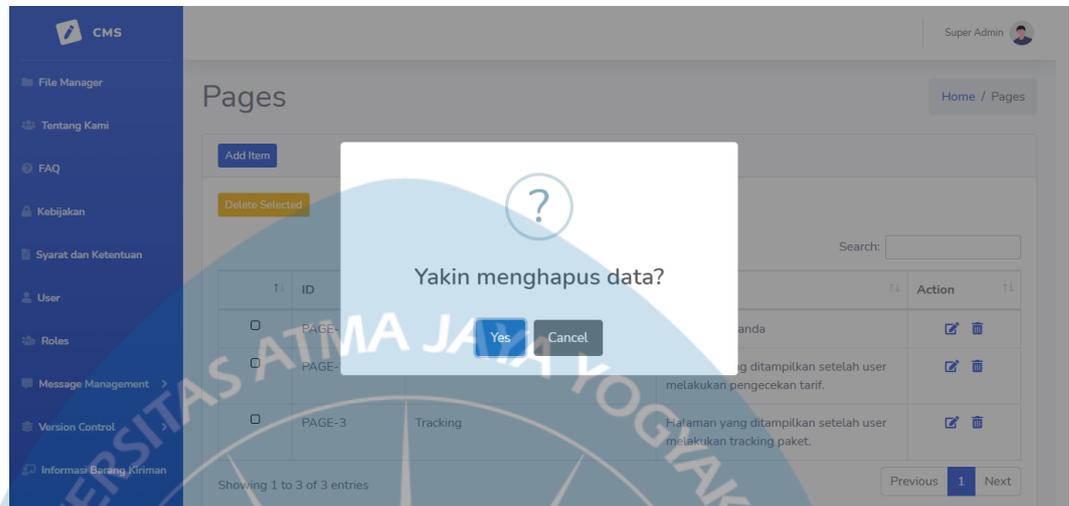


The screenshot displays the 'Edit Page' interface of a CMS. On the left is a blue sidebar menu with various management options. The main area is titled 'Edit Page' and contains a form with three input fields: 'Page ID' (containing 'PAGE-1'), 'Nama Page' (containing 'Home'), and 'Description' (containing 'Halaman beranda'). A 'Save' button is positioned below the form. The top right corner shows the user 'Super Admin' and a breadcrumb 'Home / Edit Page'.

**Gambar 72.** Tampilan Antarmuka PageUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 72 berfungsi untuk menampilkan dan mengubah data halaman terpilih. Bagian Page ID pada halaman ini hanya berfungsi untuk menampilkan ID halaman dan tidak bisa *di-edit*. Tombol “Save” berfungsi untuk menyimpan perubahan terhadap halaman terpilih.

#### 5.3.13.4. Antarmuka *delete()*



**Gambar 73.** Tampilan Antarmuka PageUI Halaman *delete()*

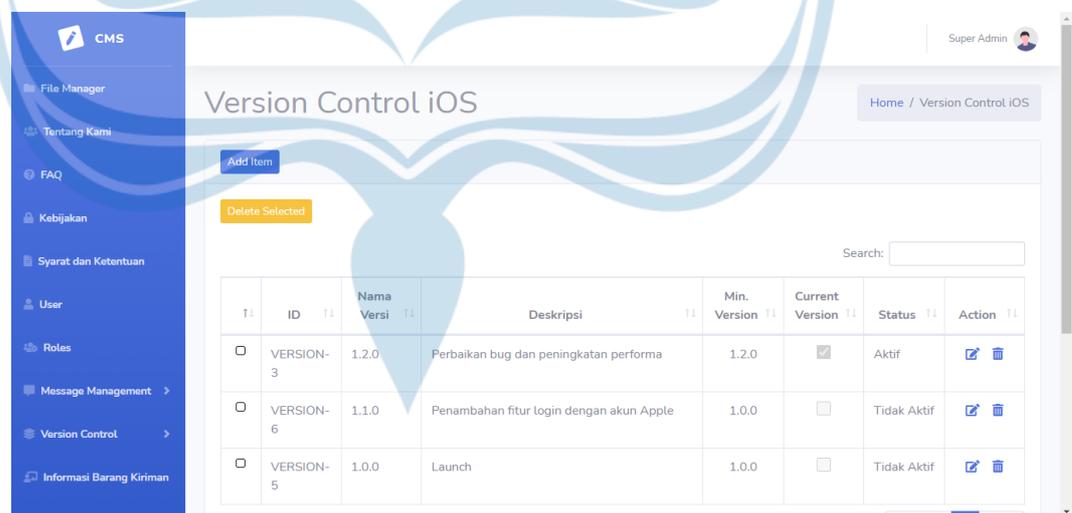
Halaman *delete()* yang ditampilkan pada Gambar 73 berfungsi sebagai halaman konfirmasi penghapusan data halaman. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data halaman terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan halaman terpilih.

### 5.3.14. Antarmuka VersionUI

#### 5.3.14.1. Antarmuka *index()*



Gambar 74. Tampilan Antarmuka VersionUI Halaman *index()* versi Android



Gambar 75. Tampilan Antarmuka VersionUI Halaman *index()* versi iOS

Halaman *index()* yang ditampilkan pada Gambar 74 dan Gambar 75 berfungsi untuk menampilkan data *version* yang tersimpan pada *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()*

untuk menambahkan data *version* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *version* terpilih. Tombol “Delete Selected” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *version* terpilih.

### 5.3.14.2. Antarmuka *add()*

The screenshot shows the 'Add Version' form in the CMS interface. The form is titled 'Add Version' and is located in the 'Version Control' section of the CMS. The form includes the following fields and options:

- Nama Versi:** A text input field with the placeholder text 'Nama Versi'.
- Deskripsi:** A text area with the placeholder text 'Deskripsi'.
- Platform:** A dropdown menu with 'Android' selected.
- Versi Minimum:** A text input field with the placeholder text 'Versi Minimum'.
- Atribut:** Two checkboxes: 'Current Version' and 'Aktif', both of which are currently unchecked.

A 'Save' button is located at the bottom left of the form. The CMS interface also shows a sidebar with various menu items and a top navigation bar with the user 'Super Admin'.

**Gambar 76.** Tampilan Antarmuka VersionUI Halaman *add()* versi Android

The screenshot shows the 'Add Version' form in the CMS interface for the iOS platform. The form is titled 'Add Version' and is located in the 'Version Control' section of the CMS. The form includes the following fields and options:

- Nama Versi:** A text input field with the placeholder text 'Nama Versi'.
- Deskripsi:** A text area with the placeholder text 'Deskripsi'.
- Platform:** A dropdown menu with 'iOS' selected.
- Versi Minimum:** A text input field with the placeholder text 'Versi Minimum'.
- Atribut:** Two checkboxes: 'Current Version' and 'Aktif', both of which are currently unchecked.

A 'Save' button is located at the bottom left of the form. The CMS interface also shows a sidebar with various menu items and a top navigation bar with the user 'Super Admin'.

**Gambar 77.** Tampilan Antarmuka VersionUI Halaman *add()* versi iOS

Halaman *add()* yang ditampilkan pada Gambar 76 dan Gambar 77 berfungsi untuk menambahkan data *version* baru. Bagian *platform* pada

halaman ini hanya berfungsi untuk menampilkan *platform* terpilih dan tidak bisa *di-edit*. Jika atribut “Current Version” terpilih, maka atribut aktif juga terpilih dan tidak dapat dihilangkan. Tombol “Save” digunakan untuk menambahkan versi baru ke dalam *database*.

### 5.3.14.3. Antarmuka *edit()*

The screenshot shows the 'Edit Version' form for an Android version. The form fields are as follows:

ID	VERSION-4
Nama Versi	1.0.0
Deskripsi	Launch aplikasi
Platform	Android
Versi Minimum	1.0.0
Atribut	<input type="checkbox"/> Aktif <input type="checkbox"/> Current Version

Gambar 78. Tampilan Antarmuka VersionUI Halaman *edit()* versi Android

The screenshot shows the 'Edit Version' form for an iOS version. The form fields are as follows:

ID	VERSION-3
Nama Versi	1.2.0
Deskripsi	Perbaiki bug dan peningkatan performa
Platform	iOS
Versi Minimum	1.2.0
Atribut	<input checked="" type="checkbox"/> Aktif <input checked="" type="checkbox"/> Current Version

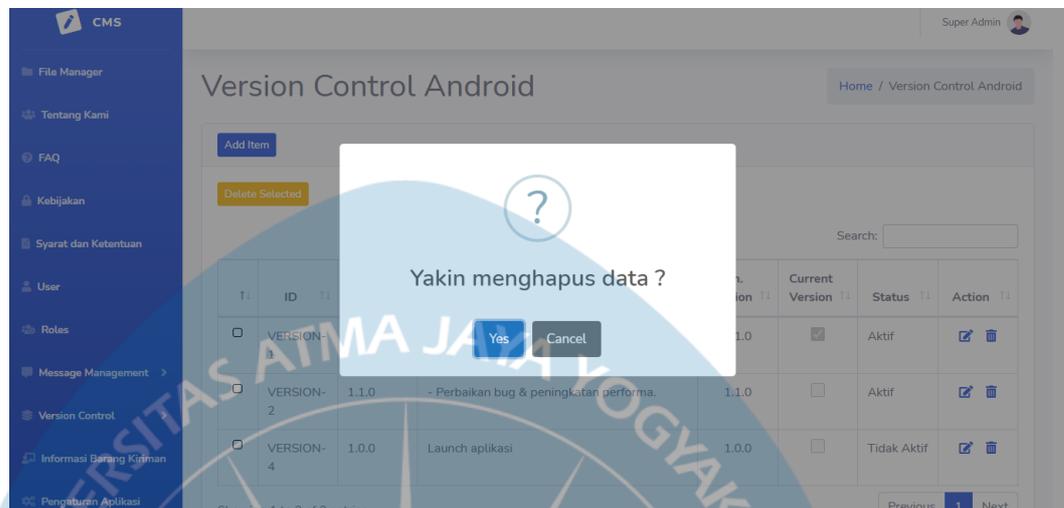
Gambar 79. Tampilan Antarmuka VersionUI Halaman *edit()* versi iOS

Halaman *edit()* yang ditampilkan pada Gambar 78 dan Gambar 79 berfungsi untuk menampilkan dan mengubah data *version* terpilih. Bagian

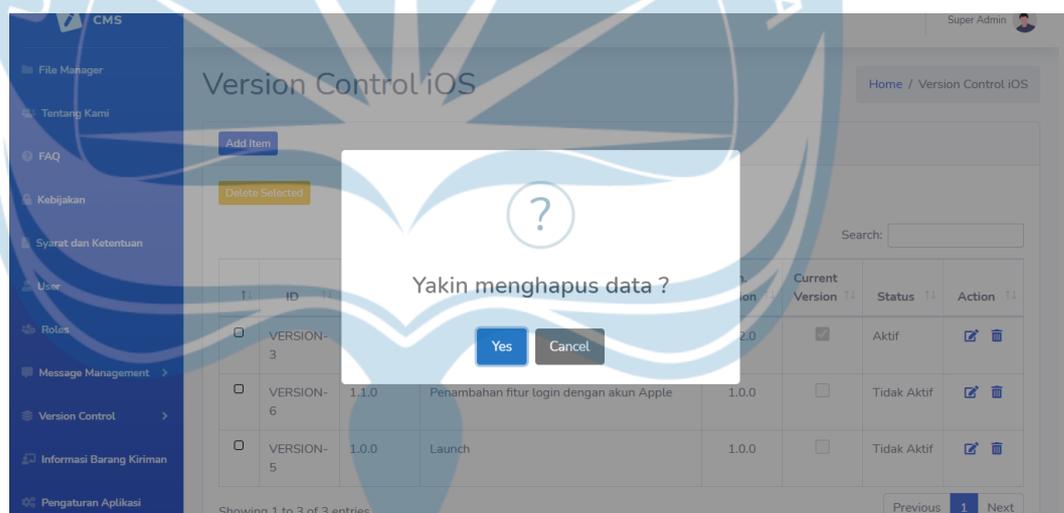
ID dan *Platform* pada halaman ini hanya berfungsi untuk menampilkan data saja dan tidak bisa *di-edit*. Versi yang masih menjadi *current version* tidak bisa diubah atributnya hingga versi lain dibuat menjadi *current version*. Tombol “*Save*” berfungsi untuk menyimpan perubahan data dari versi terpilih.



#### 5.3.14.4. Antarmuka *delete()*



Gambar 80. Tampilan Antarmuka VersionUI Halaman *delete()* versi Android

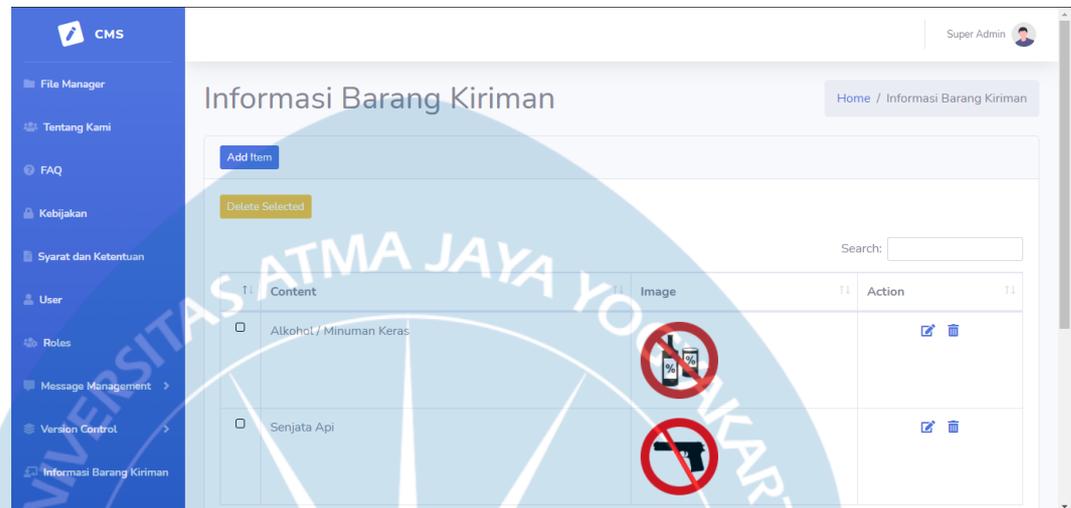


Gambar 81. Tampilan Antarmuka VersionUI Halaman *delete()* versi iOS

Halaman *delete()* yang ditampilkan pada Gambar 80 dan Gambar 81 berfungsi sebagai halaman konfirmasi penghapusan data *version*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data *version* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan *version* terpilih. *Version* yang masih berstatus aktif tidak bisa dihapus hingga status *version* diubah menjadi tidak aktif.

### 5.3.15. Antarmuka InformasiBarangKirimanUI

#### 5.3.15.1. Antarmuka *index()*



**Gambar 82.** Tampilan Antarmuka InformasiBarangKirimanUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 82 berfungsi untuk menampilkan data informasi barang kiriman yang tersimpan pada *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data informasi barang kiriman baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data informasi barang kiriman terpilih. Tombol “Delete Selected” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus informasi barang kiriman terpilih.

### 5.3.15.2. Antarmuka *add()*

The image shows a screenshot of a CMS interface. On the left is a blue sidebar menu with the following items: File Manager, Tentang Kami, FAQ, Kebijakan, Syarat dan Ketentuan, User, Roles, Message Management, Version Control, and Informasi Barang Kiriman. The main content area is titled 'Add Informasi Barang Kiriman' and contains a form with the following fields: a title field 'Add Informasi Barang Kiriman', a 'Content' text area, an 'Image' field with a 'Choose' button, and a 'Save' button. The top right corner shows the user 'Super Admin' and a breadcrumb trail 'Home / Add Informasi Barang Kiriman'. A large, semi-transparent watermark of the Universitas Atma Jaya Yogyakarta logo is overlaid on the page.

**Gambar 83.** Tampilan Antarmuka InformasiBarangKirimanUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 83 berfungsi untuk menambahkan data informasi barang kiriman baru. Tombol “*Save*” digunakan untuk menambahkan informasi barang kiriman baru ke dalam *database*.

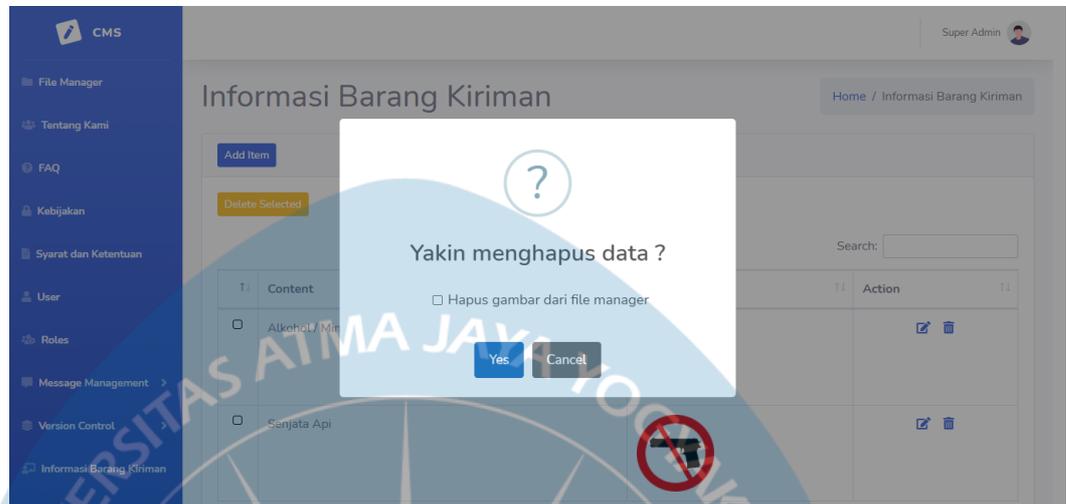
### 5.3.15.3. Antarmuka *edit()*



**Gambar 84.** Tampilan Antarmuka InformasiBarangKirimanUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 84 berfungsi untuk menampilkan dan mengubah data informasi barang kiriman terpilih. Tombol “*Save*” berfungsi untuk menyimpan perubahan data dari informasi barang kiriman terpilih.

### 5.3.15.3. Antarmuka *delete()*

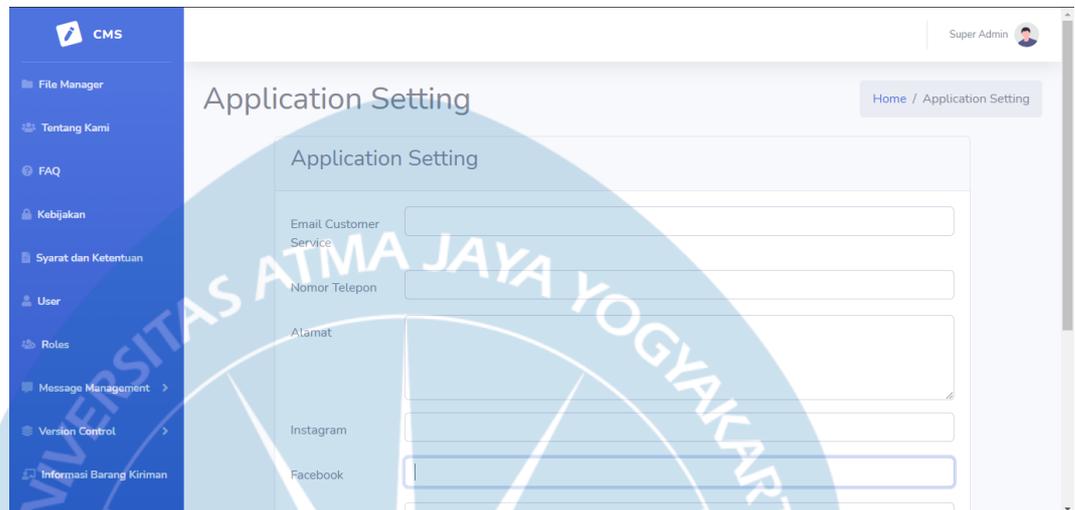


**Gambar 85.** Tampilan Antarmuka InformasiBarangKirimanUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 85 berfungsi sebagai halaman konfirmasi penghapusan data informasi barang kiriman. Tombol “*Yes*” berfungsi untuk mengkonfirmasi penghapusan data informasi barang kiriman terpilih. Tombol “*Cancel*” berfungsi untuk menutup dan membatalkan penghapusan informasi barang kiriman terpilih. Jika *checkboxlist* hapus gambar dari *file manager*, maka fungsi *delete* juga akan menghapus *file* gambar dari sistem penyimpanan.

### 5.3.16. Antarmuka SettingUI

#### 5.3.16.1. Antarmuka *index()*

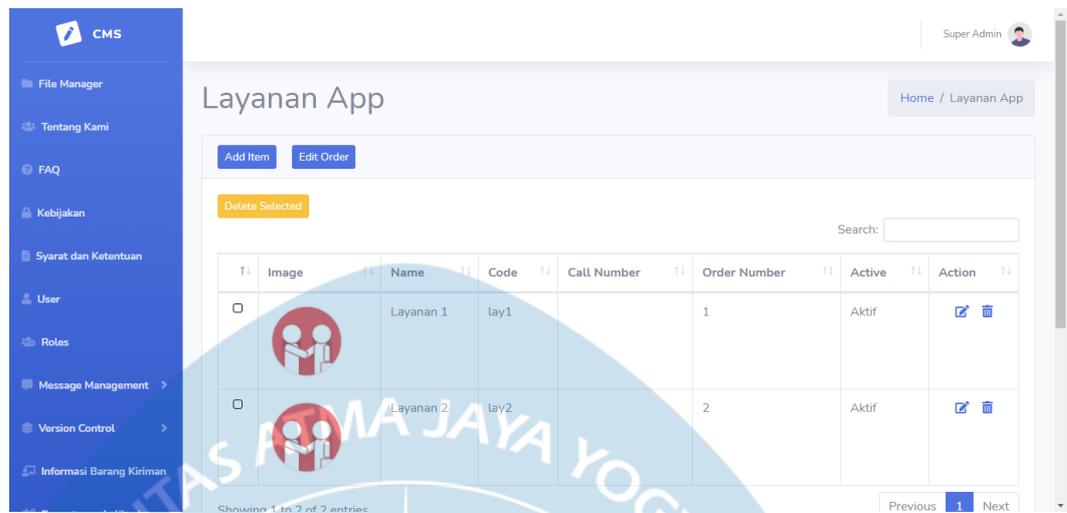


**Gambar 86.** Tampilan Antarmuka SettingUI Halaman *index()*

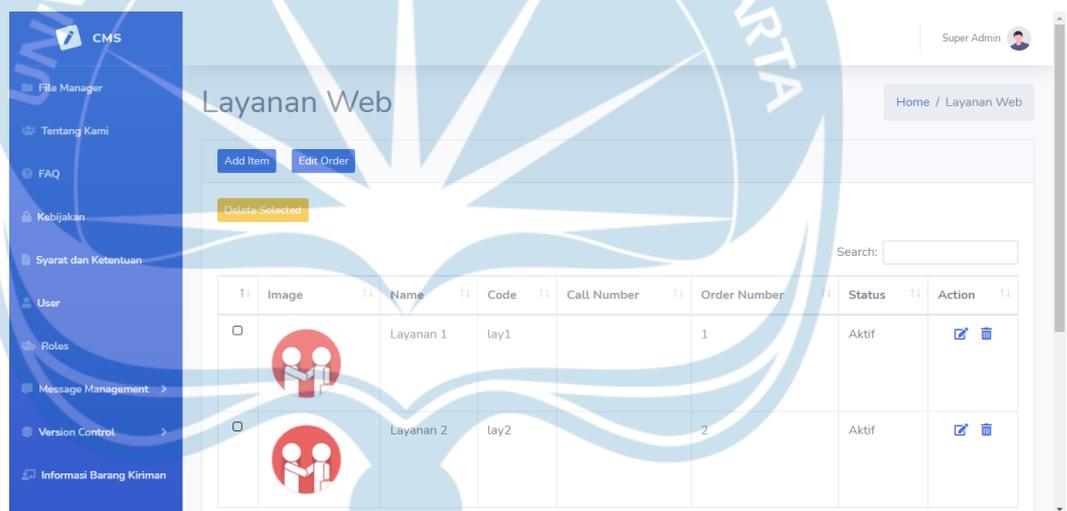
Halaman *index()* yang ditampilkan pada Gambar 86 berfungsi untuk menampilkan dan meng-*edit* data pengaturan aplikasi. Pada bagian bawah *form setting* aplikasi terdapat tombol “*Save*” yang berfungsi untuk menyimpan perubahan pada data pengaturan aplikasi pada *database*.

### 5.3.17. Antarmuka LayananUI

#### 5.3.17.1. Antarmuka *index()*



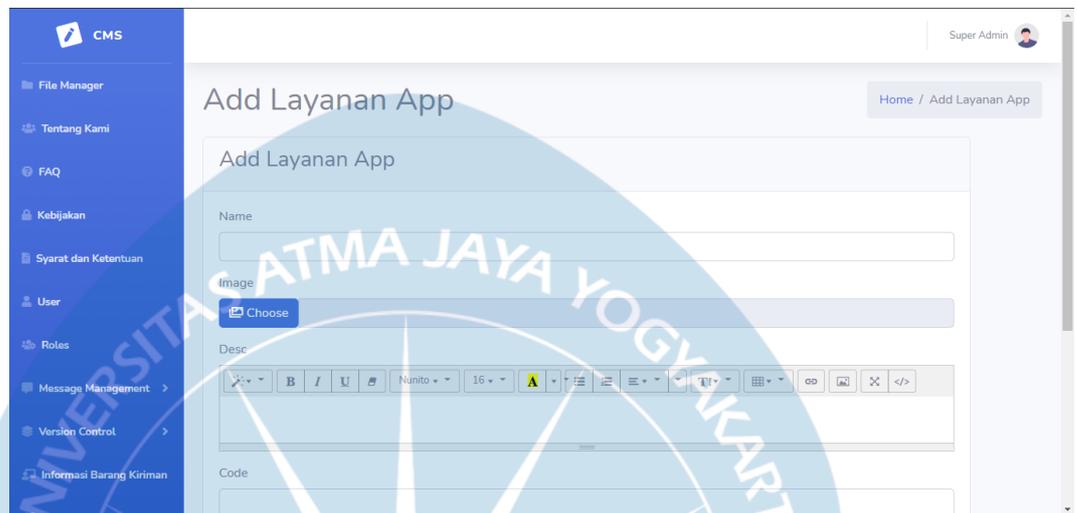
**Gambar 87.** Tampilan Antarmuka LayananUI Halaman *index()* versi *app*



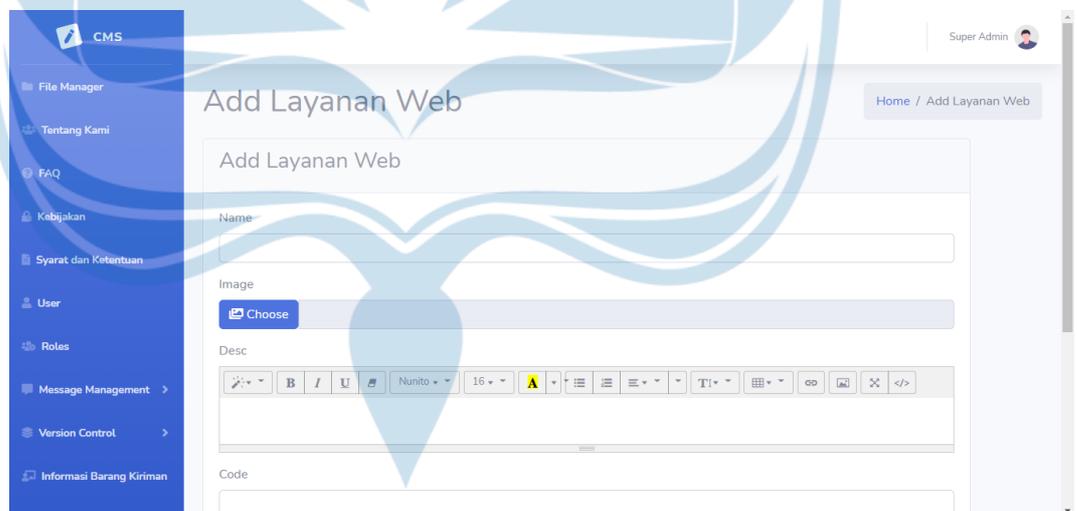
**Gambar 88.** Tampilan Antarmuka LayananUI Halaman *index()* versi *web*

Halaman *index()* yang ditampilkan pada Gambar 87 dan Gambar 88 berfungsi untuk menampilkan data layanan yang tersimpan pada *database*. Tombol “*Add Item*” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data layanan baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data layanan terpilih. Tombol “*Delete Selected*” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus layanan terpilih.

### 5.3.17.2. Antarmuka *add()*



**Gambar 89.** Tampilan Antarmuka LayananUI Halaman *add()* versi *app*

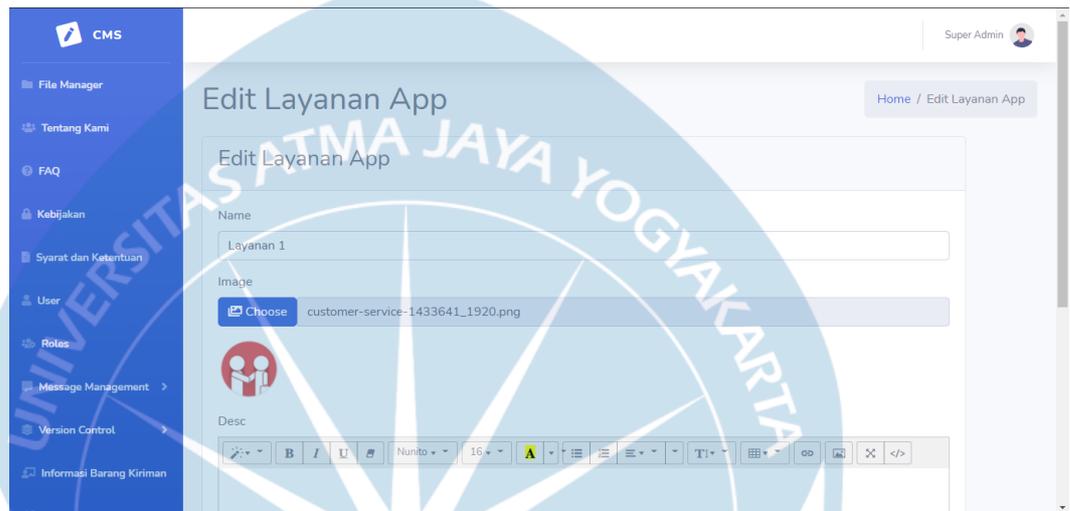


**Gambar 90.** Tampilan Antarmuka LayananUI Halaman *add()* versi *web*

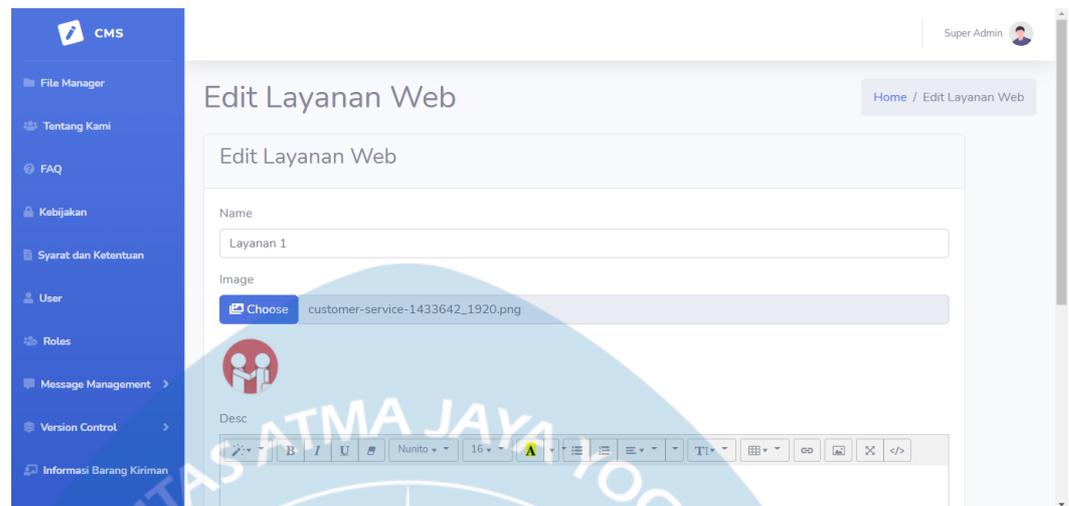
Halaman *add()* yang ditampilkan pada Gambar 89 dan Gambar 90 berfungsi untuk menambahkan data layanan baru. Bagian *platform* pada halaman ini hanya berfungsi untuk menampilkan *platform* terpilih dan tidak bisa *di-edit*. Bagian Desc pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Atribut “Active” merupakan *dropdown* berisi

pilihan “Yes” atau “No”. Jika atribut Active diisi dengan *Yes*, maka *order* number layanan adalah *order* number tertinggi + 1, sedangkan jika diisi dengan No, maka *order* number layanan adalah 0.

### 5.3.17.3. Antarmuka *edit()*



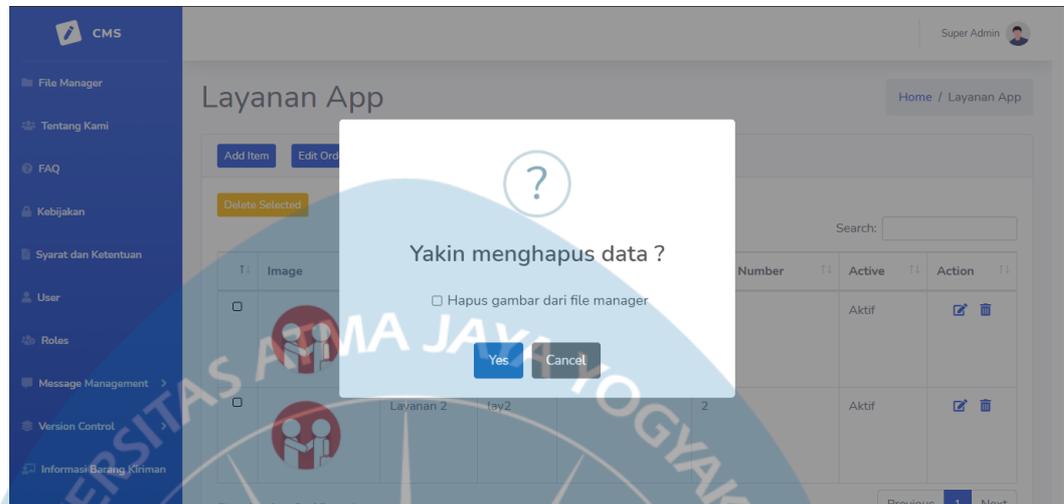
**Gambar 91.** Tampilan Antarmuka LayananUI Halaman *edit()* versi *app*



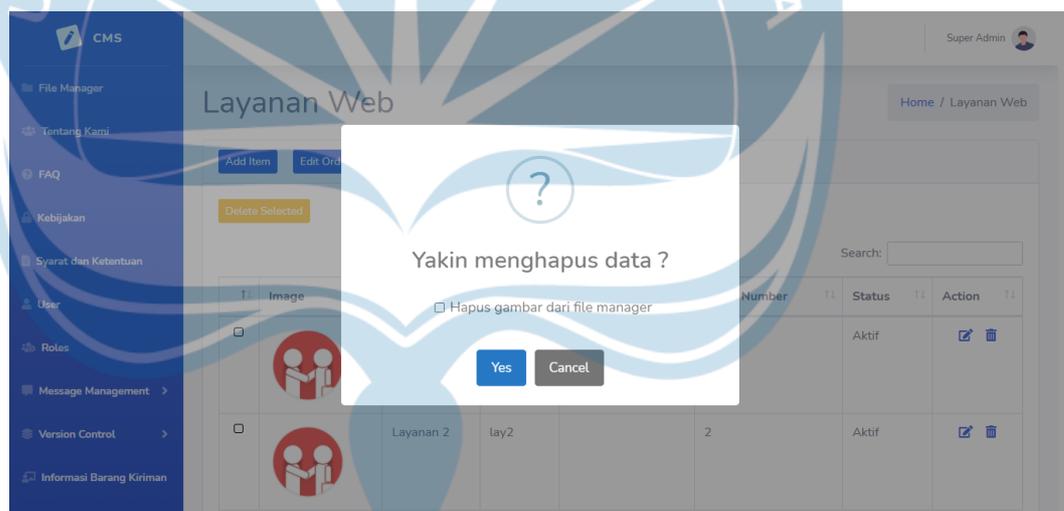
**Gambar 92.** Tampilan Antarmuka LayananUI Halaman *edit()* versi *web*

Halaman *edit()* yang ditampilkan pada Gambar 91 dan Gambar 92 berfungsi untuk menampilkan dan mengubah data layanan terpilih. Bagian Desc menggunakan *plugin text editor* yaitu Summernote. Jika status Active yang sebelumnya berisi “No” diubah menjadi “Yes” maka *order number* akan berubah menjadi *order number* terakhir +1, sedangkan jika status Active yang sebelumnya berisi “Yes” diubah menjadi “No” maka *order number* layanan diubah menjadi 0.

#### 5.3.17.4. Antarmuka *delete()*



**Gambar 93.** Tampilan Antarmuka LayananUI Halaman *delete()* versi *app*

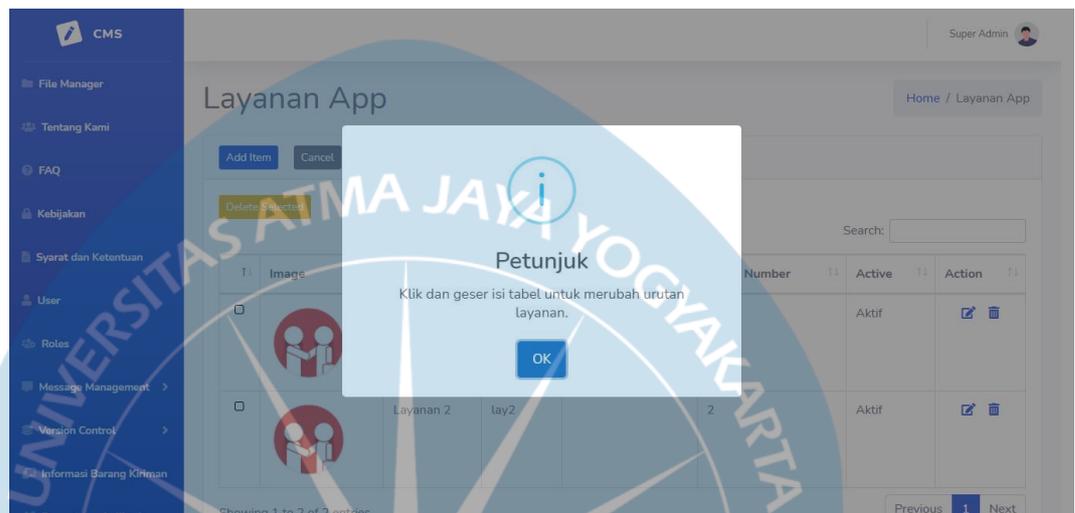


**Gambar 94.** Tampilan Antarmuka LayananUI Halaman *delete()* versi *web*

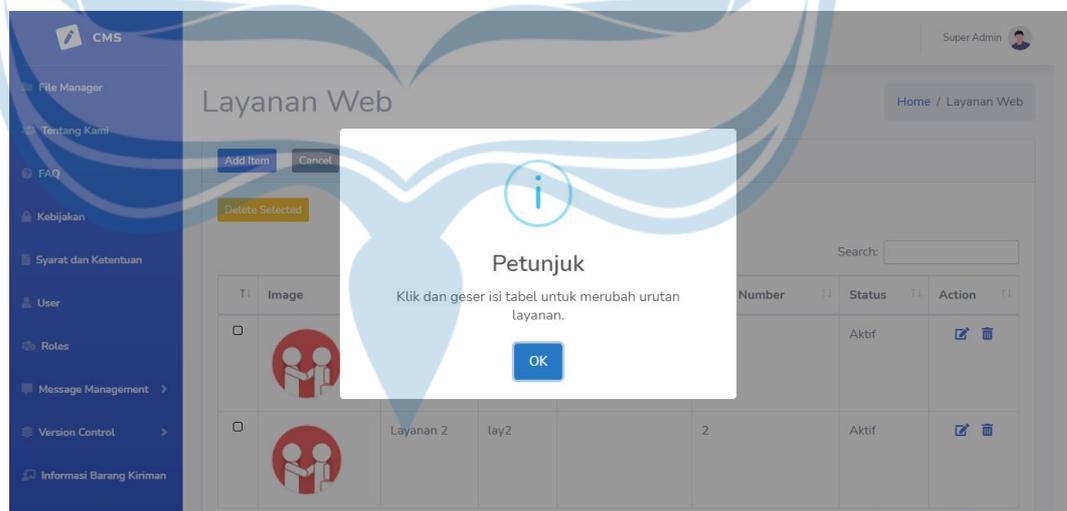
Halaman *delete()* yang ditampilkan pada Gambar 93 dan Gambar 94 berfungsi sebagai halaman konfirmasi penghapusan data layanan. Tombol “*Yes*” berfungsi untuk mengkonfirmasi penghapusan data *version* terpilih. Tombol “*Cancel*” berfungsi untuk menutup dan membatalkan penghapusan layanan terpilih. Layanan yang masih berstatus aktif tidak bisa dihapus hingga status layanan diubah menjadi tidak aktif. Jika

*checklist* hapus gambar dari *file manager* dipilih, maka *file* gambar dari layanan akan dihapus dari sistem penyimpanan.

### 5.3.17.5. Antarmuka editorder()



**Gambar 95.** Tampilan Antarmuka LayananUI Halaman editorder() versi *app*



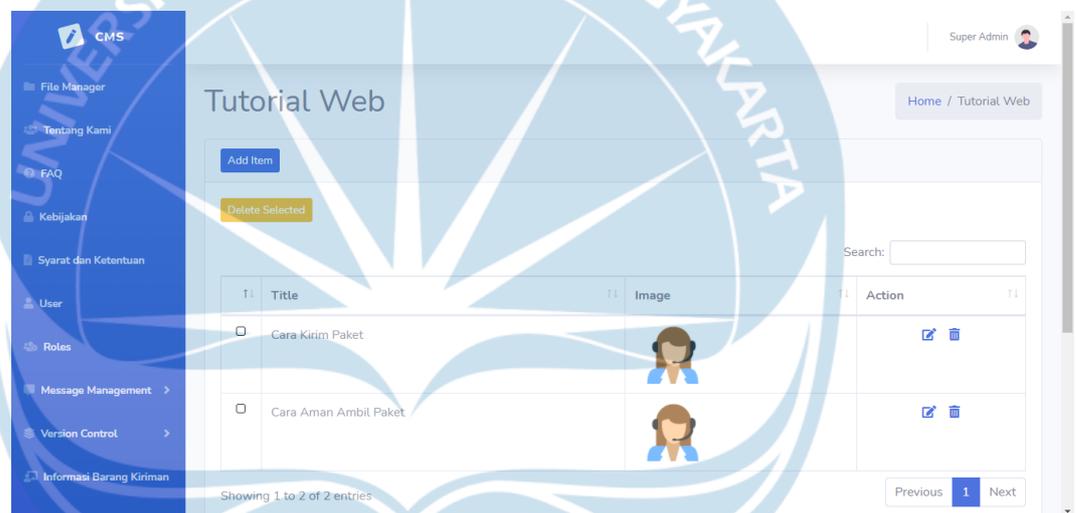
**Gambar 96.** Tampilan Antarmuka LayananUI Halaman editorder() versi *web*

Halaman *editorder()* yang ditampilkan pada Gambar 95 dan Gambar 96 merupakan kotak *dialog* untuk menampilkan petunjuk pengeditan urutan layanan yang ditampilkan. Setelah *user* mengklik tombol OK, halaman *editorder()* ditutup dan kembali ke halaman *index()*, namun tombol *edit*

*order* berubah menjadi tombol “*Cancel*” dan tombol “*Save Order*”, daftar layanan yang terdapat pada tabel juga dapat diklik dan digeser untuk diubah urutannya. Tombol “*Save Order*” berfungsi untuk menyimpan urutan layanan yang sudah diatur. Tombol “*Cancel*” berfungsi untuk membatalkan fungsi *edit order*, dan kembali ke halaman *index()* seperti awal.

### 5.3.18. Antarmuka TutorialWebUI

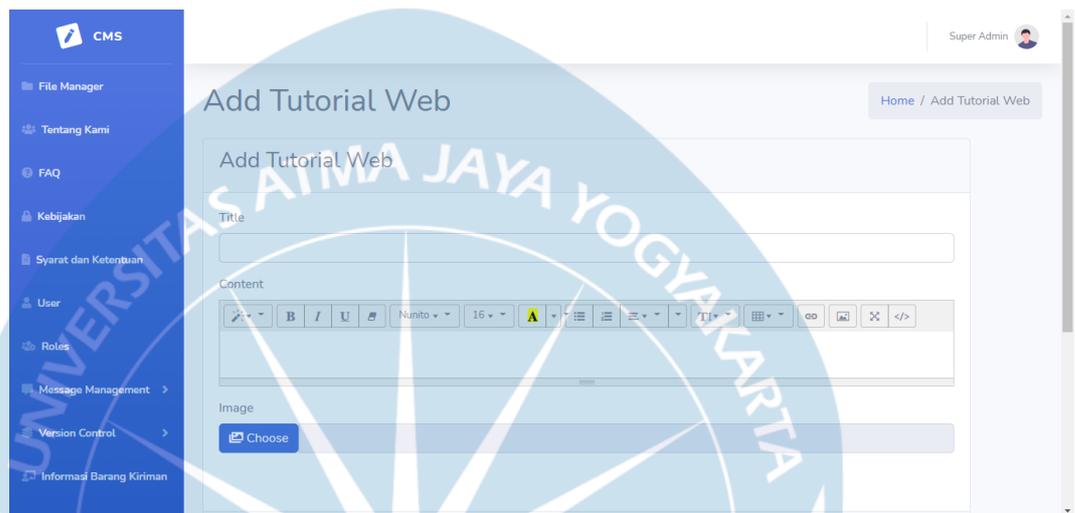
#### 5.3.18.1. Antarmuka *index()*



Gambar 97. Tampilan Antarmuka TutorialWebUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 97 berfungsi untuk menampilkan data *tutorial web* yang tersimpan pada *database*. Tombol “*Add Item*” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data *version* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *tutorial web* terpilih. Tombol “*Delete Selected*” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *tutorial web* terpilih.

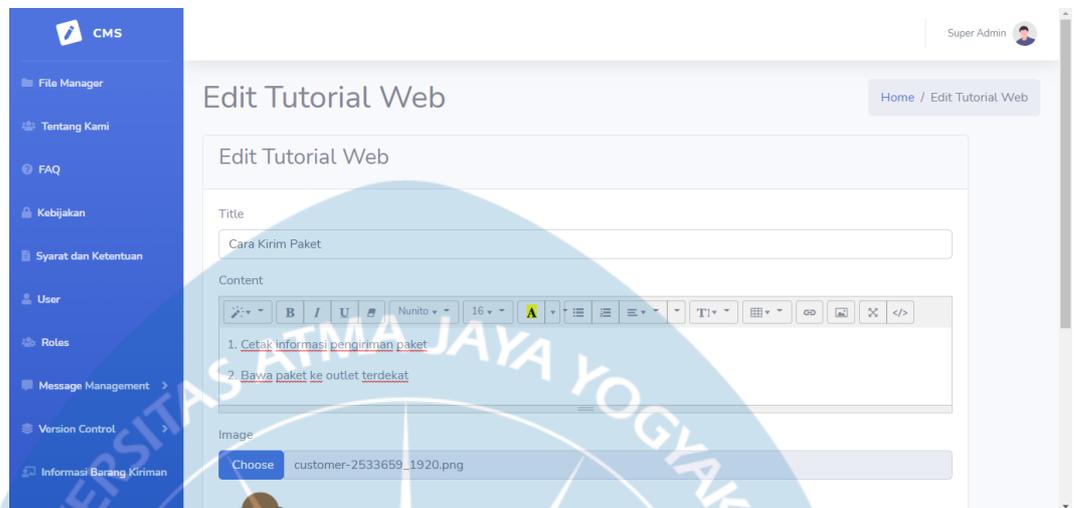
### 5.3.18.2. Antarmuka *add()*



**Gambar 98.** Tampilan Antarmuka TutorialWebUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 98 berfungsi untuk menambahkan data *tutorial web* baru. Bagian *Content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “*Save*” berfungsi untuk menambahkan *tutorial web* baru ke dalam *database*.

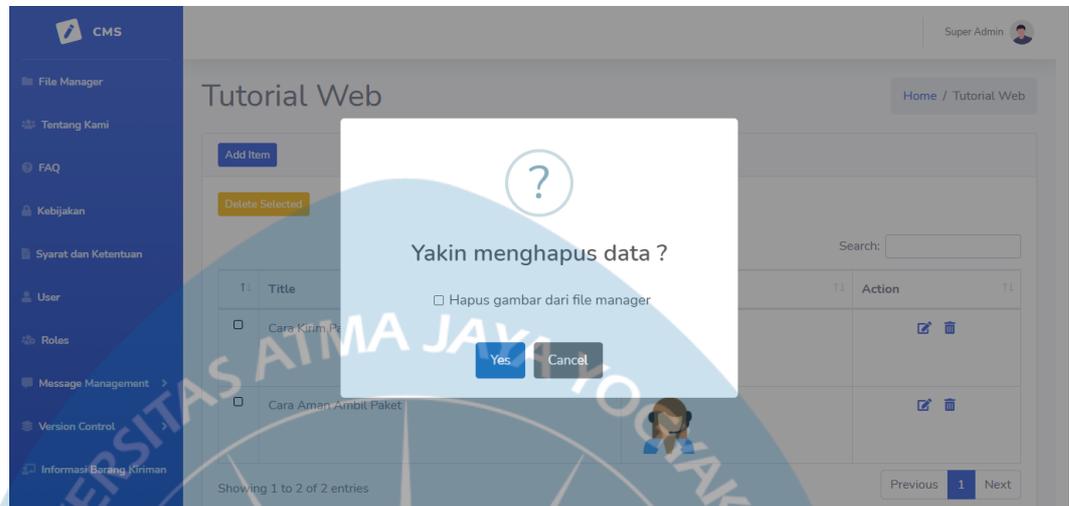
### 5.3.18.3. Antarmuka *edit()*



**Gambar 99.** Tampilan Antarmuka TutorialWebUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 99 berfungsi untuk menampilkan dan mengubah data *tutorial web* terpilih. Bagian *Content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “Save” berfungsi untuk menyimpan perubahan dari *tutorial web* terpilih.

#### 5.3.18.4. Antarmuka *delete()*

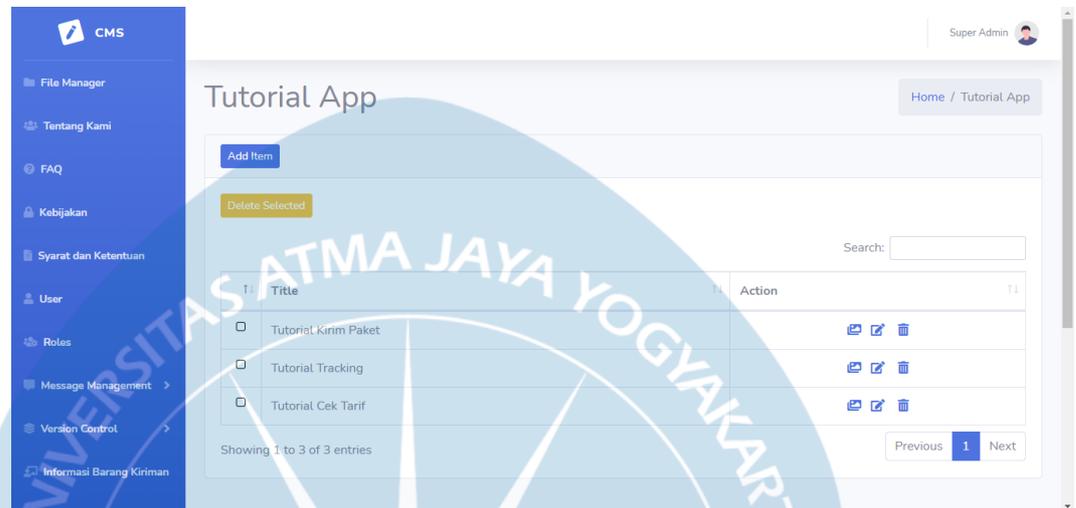


**Gambar 100.** Tampilan Antarmuka TutorialWebUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 100 berfungsi sebagai halaman konfirmasi penghapusan data *tutorial web*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data *tutorial web* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan *tutorial web* terpilih. Jika *checkboxlist* hapus gambar dari *file manager* dipilih, maka *file* gambar dari *tutorial web* akan dihapus dari sistem penyimpanan.

### 5.3.19. Antarmuka TutorialAppUI

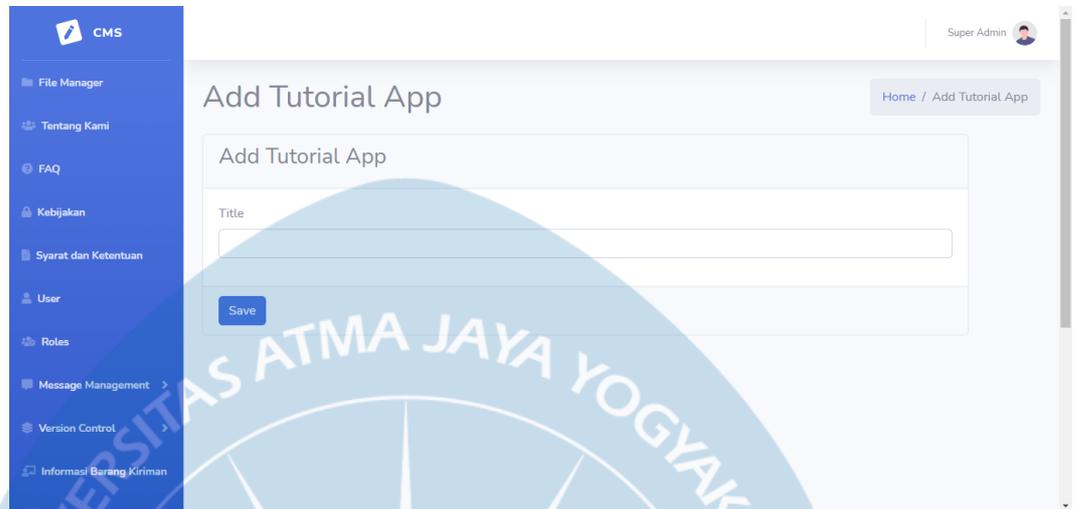
#### 5.3.19.1. Antarmuka *index()*



**Gambar 101.** Tampilan Antarmuka TutorialAppUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 101 berfungsi untuk menampilkan data *tutorial app* yang tersimpan pada *database*. Tombol “*Add Item*” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data *tutorial app* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *tutorial app* terpilih. Tombol “*Delete Selected*” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *tutorial app* terpilih.

### 5.3.19.2. Antarmuka *add()*

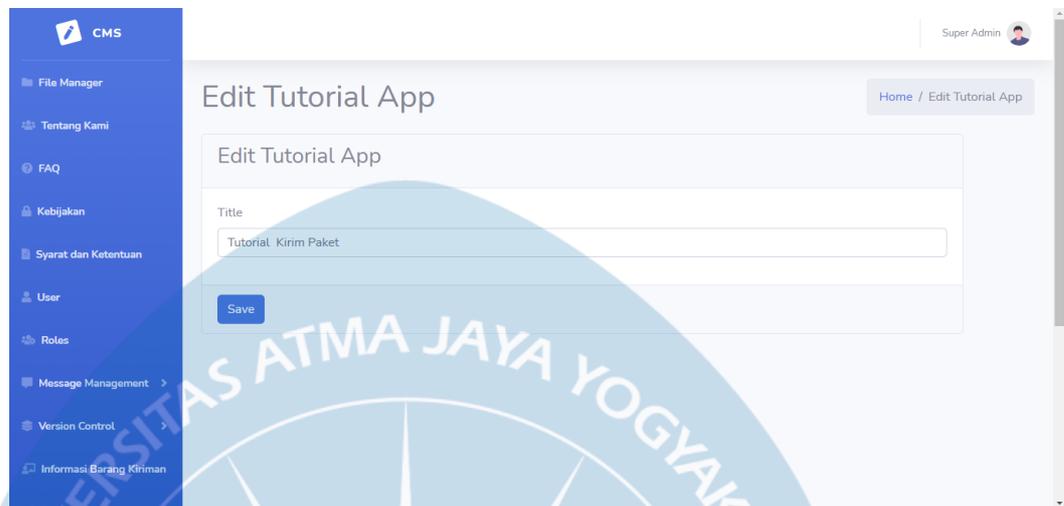


The screenshot shows a web interface for adding a new tutorial app. On the left is a blue sidebar menu with the 'CMS' logo and various navigation items like 'File Manager', 'Tentang Kami', 'FAQ', 'Kebijakan', 'Syarat dan Ketentuan', 'User', 'Roles', 'Message Management', 'Version Control', and 'Informasi Barang Kiriman'. The main content area is titled 'Add Tutorial App' and contains a form with a 'Title' input field and a 'Save' button. The breadcrumb trail at the top right reads 'Home / Add Tutorial App'. A large, semi-transparent watermark of the Universitas Atma Jaya Yogyakarta logo is overlaid on the page.

**Gambar 102.** Tampilan Antarmuka TutorialAppUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 102 berfungsi untuk menambahkan data *tutorial app* baru. Tombol “*Save*” digunakan untuk menambahkan *tutorial app* baru ke dalam *database*.

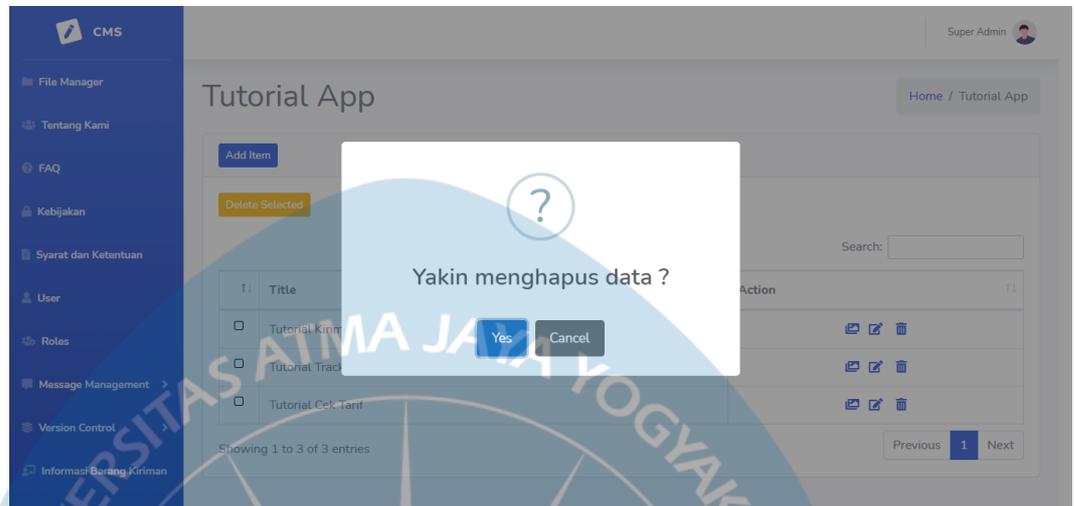
### 5.3.19.3. Antarmuka *edit()*



**Gambar 103.** Tampilan Antarmuka TutorialAppUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 103 berfungsi untuk menampilkan dan mengubah data *tutorial app* terpilih. Tombol “Save” berfungsi untuk menyimpan perubahan data dari *tutorial app* terpilih.

#### 5.3.19.4. Antarmuka *delete()*

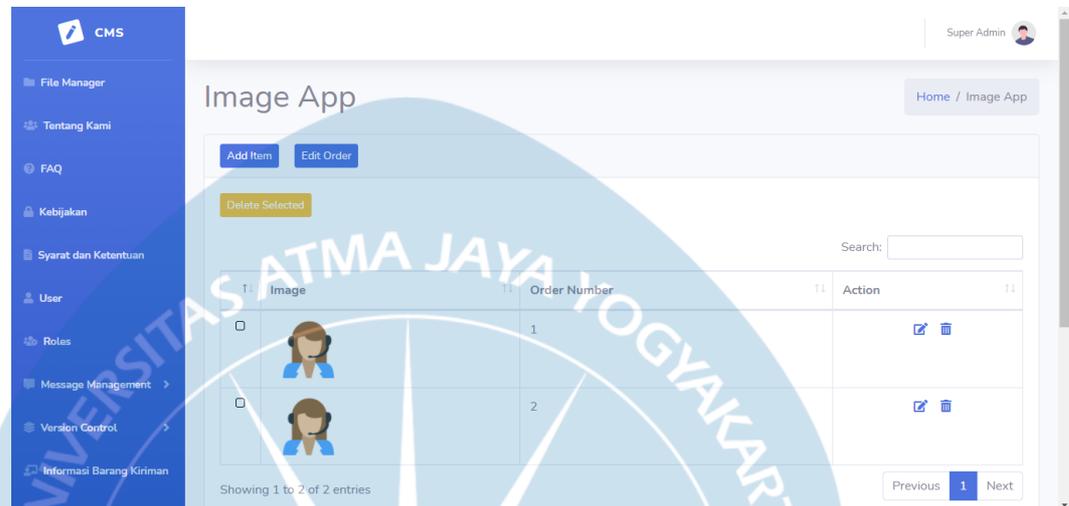


**Gambar 104.** Tampilan Antarmuka TutorialAppUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 104 berfungsi sebagai halaman konfirmasi penghapusan data *tutorial app*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data *tutorial app* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan *tutorial app* terpilih. *Tutorial app* yang masih mempunyai *image* di dalamnya tidak bisa dihapus hingga *tutorial app* tidak mempunyai *image* di dalamnya.

### 5.3.20. Antarmuka TutorialAppImageUI

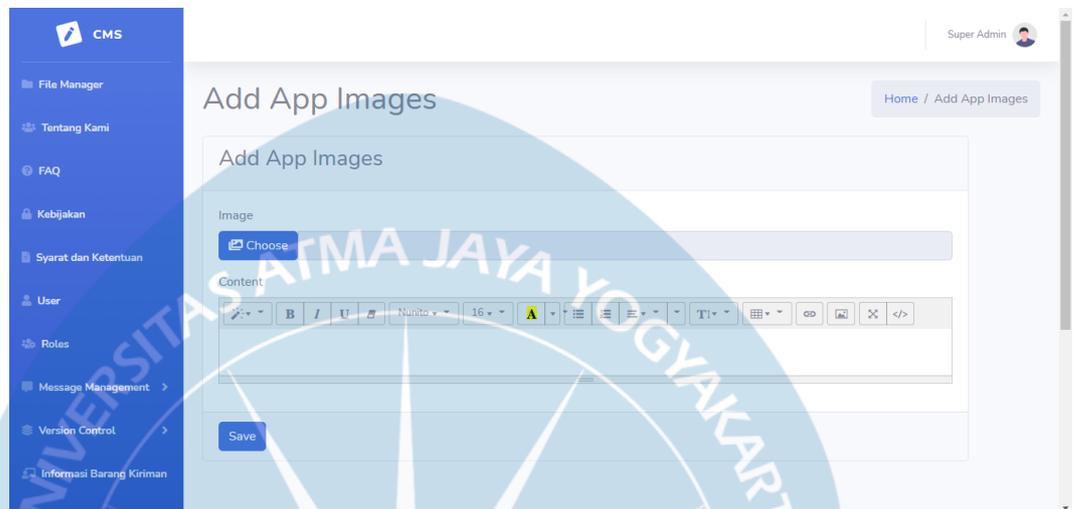
#### 5.3.20.1. Antarmuka *index()*



**Gambar 105.** Tampilan Antarmuka TutorialAppImageUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 105 berfungsi untuk menampilkan data gambar *tutorial app* yang tersimpan pada *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data gambar *tutorial app* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data gambar *tutorial app* terpilih. Tombol “Delete Selected” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus gambar *tutorial app* terpilih.

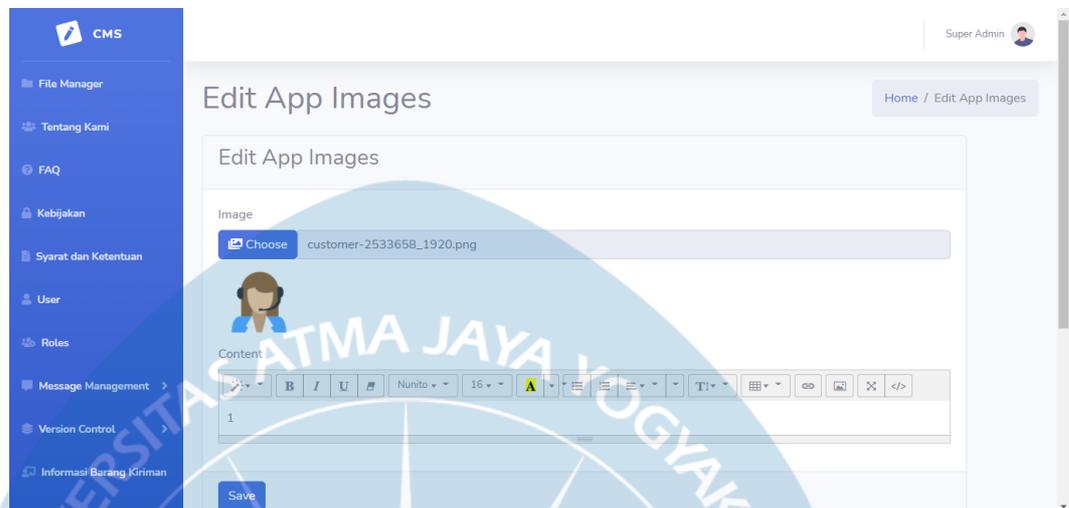
### 5.3.20.2. Antarmuka *add()*



**Gambar 106.** Tampilan Antarmuka TutorialAppImageUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 106 berfungsi untuk menambahkan data gambar *tutorial app* baru. Bagian *Content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “*Save*” digunakan untuk menambah gambar *tutorial app* baru ke dalam *database*.

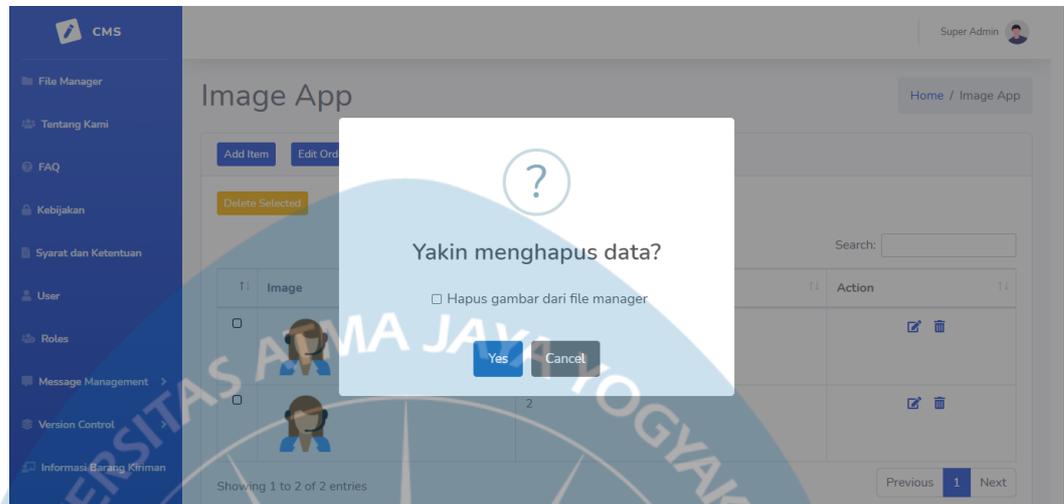
### 5.3.20.3. Antarmuka *edit()*



**Gambar 107.** Tampilan Antarmuka TutorialAppImageUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 107 berfungsi untuk menampilkan dan mengubah data gambar *tutorial app* terpilih. Bagian *Content* pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “*Save*” berfungsi untuk menyimpan perubahan data dari gambar *tutorial app* terpilih.

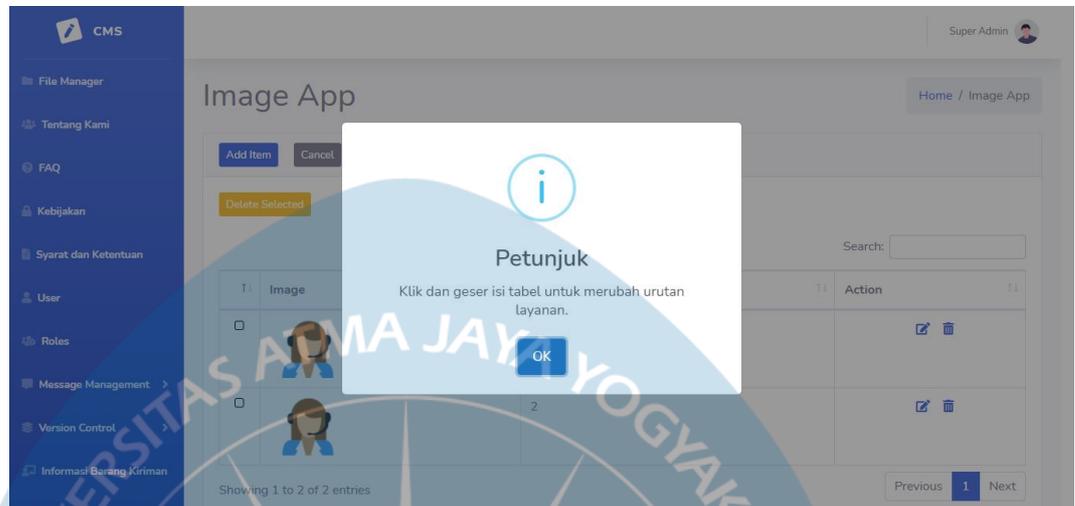
#### 5.3.20.4. Antarmuka *delete()*



**Gambar 108.** Tampilan Antarmuka TutorialAppImageUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 108 berfungsi sebagai halaman konfirmasi penghapusan data gambar *tutorial app*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data gambar *tutorial app* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan gambar *tutorial app* terpilih. Jika *checkboxlist* hapus gambar dari *file manager* dipilih, maka *file* gambar dari *tutorial app* terpilih akan dihapus dari sistem penyimpanan.

### 5.3.20.5. Antarmuka editorder()

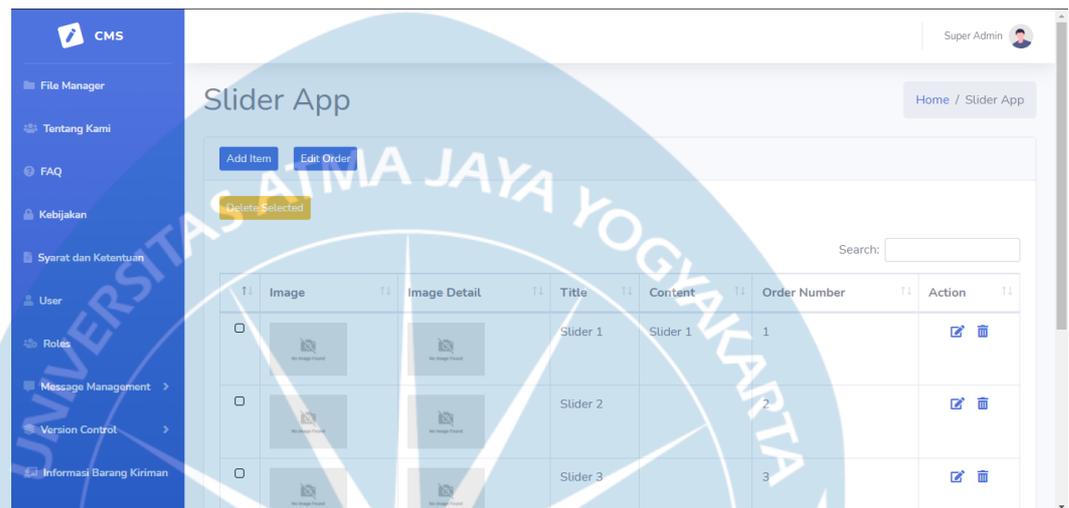


**Gambar 109.** Tampilan Antarmuka TutorialAppImageUI Halaman editorder()

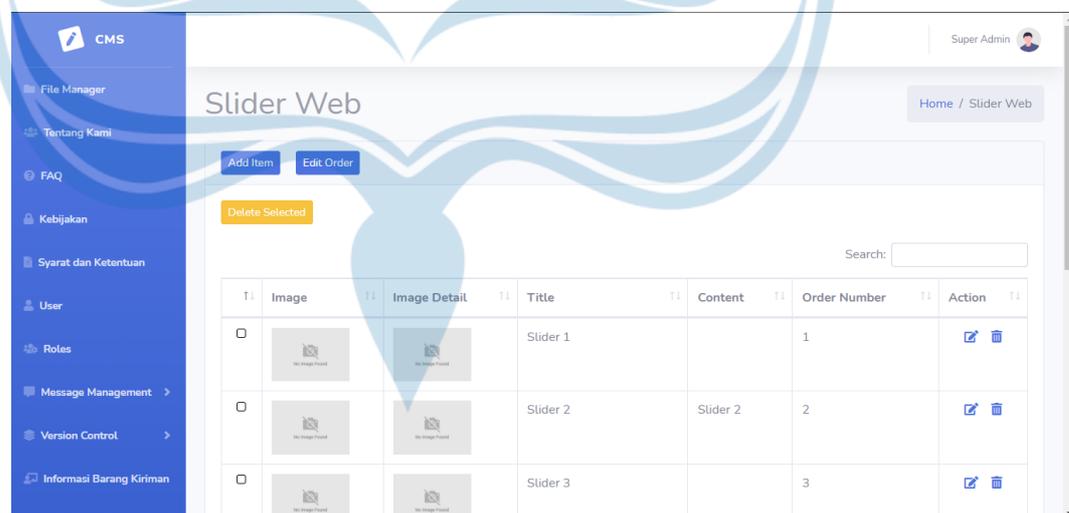
Halaman *editorder()* yang ditampilkan pada Gambar 109 merupakan kotak *dialog* untuk menampilkan petunjuk untuk mengubah urutan gambar *tutorial app*. Setelah tombol “*Edit Order*” di klik, maka tombol “*Edit Order*” akan berubah menjadi tombol “*Cancel*” dan “*Save Order*”, gambar-gambar *tutorial app* pada tabel menjadi dapat diklik dan digeser untuk diubah urutannya. Tombol “*Save Order*” digunakan untuk menyimpan urutan gambar *tutorial app* yang sudah diubah. Tombol “*Cancel*” digunakan untuk membatalkan *edit order* dan kembali ke halaman *index()* seperti semula.

### 5.3.21. Antarmuka SliderUI

#### 5.3.21.1. Antarmuka *index()*



**Gambar 110.** Tampilan Antarmuka SliderUI Halaman *index()* versi *app*



**Gambar 111.** Tampilan Antarmuka SliderUI Halaman *index()* versi *web*

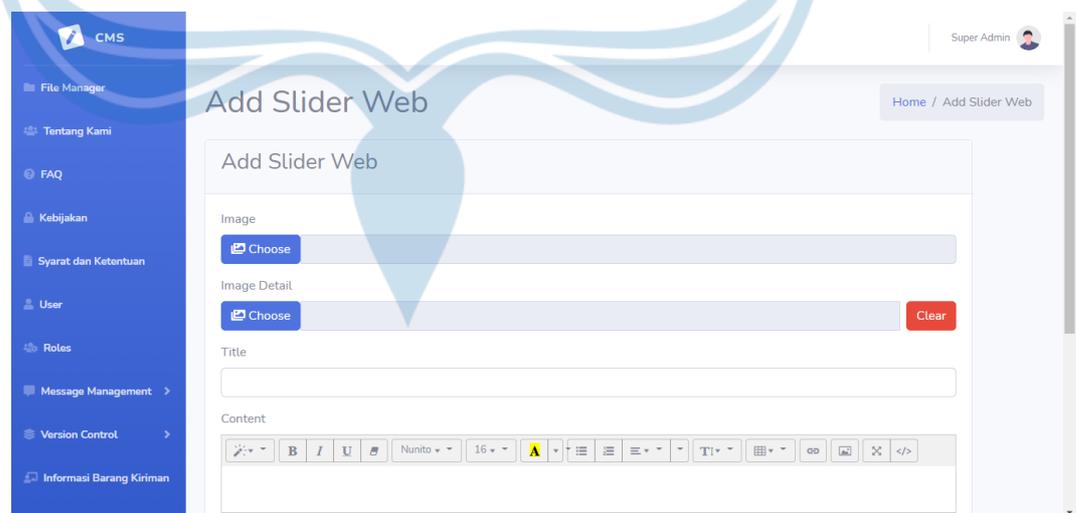
Halaman *index()* yang ditampilkan pada Gambar 110 dan Gambar 111 berfungsi untuk menampilkan data *slider* yang tersimpan pada *database*. Tombol “Add Item” berfungsi untuk masuk ke halaman *add()*

untuk menambahkan data *slider* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *slider* terpilih. Tombol “Delete Selected” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *slider* terpilih.

### 5.3.21.2. Antarmuka *add()*



**Gambar 112.** Tampilan Antarmuka SliderUI Halaman *add()* versi *app*

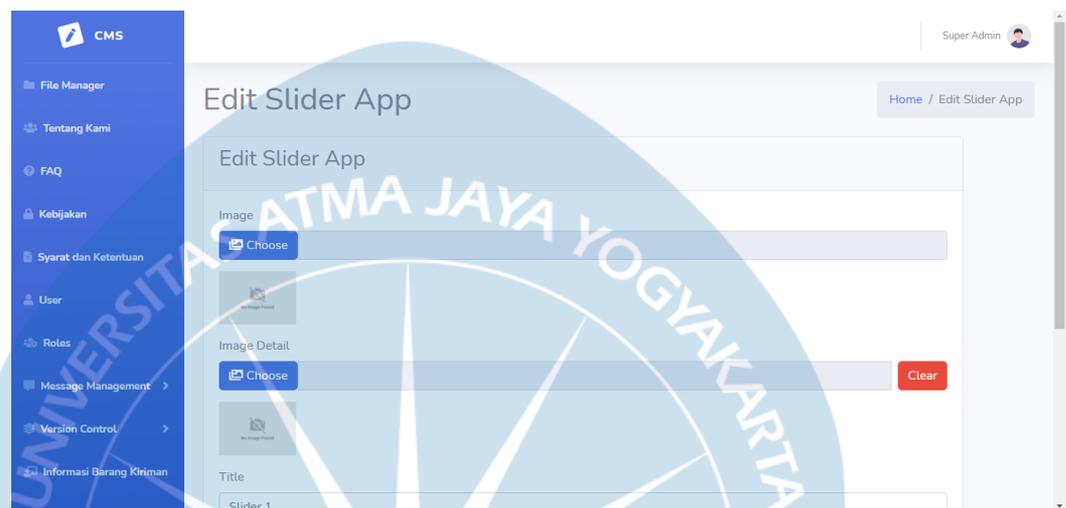


**Gambar 113.** Tampilan Antarmuka SliderUI Halaman *add()* versi *web*

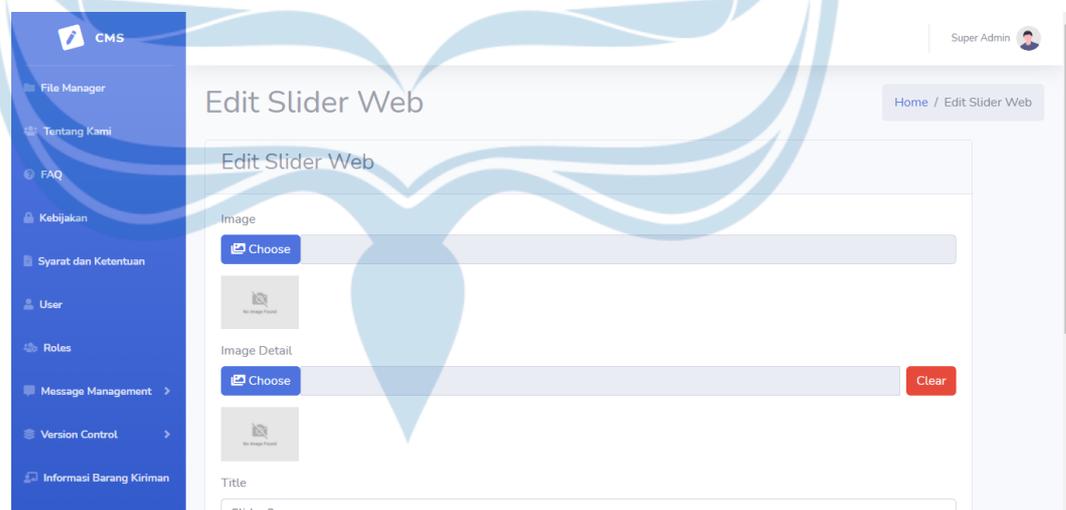
Halaman *add()* yang ditampilkan pada Gambar 112 dan Gambar 113 berfungsi untuk menambahkan data *slider* baru. Bagian *Content* pada

halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “Save” digunakan untuk menambah *slider* baru ke dalam *database*.

### 5.3.21.3. Antarmuka *edit()*



**Gambar 114.** Tampilan Antarmuka SliderUI Halaman *edit()* versi *app*

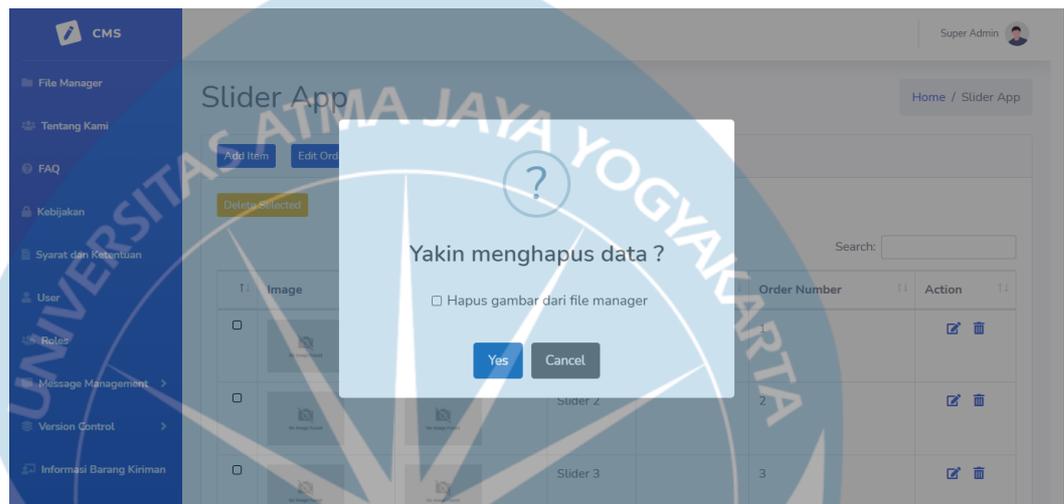


**Gambar 115.** Tampilan Antarmuka SliderUI Halaman *edit()* versi *web*

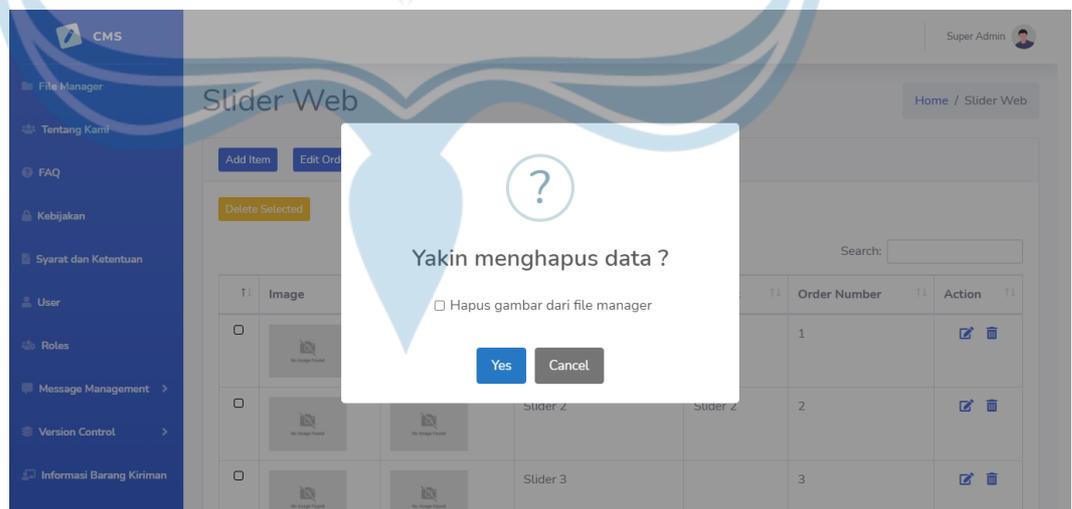
Halaman *edit()* yang ditampilkan pada Gambar 114 dan Gambar 115 berfungsi untuk menampilkan dan mengubah data *slider* terpilih. Bagian *Content* pada halaman ini menggunakan *plugin text editor* yaitu

Summernote. Tombol “Save” berfungsi untuk menyimpan perubahan data dari gambar *tutorial app* terpilih.

#### 5.3.21.4. Antarmuka *delete()*



Gambar 116. Tampilan Antarmuka SliderUI Halaman *delete()* versi *app*



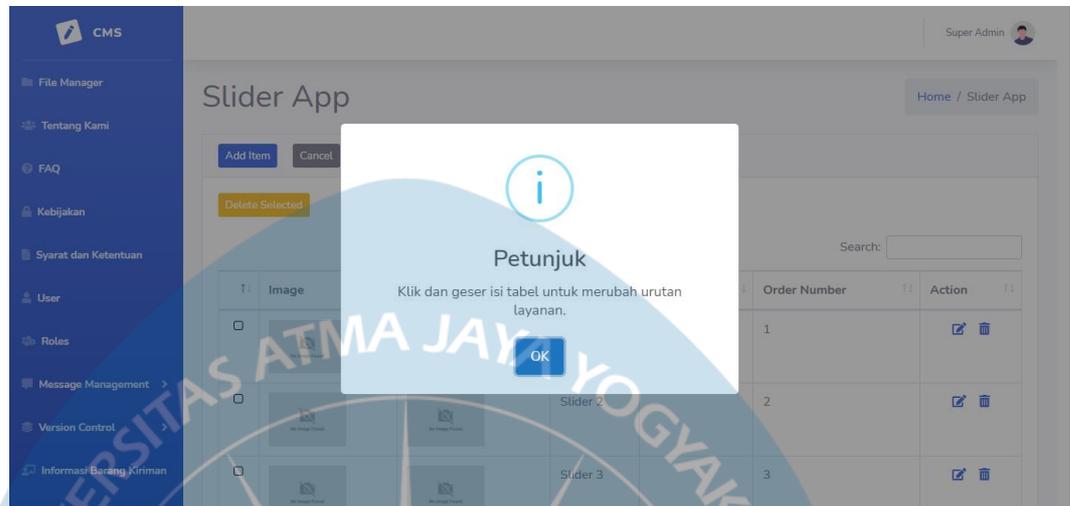
Gambar 117. Tampilan Antarmuka SliderUI Halaman *delete()* versi *web*

Halaman *delete()* yang ditampilkan pada Gambar 116 dan Gambar 117 berfungsi sebagai halaman konfirmasi penghapusan data *slider*.

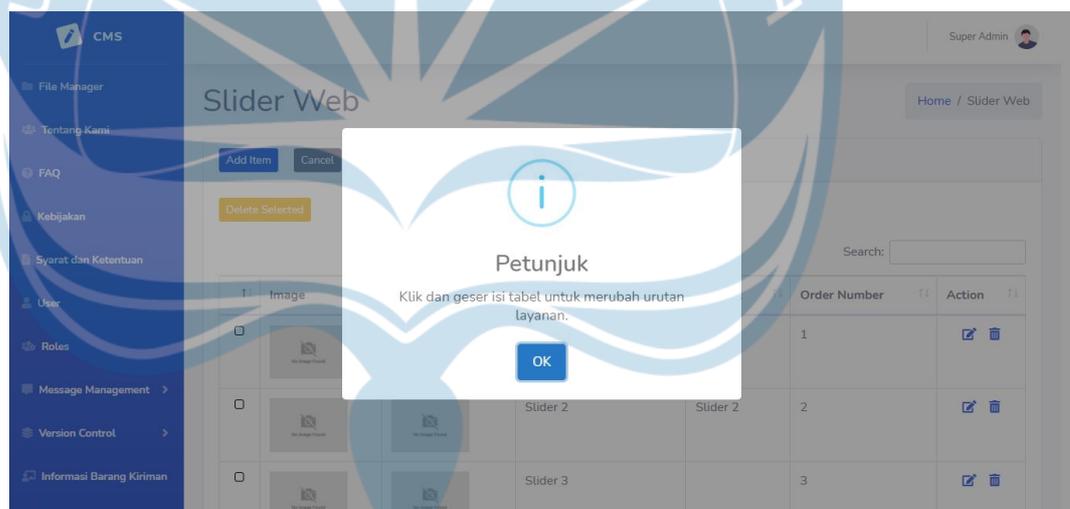
Tombol “*Yes*” berfungsi untuk mengkonfirmasi penghapusan data *slider* terpilih. Tombol “*Cancel*” berfungsi untuk menutup dan membatalkan penghapusan *slider* terpilih. Jika *checklist* hapus gambar dari *file manager* dipilih, maka *file* gambar dari *slider* terpilih akan dihapus dari sistem penyimpanan.



### 5.3.21.5. Antarmuka editorder()



**Gambar 118.** Tampilan Antarmuka SliderUI Halaman editorder() versi *app*



**Gambar 119.** Tampilan Antarmuka SliderUI Halaman editorder() versi *web*

Halaman *editorder()* yang ditampilkan pada Gambar 118 dan Gambar 119 merupakan kotak *dialog* untuk menampilkan petunjuk untuk mengubah urutan *slider*. Setelah tombol “*Edit Order*” di klik, maka tombol “*Edit Order*” akan berubah menjadi tombol “*Cancel*” dan “*Save Order*”, *slider* pada tabel menjadi dapat diklik dan digeser untuk diubah urutannya. Tombol “*Save Order*” digunakan untuk menyimpan urutan

*slider* yang sudah diubah. Tombol “*Cancel*” digunakan untuk membatalkan *edit order* dan kembali ke halaman *index()* seperti semula.

### 5.3.22. Antarmuka NotificationUI

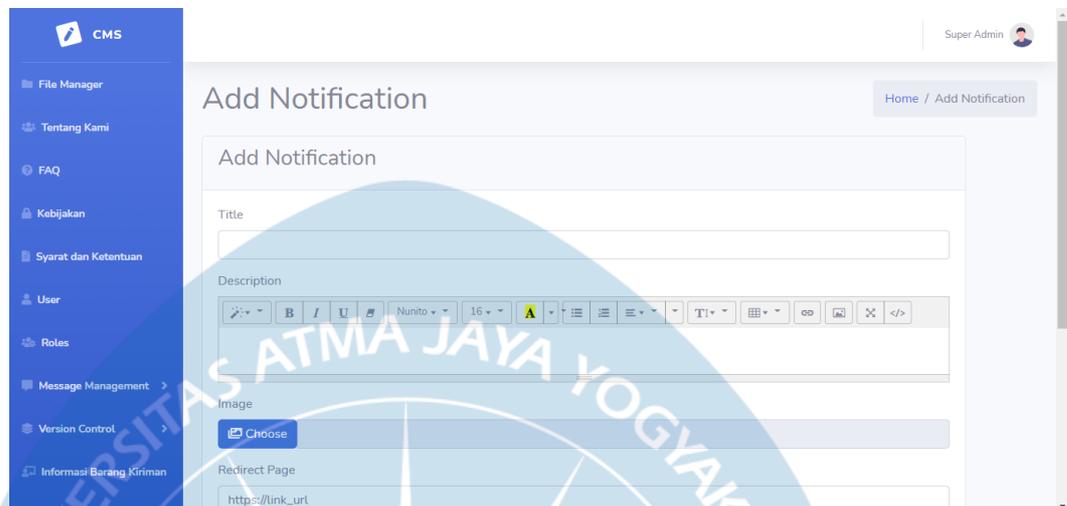
#### 5.3.22.1. Antarmuka *index()*



Gambar 120. Tampilan Antarmuka NotificationUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 120 berfungsi untuk menampilkan data *notification* yang tersimpan pada *database*. Tombol “*Add Item*” berfungsi untuk masuk ke halaman *add()* untuk menambahkan data *notification* baru. Tombol dengan logo pensil berfungsi untuk masuk ke halaman *edit()* dan meng-*edit* data *notification* terpilih. Tombol “*Delete Selected*” dan tombol berlogo tempat sampah berfungsi untuk memilih dan menghapus *notification* terpilih.

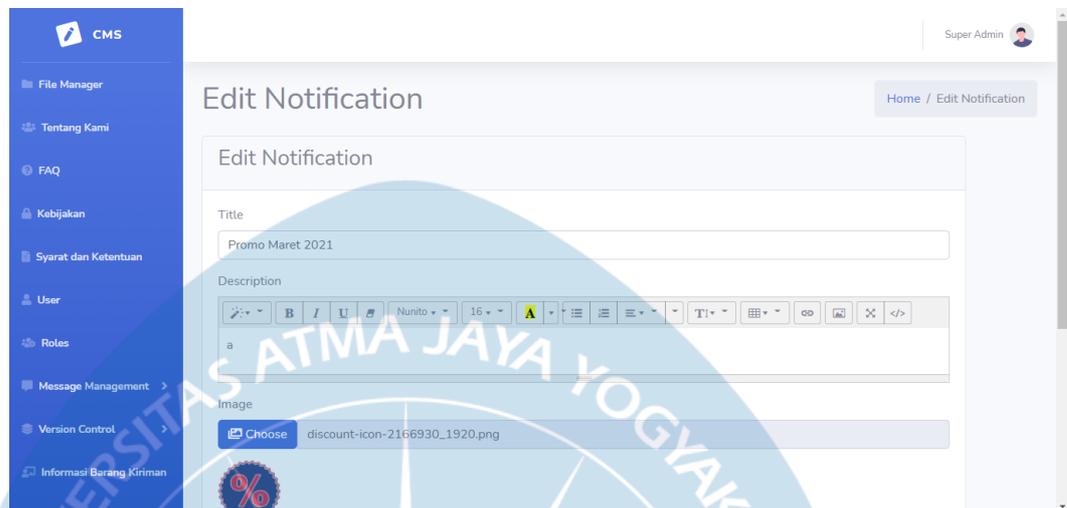
### 5.3.22.2. Antarmuka *add()*

The image shows a web interface for adding a notification. On the left is a blue sidebar menu with 'CMS' at the top and various navigation items like 'File Manager', 'Tentang Kami', 'FAQ', 'Kebijakan', 'Syarat dan Ketentuan', 'User', 'Roles', 'Message Management', 'Version Control', and 'Informasi Berang Kiriman'. The main content area is titled 'Add Notification' and contains a form with the following fields: 'Title' (a text input), 'Description' (a rich text editor with a toolbar including bold, italic, underline, link, unlink, list, and code), 'Image' (a 'Choose' button), and 'Redirect Page' (a text input with the placeholder 'https://link\_url'). The top right corner shows the user 'Super Admin' and a breadcrumb 'Home / Add Notification'.

**Gambar 121.** Tampilan Antarmuka NotificationUI Halaman *add()*

Halaman *add()* yang ditampilkan pada Gambar 121 berfungsi untuk menambahkan data *notification* baru. Bagian Description pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “Send *notification*” yang terdapat pada bagian bawah halaman digunakan untuk menambah *notification* baru ke dalam *database*, dan mengirimkan push *notification* ke device *user* yang mempunyai aplikasi perusahaan.

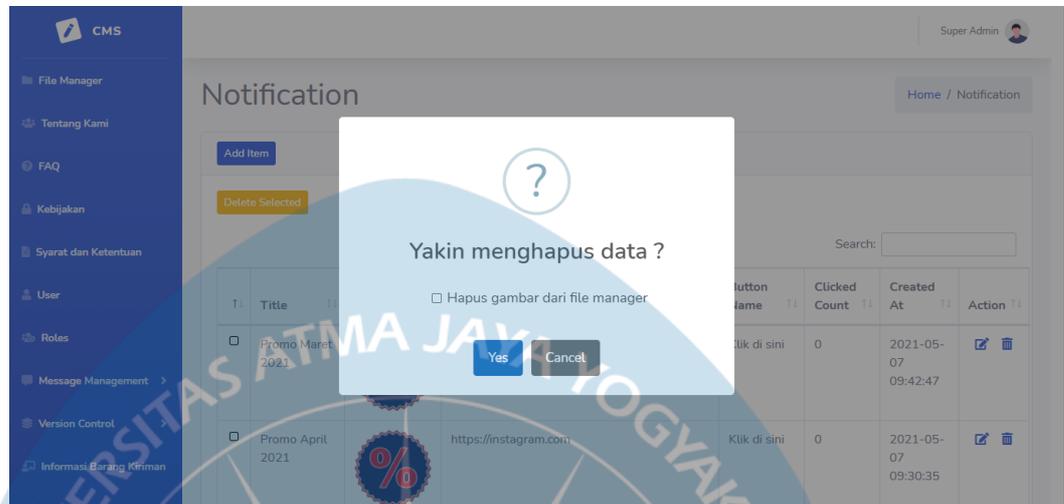
### 5.3.22.3. Antarmuka *edit()*



**Gambar 122.** Tampilan Antarmuka NotificationUI Halaman *edit()*

Halaman *edit()* yang ditampilkan pada Gambar 122 berfungsi untuk menampilkan dan mengubah data *notification* terpilih. Bagian Description pada halaman ini menggunakan *plugin text editor* yaitu Summernote. Tombol “Save” berfungsi untuk menyimpan perubahan data dari *notification* terpilih.

#### 5.3.22.4. Antarmuka *delete()*

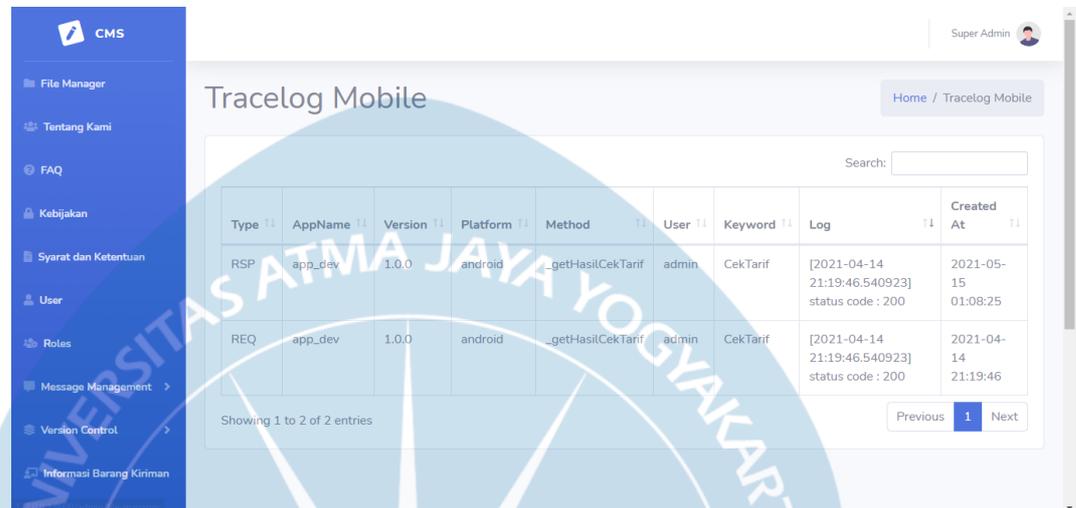


**Gambar 123.** Tampilan Antarmuka NotificationUI Halaman *delete()*

Halaman *delete()* yang ditampilkan pada Gambar 123 berfungsi sebagai halaman konfirmasi penghapusan data *notification*. Tombol “Yes” berfungsi untuk mengkonfirmasi penghapusan data *notification* terpilih. Tombol “Cancel” berfungsi untuk menutup dan membatalkan penghapusan *notification* terpilih. Jika *checkbox* hapus gambar dari *file manager* dipilih, maka *file* gambar dari *notification* terpilih akan dihapus dari sistem penyimpanan.

### 5.3.23. Antarmuka TracelogMobileUI

#### 5.3.23.1. Antarmuka *index()*



The screenshot displays the 'Tracelog Mobile' interface. On the left is a blue sidebar menu with various system management options. The main content area features a search bar and a table with two columns of data. The table has headers for Type, AppName, Version, Platform, Method, User, Keyword, Log, and Created At. Two entries are visible, both for 'app\_dev' on 'android' with method '\_getHasilCekTarif' and user 'admin'. The log entries show timestamps and status codes.

Type	AppName	Version	Platform	Method	User	Keyword	Log	Created At
RSP	app_dev	1.0.0	android	_getHasilCekTarif	admin	CekTarif	[2021-04-14 21:19:46.540923] status code : 200	2021-05-15 01:08:25
REQ	app_dev	1.0.0	android	_getHasilCekTarif	admin	CekTarif	[2021-04-14 21:19:46.540923] status code : 200	2021-04-14 21:19:46

**Gambar 124.** Tampilan Antarmuka TracelogMobileUI Halaman *index()*

Halaman *index()* yang ditampilkan pada Gambar 124 digunakan untuk menampilkan data *tracelog mobile* yang terdapat pada *database*. Jumlah data *tracelog* yang ditampilkan maksimal 1000 baris untuk mencegah halaman memuat terlalu banyak data dan membuat sistem berjalan dengan lambat.

## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1. Kesimpulan

Penelitian ini dilakukan untuk membangun sistem pengelolaan web & app yang dapat dikelola secara mandiri oleh perusahaan. Pengembangan sistem dilakukan dengan menggunakan metode *Agile-Waterfall Hybrid*. Penelitian ini berfokus pada proses analisis kebutuhan dan perancangan dari SDLC (*System Development Life Cycle*).

Analisis kebutuhan pada penelitian ini dibagi menjadi 6 bagian, yaitu:

1. *System Requirement*
2. Arsitektur Sistem
3. Fungsi Produk
4. Kebutuhan Fungsional (*Use Case Diagram*)
5. *Sequence Diagram*
6. Kebutuhan Non-Fungsional

Dari analisis kebutuhan yang dilakukan, didapatkan bahwa sistem membutuhkan total 22 fungsi yang diperlukan untuk mengelola konten *website* dan aplikasi *mobile* yaitu *log in*, *edit profil*, *file manager*, tentang kami, *faq*, kebijakan, syarat ketentuan, *role*, *user*, *message*, tipe *message*, halaman, pengelolaan versi, informasi barang kiriman, pengaturan aplikasi, pengelolaan layanan, *tutorial web*, *tutorial app*, *tutorial app image*, *slider*, notifikasi, *tracelog mobile* dan *log out*. Fungsi-fungsi tersebut didapatkan dengan melakukan observasi terhadap sistem serta permintaan kebutuhan dari perusahaan.

Hasil perancangan yang dilakukan pada penelitian ini dibagi menjadi 3, yaitu:

1. ERD (*Entity Relationship Diagram*)
2. *Class Diagram*
3. Perancangan Antarmuka

Dari hasil perancangan, didapatkan bahwa sistem membutuhkan 21 tabel pada database, dimana terdapat 15 tabel master, serta 6 tabel transaksional.

## 6.2. Saran

Berdasarkan hasil yang didapatkan dari hasil penelitian ini, diharapkan untuk penelitian di masa mendatang yang melibatkan topik yang serupa dapat memberikan peningkatan, yaitu:

1. Penelitian dapat dilanjutkan menuju tahap SDLC (*System Development Life Cycle*) selanjutnya, yaitu tahapan testing, deployment, serta maintenance sehingga dapat tercipta dokumentasi perangkat lunak yang lengkap yang dapat digunakan untuk pengembangan lebih lanjut di masa mendatang.
2. Penelitian dapat dilanjutkan untuk mencakup manajemen konfigurasi pada *web* dan aplikasi yang berkaitan dengan lingkungan atau *environment* pengembangan, sehingga dapat mempermudah proses pengembangan fitur-fitur baru untuk kedepannya.

**DAFTAR PUSTAKA**

- [1] Asosiasi Penyelenggara Jasa Internet Indonesia, "Buletin APJII Edisi 74," APJII, Jakarta, November 2020.
- [2] B. Pink and N. Laoli, "BI catat jumlah transaksi e-commerce selama wabah corona tumbuh signifikan," KONTAN, Jakarta, 2020.
- [3] D. J. Bayu, "Survei MarkPlus: Frekuensi Penggunaan Jasa Kurir Meningkatkan saat Pandemi," katadata.co.id, Jakarta, 2020.
- [4] C. M. Annur, "Prospeknya Cerah, Bisnis Logistik Diprediksi Tumbuh Lebih 30% di 2020," katadata.co.id, Jakarta, 2019.
- [5] M. A. Q. Destiningrum, "Sistem Informasi Penjadwalan Dokter Berbasis Web Dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yukum Medical Centre)," *Jurnal Teknoinfo*, pp. 30-37, 2017.
- [6] M. Susilo, "Rancang Bangun Website Toko Online Menggunakan Metode Waterfall," *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*, pp. 98-105, 2018.
- [7] I. Hariman and C. Meilisa, "SISTEM INFORMASI MANAJEMEN TRAINING MENGGUNAKAN METODE AGILE SOFTWARE DEVELOPMENT THE PAPANDAYAN HOTEL BANDUNG," *ENSAINS: Vol. 3 Nomor 1*, pp. 60-67, Januari 2020.
- [8] I. Mahendra and D. T. E. Yanto, "Agile Development Methods dalam Pengembangan Sistem Informasi Pengajuan Kredit Berbasis Web (Studi Kasus: Bank BRI Unit Kolonel Sugiono)," *Jurnal Teknologi dan Open*

*Source Vol.1 No. 2, pp. 13-24, Desember 2018.*

- [9] K. Imtihan, R. Hadawiyah and H. Asyari, "Sistem Informasi Penggajian Guru Honorer Menggunakan Konsep Agile Software Development dengan Metodologi Extreme Programming (XP) pada SMK Bangun Bangsa," *Indonesian Journal on Networking and Security - Volume 7 No 2*, pp. 32-37, 2017.
- [10] H. Sihotang, "Sistem Informasi Pengagendaan Surat Berbasis Web Pada Pengadilan Tinggi Medan," *Journal of Informatic Pelita Nusantara*, p. 3(1), 2018.
- [11] Yuhefizar, *Cara Mudah Dan Murah Membangun Dan Mengelola Website*, Yogyakarta: Graha Ilmu, 2013.
- [12] R. Hidayat, *Cara Praktis Membangun Website Gratis*, Jakarta: PT. Alex Media Komputindo, 2010.
- [13] P. Sulistyorini, "Pemodelan Visual dengan Menggunakan UML dan Rational Rose," *Jurnal Teknologi Informasi DINAMIK Volume XIV, No. 1*, pp. 23-29, 2009.
- [14] G. Booch, R. A. Maksimchuk, M. W. Engle, J. Conallen, B. J. Young and K. A. Houston, *Object-oriented Analysis and Design with Applications*, Berkeley: Addison-Wesley, 2007.
- [15] A. Kumar, *Object Oriented Programming: Principal's and Theory*, 2014.
- [16] C. Pamungkas, *Pengantar dan Implementasi Basis Data*, Deepublish, 2017.
- [17] B. Sidik, *Menggunakan Framework CodeIgniter 2.x untuk Memudahkan Pengembangan Pemrograman Aplikasi WEB dengan PHP 5*, Bandung: Informatika Bandung, 2012.

- [18] A. Martias, "Analisa Kecukupan Penerapan Pengawasan Internal Dengan Metode," *Moneter*, vol. 5(2), pp. 23-32, 2018.
- [19] M. P. T. Mendoza, "Payroll System Design with SDLC (System Development Life Cycle) Approach," *Jurnal Mantik*, pp. 27-32, 2020.
- [20] Lucid Software Inc., "Agile-Waterfall Hybrid: Is It Right for Your Team?," [Online].
- [21] A. M. F. N. Sapto, "Rancangan Bangunan Sistem Informasi Disposisi Surat Berbasis Web (Studi Kasus Kementerian Pekerjaan Umum)," *Program Studi Teknik Informatika STMIK Provinsi, Semarang*, pp. 25-27, 2014.
- [23] B. P. Widodo and H. D. Purnomo, "Perancangan Aplikasi Pencarian Layanan Kesehatan Berbasis HTML5 Geolocation," *Jurnal Sistem Komputer*, 2016.
- [24] H. P. Permatasari and K. Shabrina, "PENERAPAN CONTENT MANAGEMENT SYSTEM (CMS) PADA PEMBANGUNAN SITUS WEB SMA PERINTIS DEPOK," *Proceeding, Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT 2008)*, pp. 653-660, 2008.
- [25] A. Achmad, "STUDI CONTENT MANAGEMENT SYSTEM (CMS) DAN PEMBUATAN WEB EDUKASI DI JURUSAN PENDIDIKAN TEKNIK ELEKTRO FT UNM," *Jurnal Ilmiah d'Computare Volume 6*, pp. 52-57, 2016.
- [26] T. Santia, "Transaksi E-Commerce Indonesia Diprediksi Tembus Rp 1.000 Triliun di 2025," *Liputan 6*, Jakarta, 2019.
- [27] A. S. Rosa and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Bandung: Informatika, 2013.

- [28] R. Pressman, Software Engineering: A Practitioner's Approach, New York: McGraw - Hill, 2010.



## Lampiran

### 1. Transkrip Wawancara

#### Meeting pertama terkait introduksi mengenai sistem

Narasumber: Manajer IT PT. XYZ

Tanggal: 14 Januari 2021

Media: Elektronik (Google Meet)

#### Keterangan

A: Peneliti

M: Manajer IT

M : Pagi Vincent, jadi gua jelasin dulu ya soal *web & app* kita, jadi kita punya *web* sama *app* itu bisa dipake *user* biasanya buat *tracking* paket, bikin nomor resi, ngecek tarif, dll.

M : (Mendemokan dan menjelaskan penggunaan *web*).

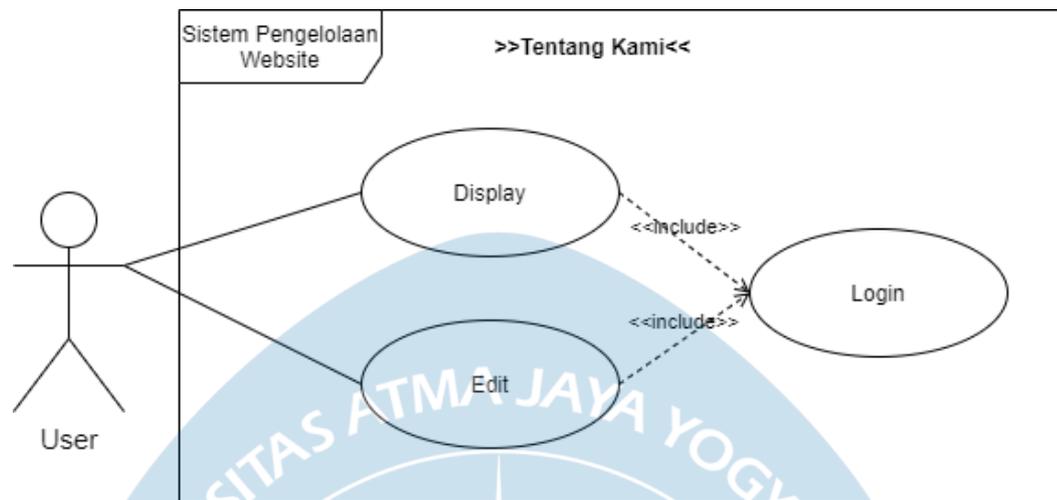
M : Nah, jadi ini kita sistemnya masih di-*develop* sama perusahaan luar, jadi biasa tuh kalo dari tim marketing pengen nambah fitur, itu mesti bayar lagi jadi lumayan berat tiap pengen nambah sesuatu. Selain itu juga, dari marketing udah ada beberapa fitur *pending* yang pengen ditambahin.

A : Oh, gitu Pak, terus kok dari awal nggak di *develop* langsung dari perusahaan Pak?

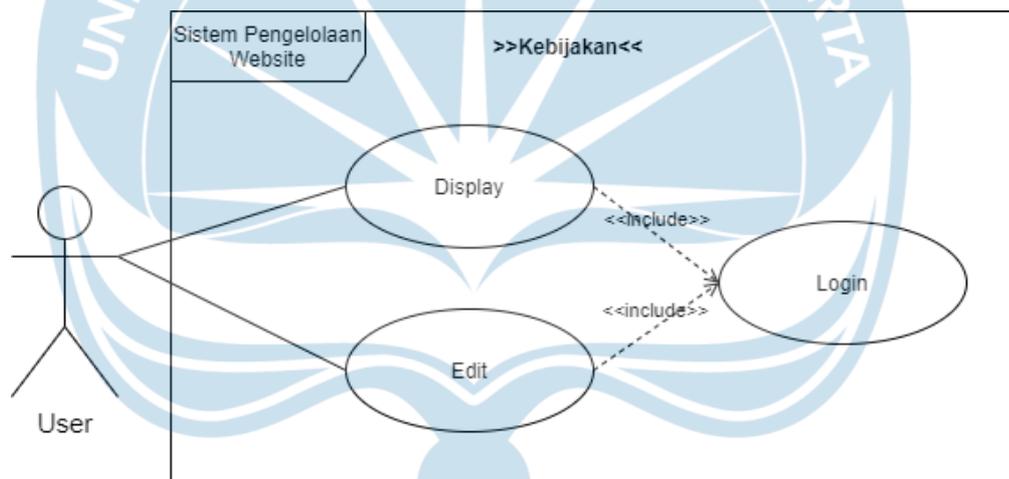
M : Kita disini tim IT nya masih kecil, baru 3 orang terus masih ada kerjaan lain juga jadi lumayan susah, ini aja baru beberapa bulan lalu kita nambah 1 orang, sama tambah kalian yang magang jadi gua arahin untuk buat *web* sama *app* dulu.

A : Oh, oke Pak.

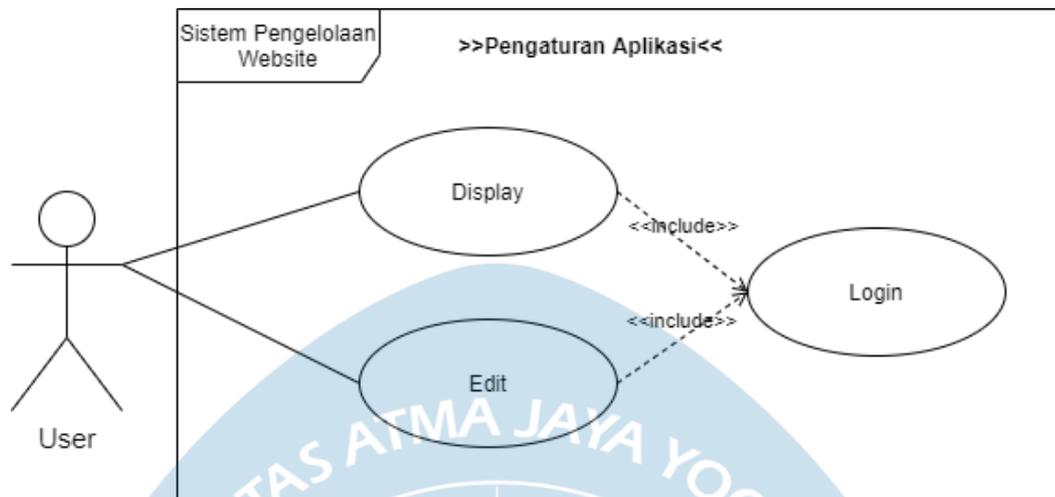
- M : Oke ya, jadi lanjut dulu, nah selain *web* sama *app* tadi, yang *web* itu ada 1 lagi yaitu *web CMS*. Sistem ini dipakenya untuk ngelola isi *web* sama *app* tadi.
- A : Jadi biar ngubah isinya nggak perlu ngubah *code* lagi ya Pak.
- M : Iya, jadi *web* sama *app* nya nanti dibikin sebisa mungkin nggak ada yang *hard code*, jadi kalo mau ngedit isinya langsung aja melalui sistem itu, soalnya pernah kejadian dulu kayak kantor kita pindah alamat doang *user*-nya mesti update aplikasi jadi kan *user experience*-nya kurang enak.
- A : Oke Pak, terus nanti *web* baru nya dibikin ngikutin *web* lama atau bener-bener bikin baru Pak?
- M : Nah, kalo yang sekarang kan dibikinnya oleh pihak luar, terus kalo kita mau lepas dari mereka kan nggak mungkin kita pake *library* mereka semuanya, jadi pastinya kita pake *code* kita sendiri. Untuk sementara sih gua targetin biar bisa ngereplika dulu dari *web & app* mereka. Tapi kalo bisa sih selain kita replika, kita bisa *improve* juga apa yang kurang dari sistemnya mereka, untuk detailnya nanti kapan-kapan gua ajak meeting *project manager*-nya.
- A : Oh, baik Pak berarti saya *explore-explore web* nya dulu ya.
- M : Iya, abis ini *explore* aja dulu terus mungkin besok kita *meet* lagi ngebahas hasil eksplorasinya. Ya udah, sementara gitu dulu aja ya, nanti abis ini gua kasih detail nya untuk kerjaan hari ini ya.
- A : Oke, Pak habis ini saya *explore* lagi, terima kasih pak.



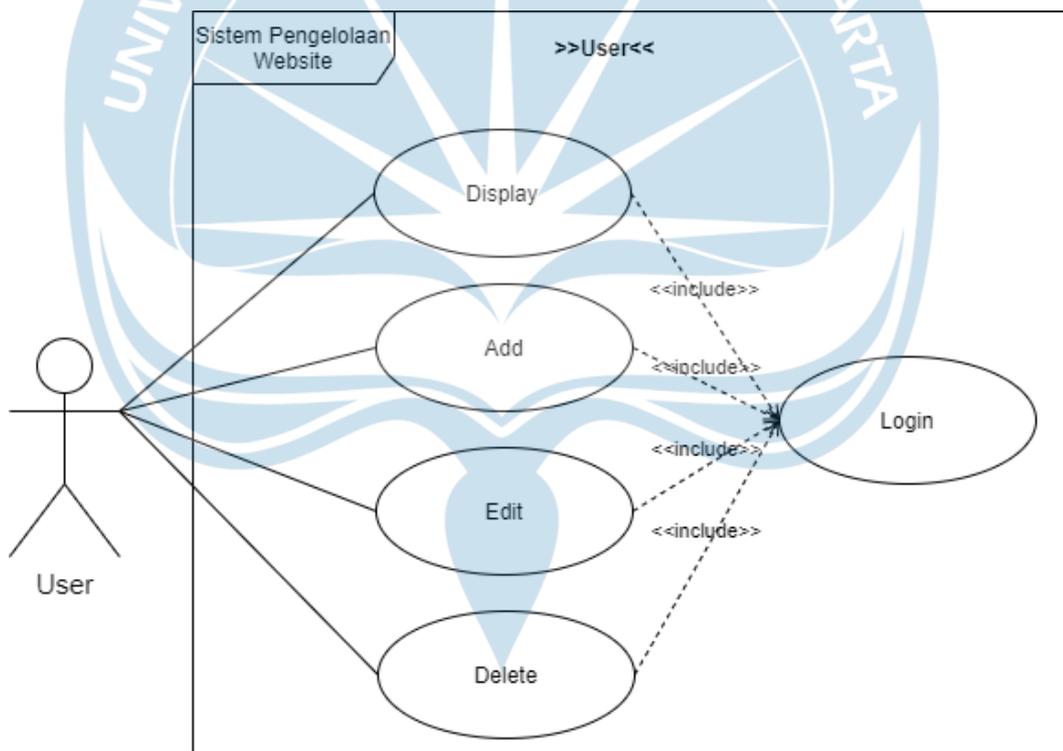
Lampiran 2. Use Case Diagram Level-1 (Tentang Kami)



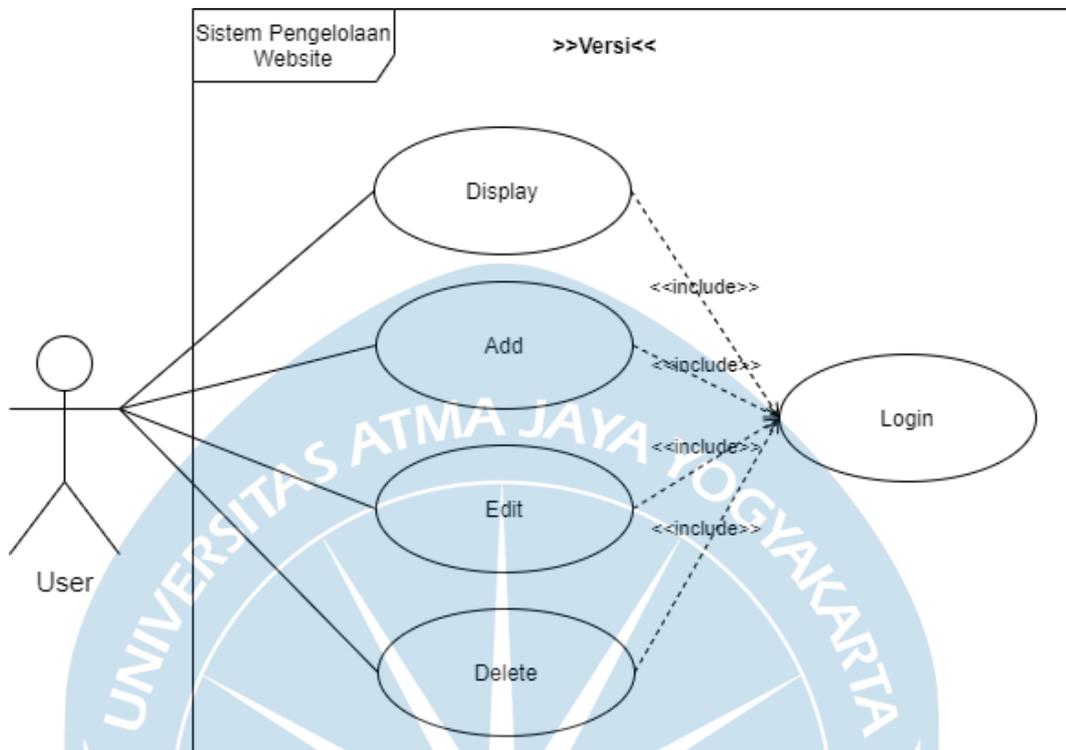
Lampiran 3. Use Case Diagram Level-1 (Kebijakan)



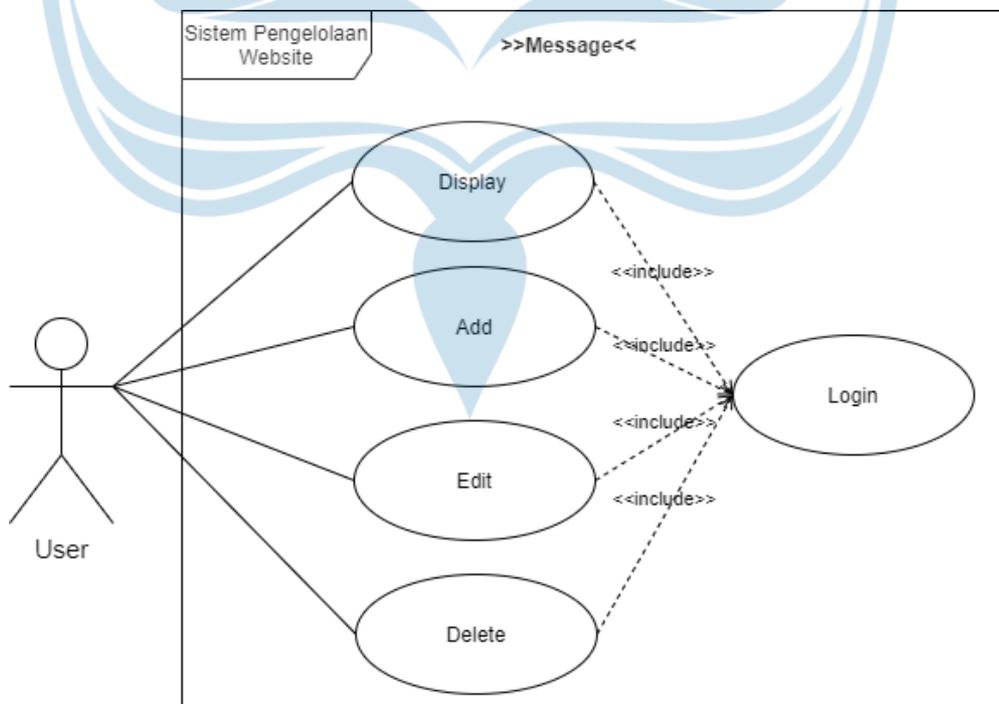
Lampiran 4. Use Case Diagram Level-1 (Pengaturan Aplikasi)



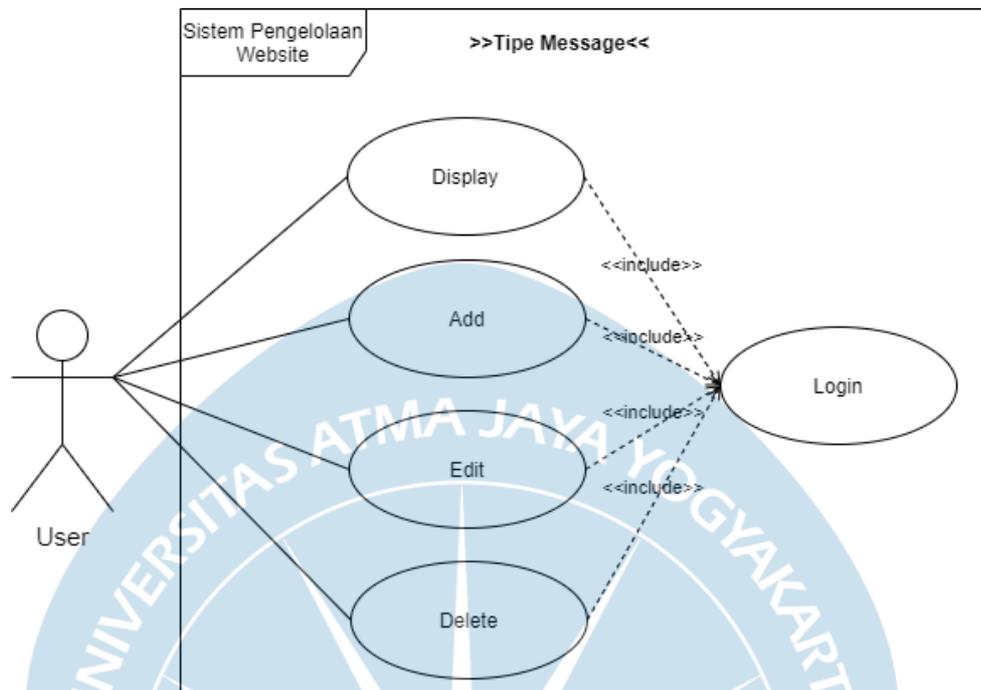
Lampiran 5. Use Case Diagram Level-1 (User)



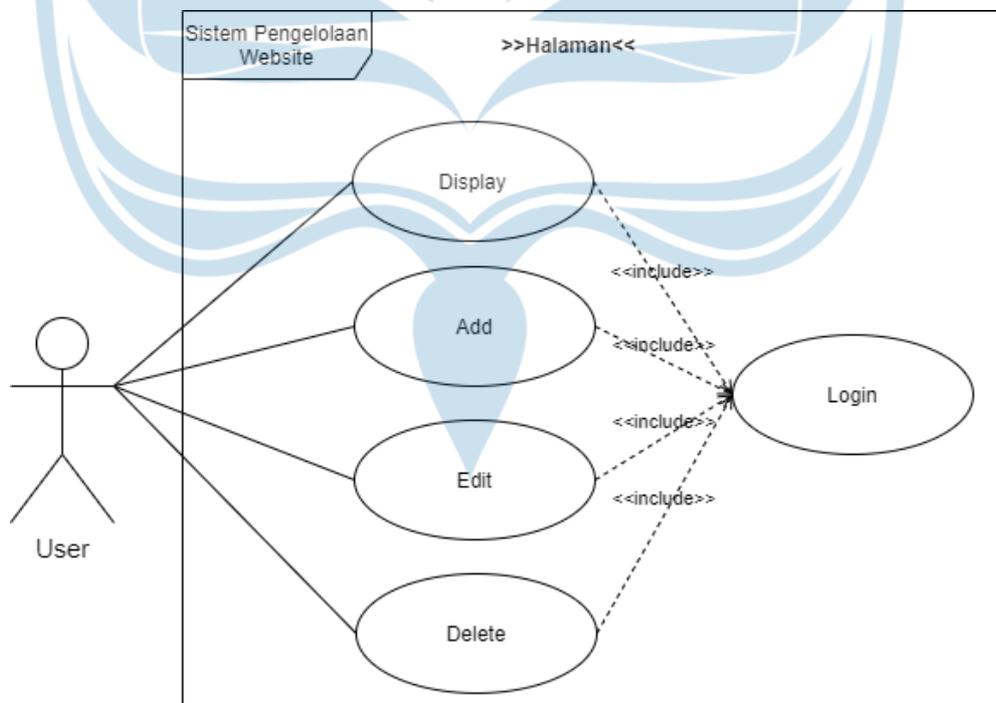
Lampiran 6. Use Case Diagram Level-1 (Versi)



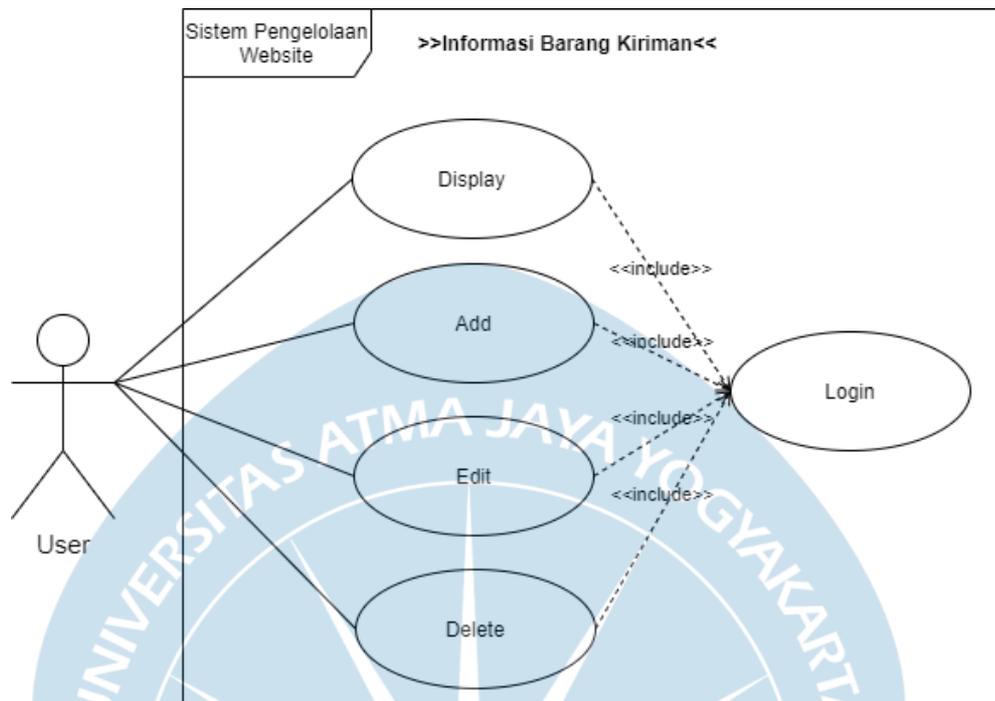
Lampiran 7. Use Case Diagram Level-1 (Message)



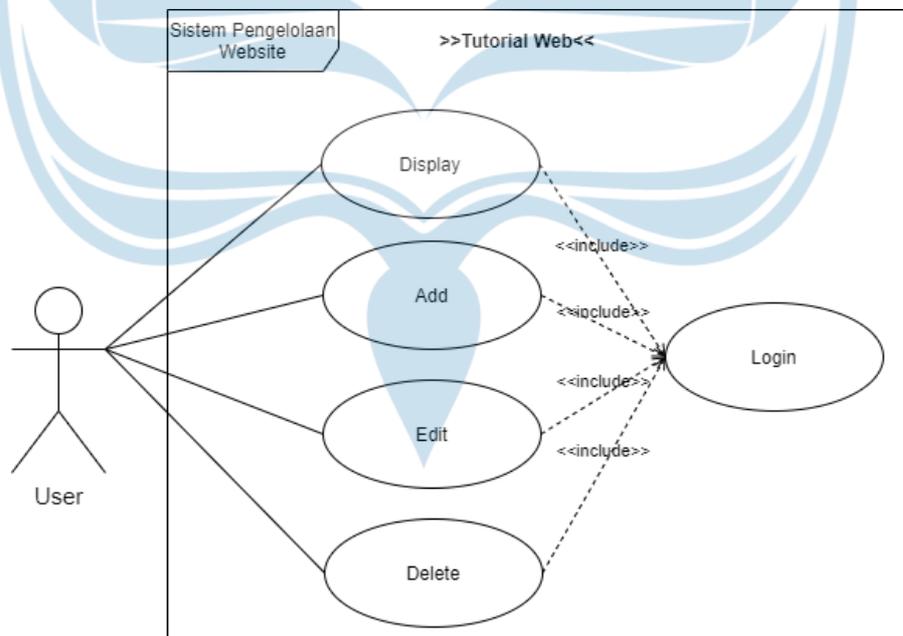
Lampiran 8. Use Case Diagram Level-1 (Tipe Message)



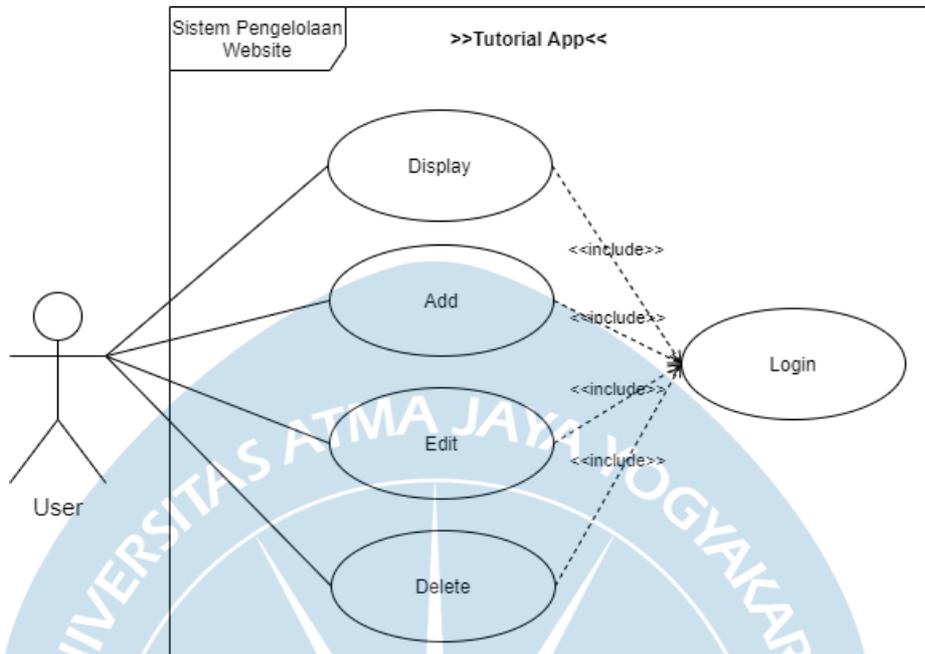
Lampiran 9. Use Case Diagram Level-1 (Halaman)



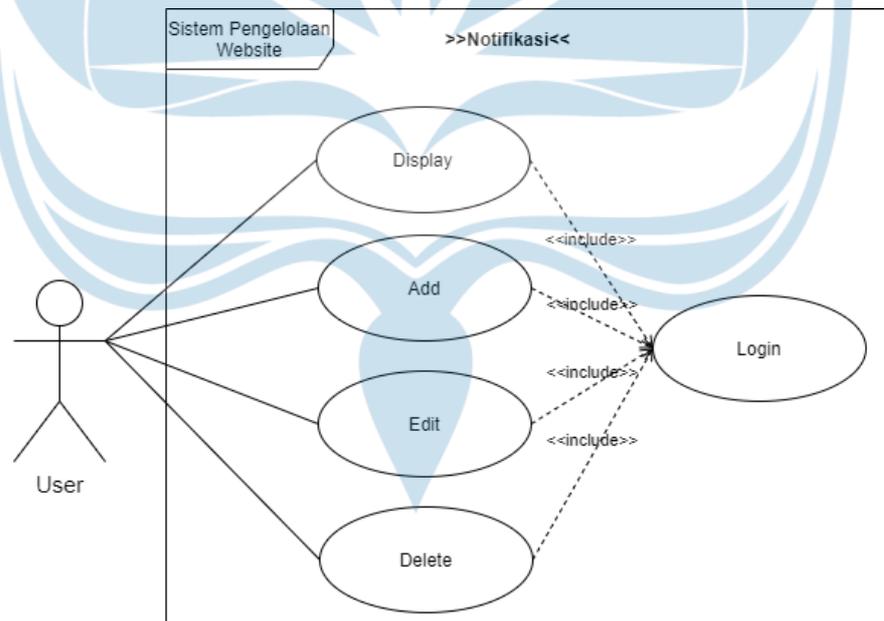
Lampiran 10. Use Case Diagram Level-1 (Informasi Barang Kiriman)



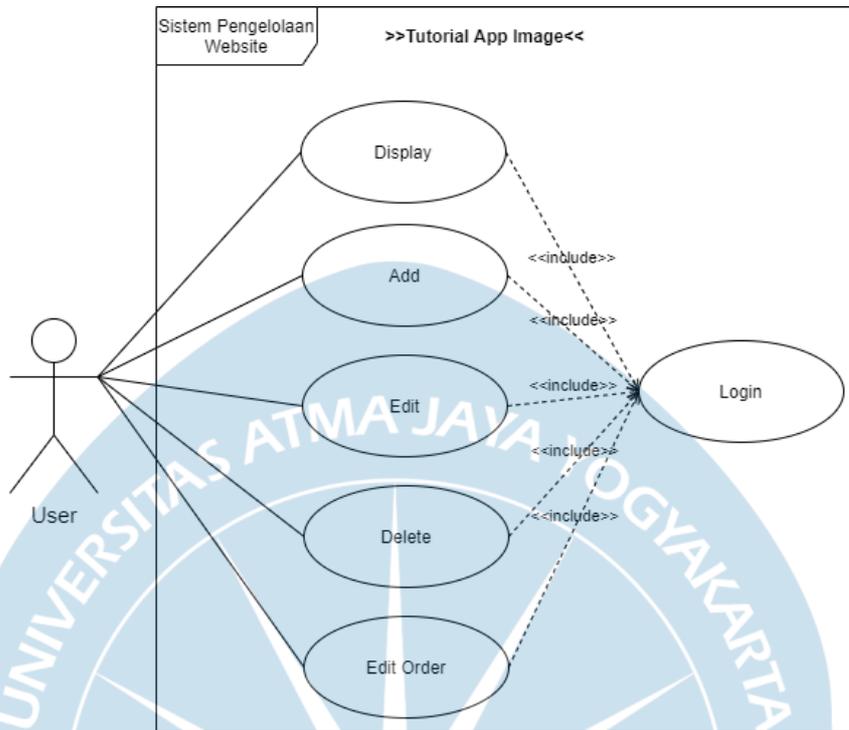
Lampiran 11. Use Case Diagram Level-1 (Tutorial Web)



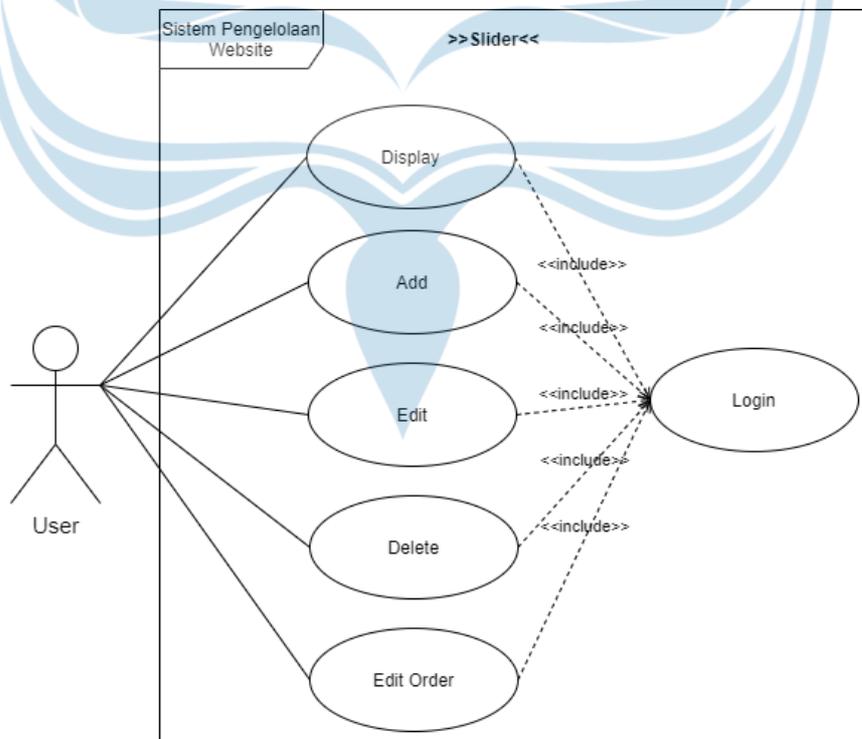
Lampiran 12. Use Case Diagram Level-1 (Tutorial App)



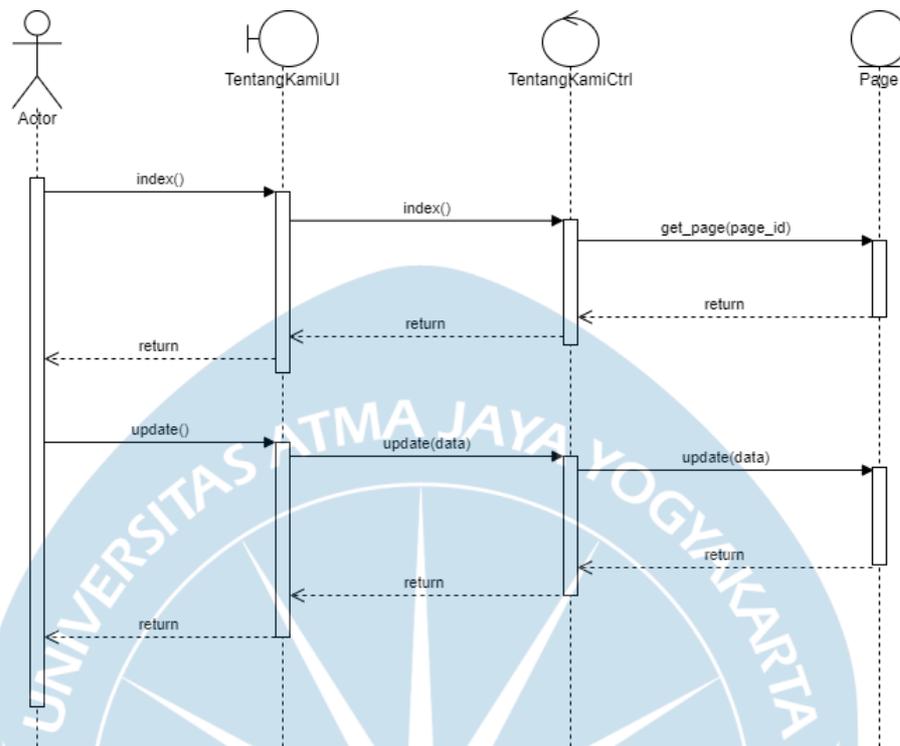
Lampiran 13. Use Case Diagram Level-1 (Notifikasi)



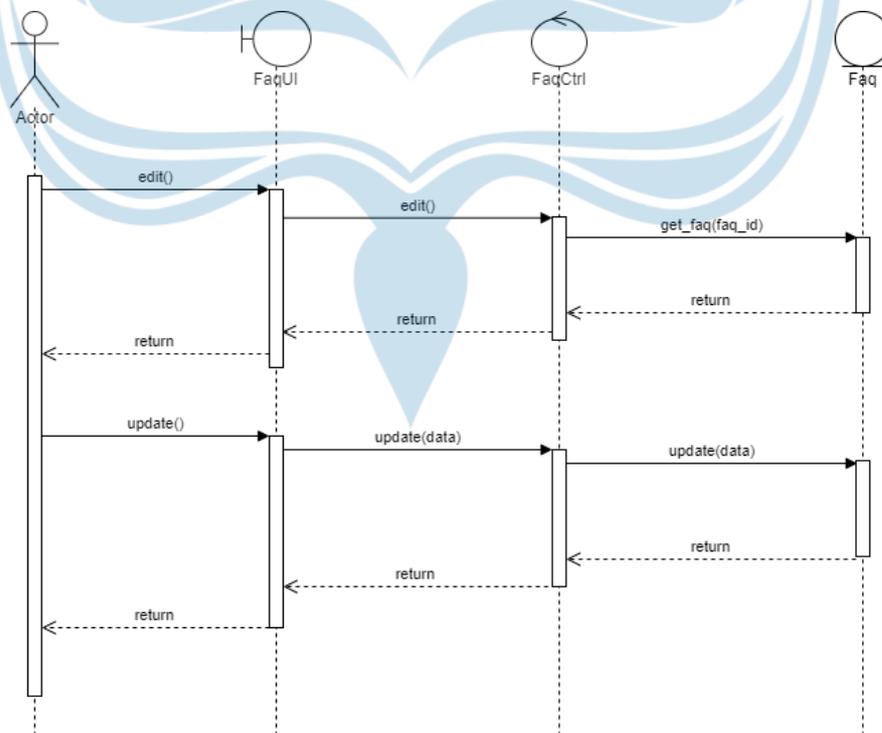
Lampiran 14. Use Case Diagram Level-1 (Tutorial App Image)



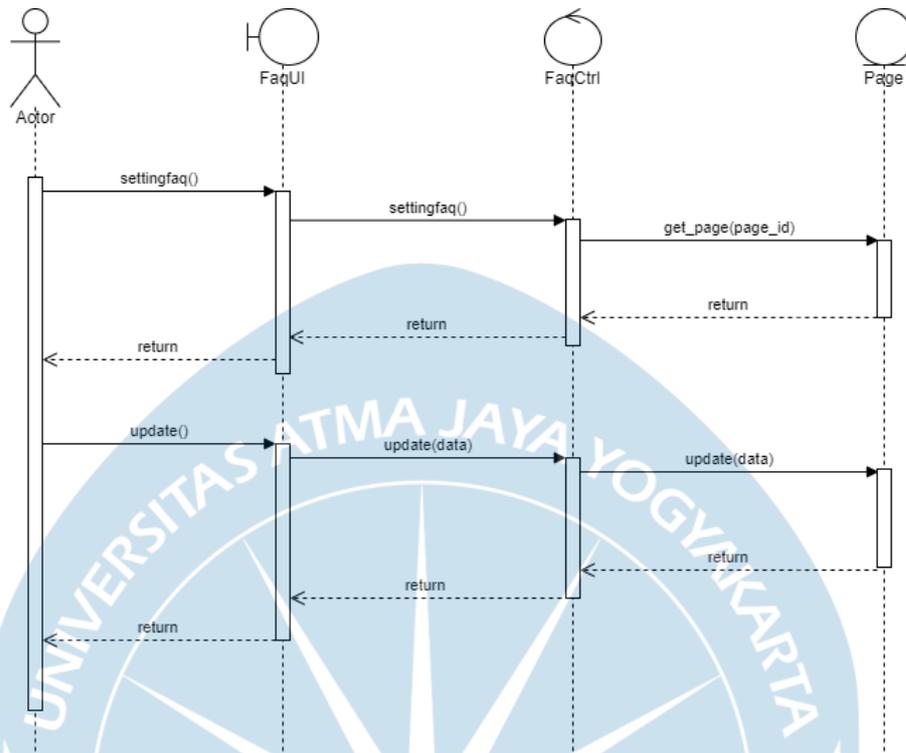
Lampiran 15. Use Case Diagram Level-1 (Slider)



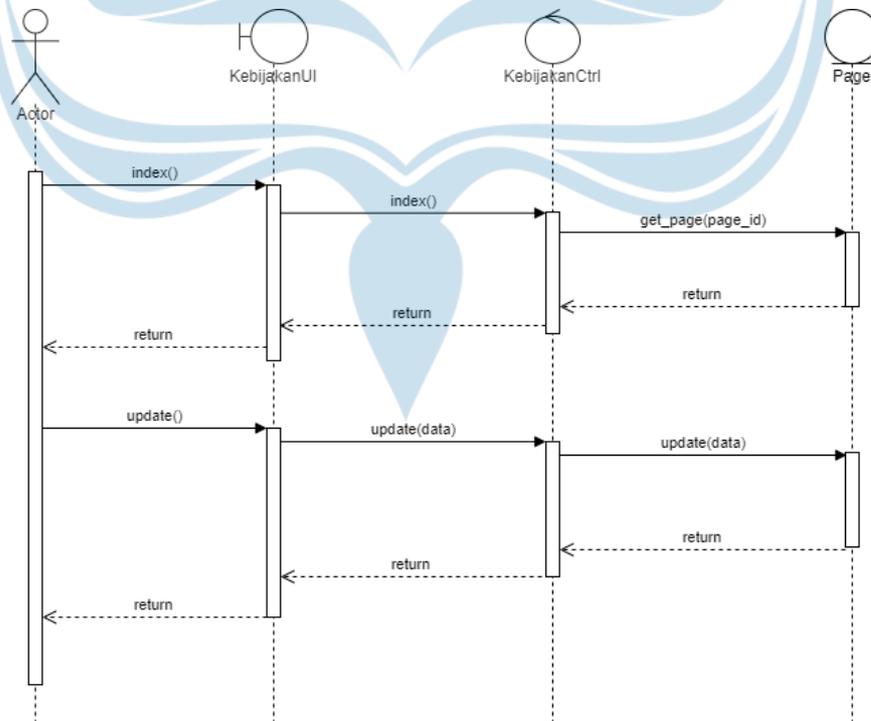
Lampiran 16. Sequence Diagram Tentang Kami



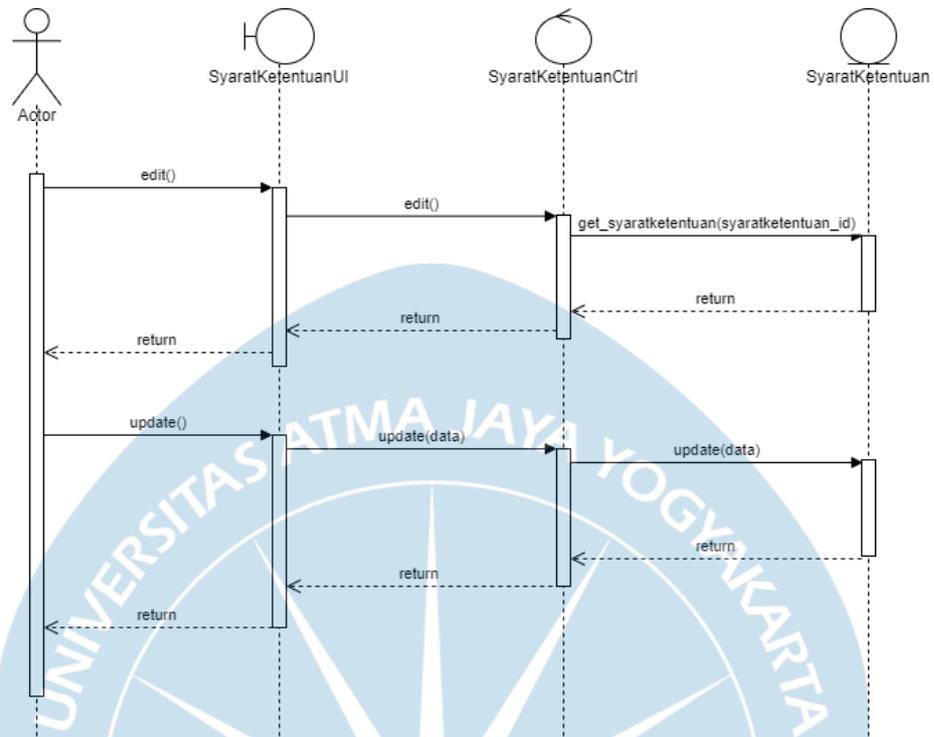
Lampiran 17. Sequence Diagram Edit FAQ



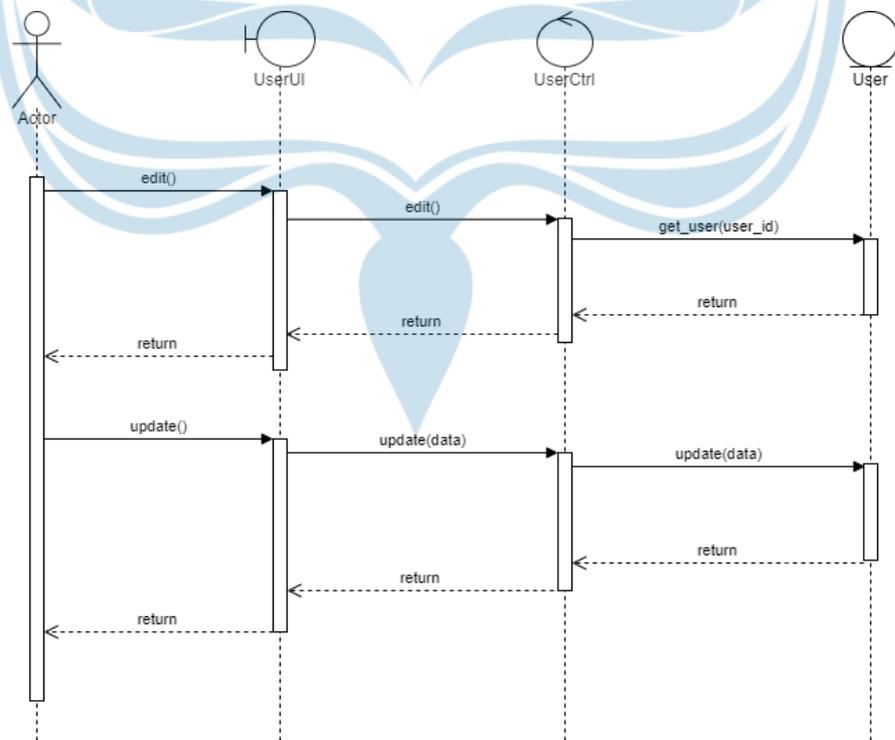
Lampiran 18. Sequence Diagram Setting FAQ



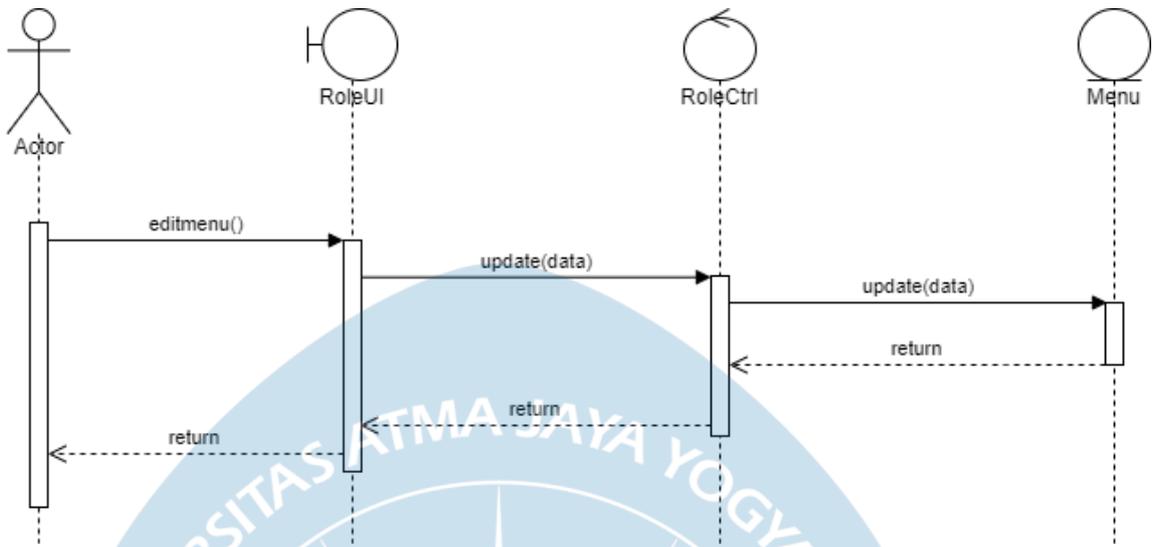
Lampiran 19. Sequence Diagram Kebijakan



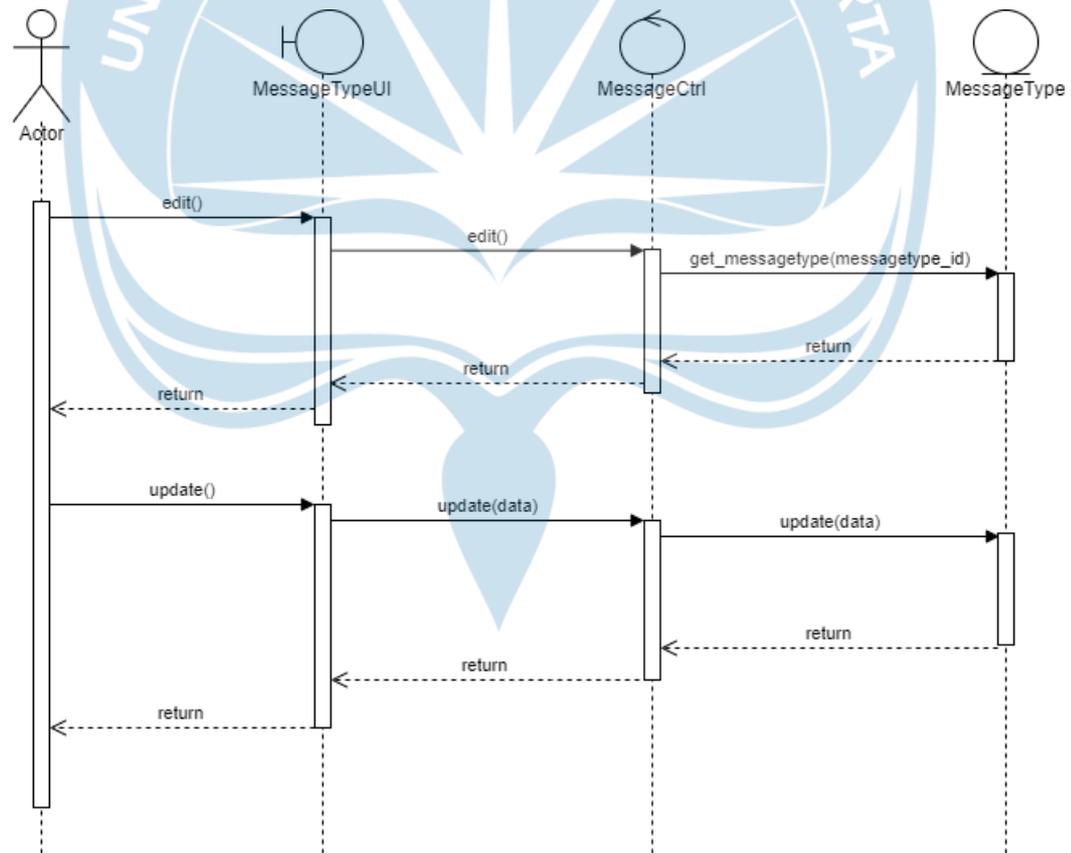
Lampiran 20. Sequence Diagram Edit Syarat Ketentuan



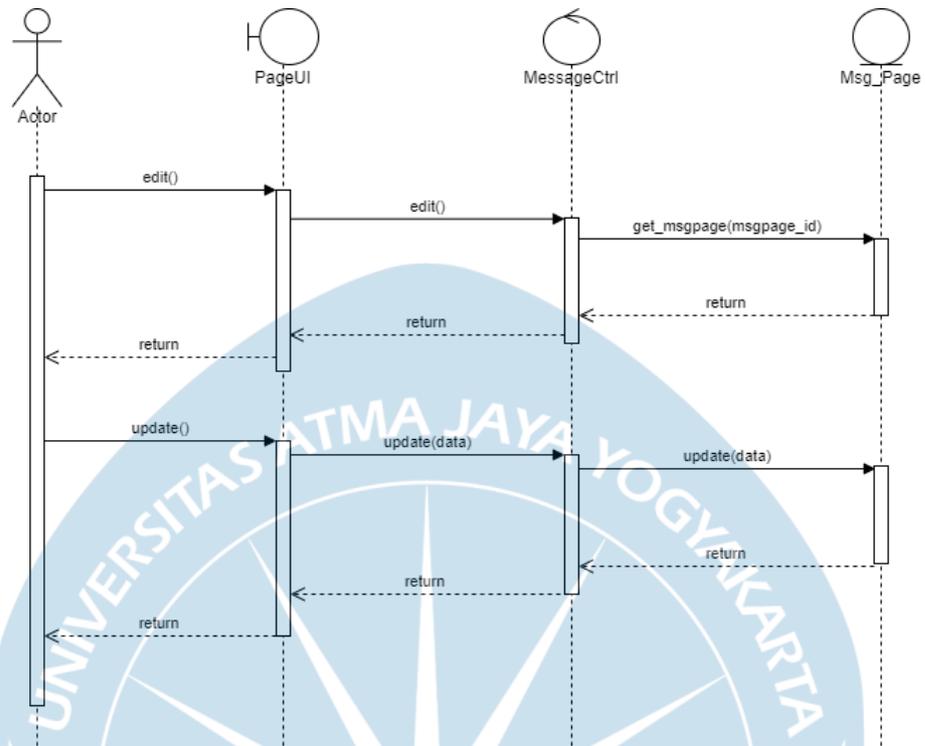
Lampiran 21. Sequence Diagram Edit User



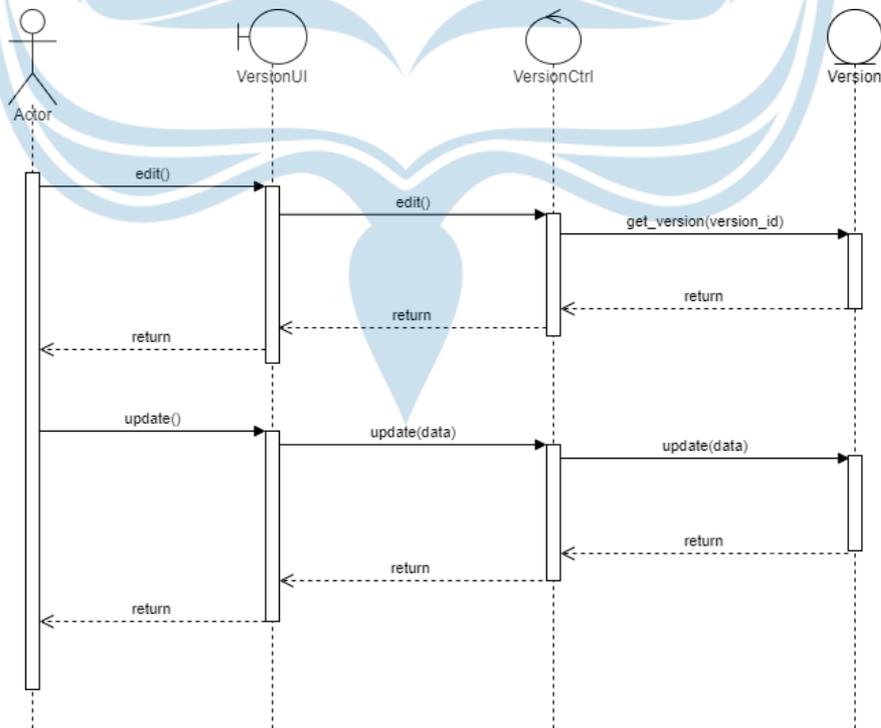
Lampiran 22. Sequence Diagram Edit Menu



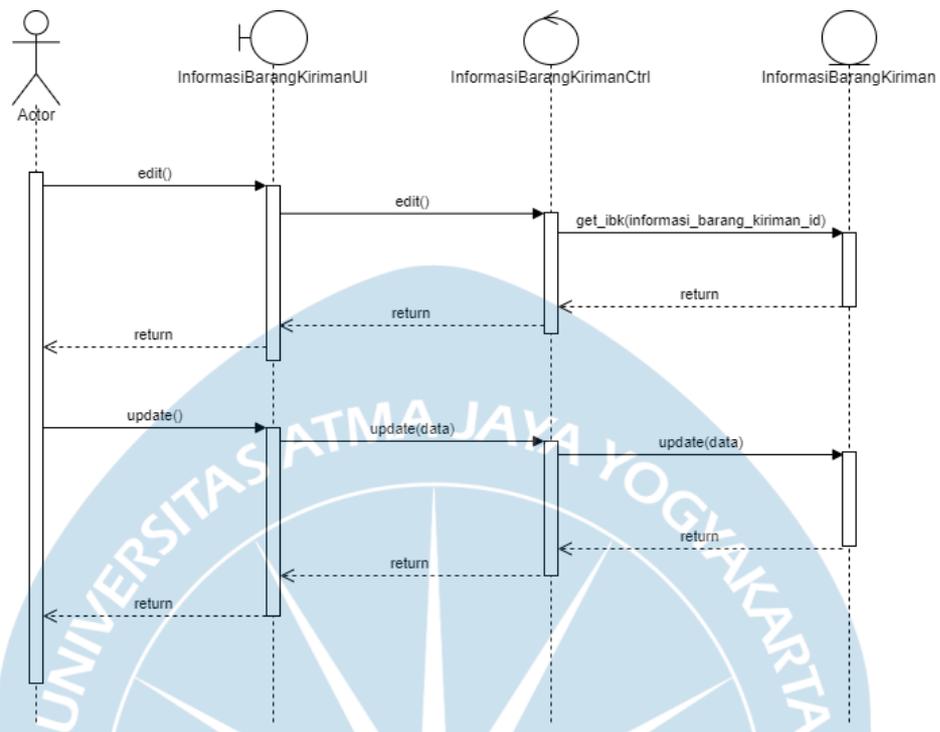
Lampiran 23. Sequence Diagram Edit Message Type



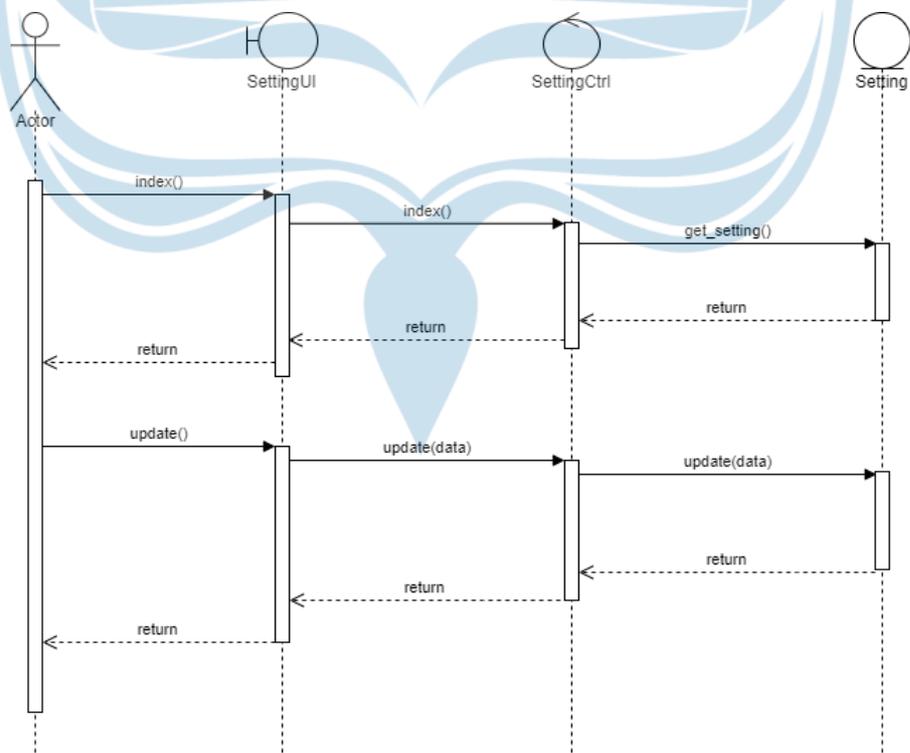
Lampiran 24. Sequence Diagram Edit Halaman



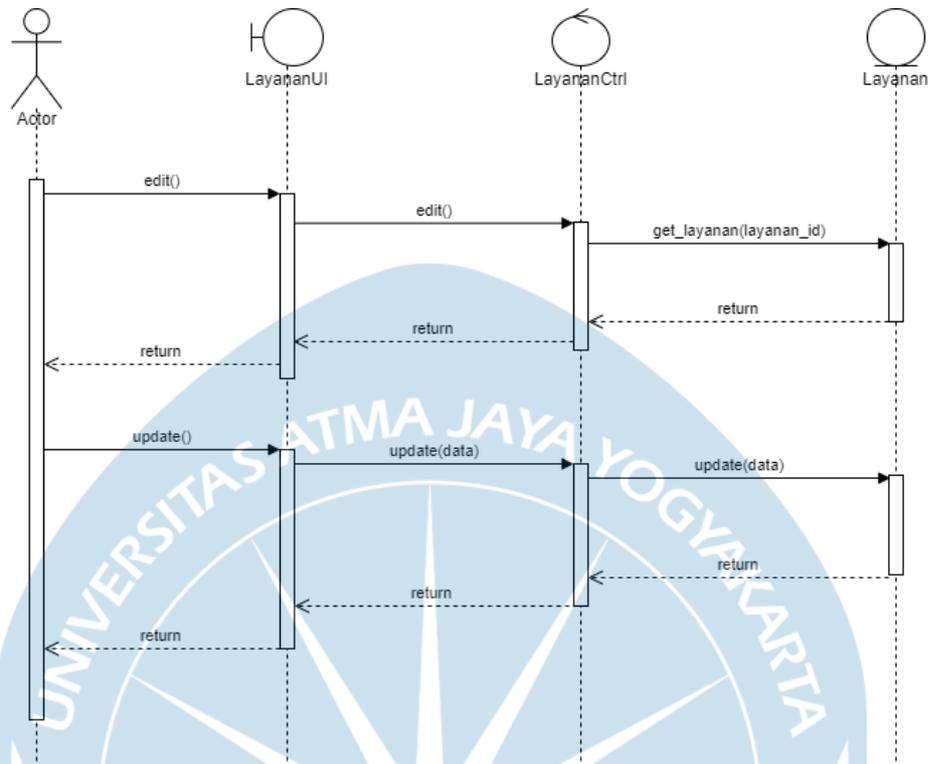
Lampiran 25. Sequence Diagram Edit Version



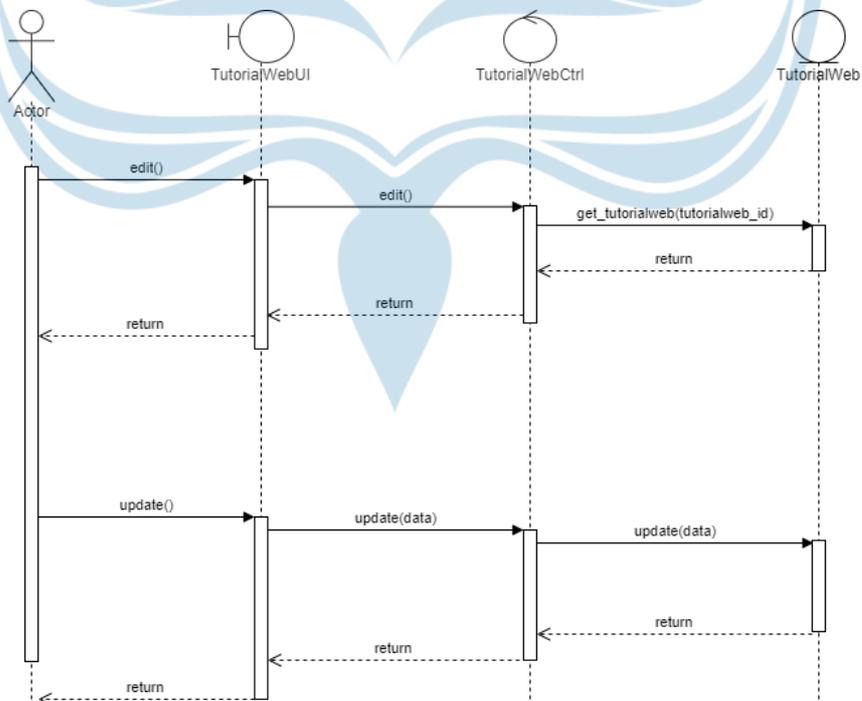
Lampiran 26. Sequence Diagram Edit Informasi Barang Kiriman



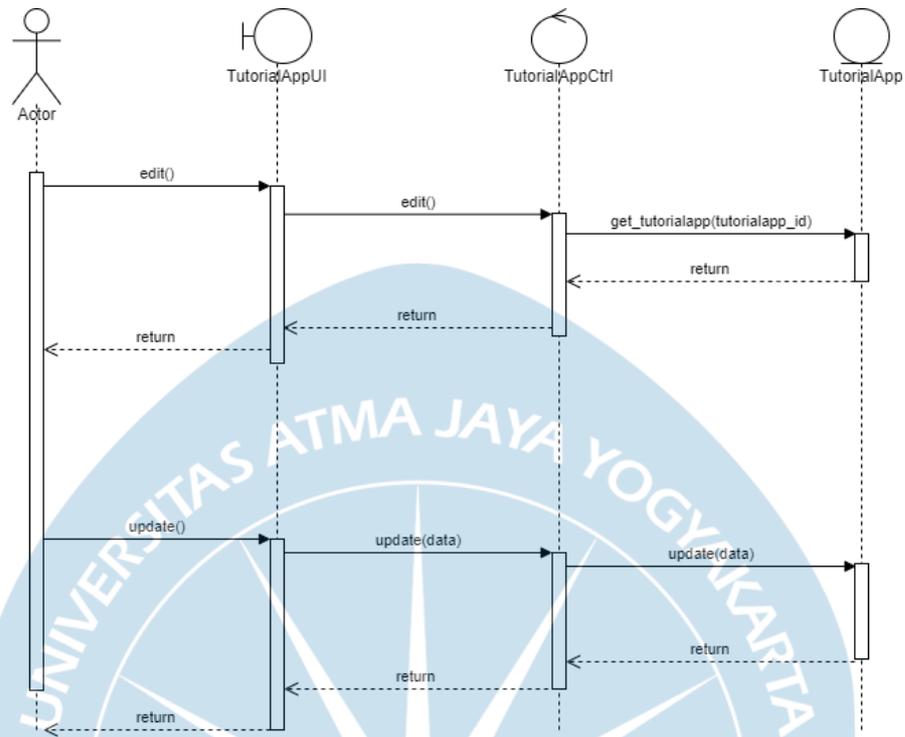
Lampiran 27. Sequence Diagram Pengaturan Aplikasi



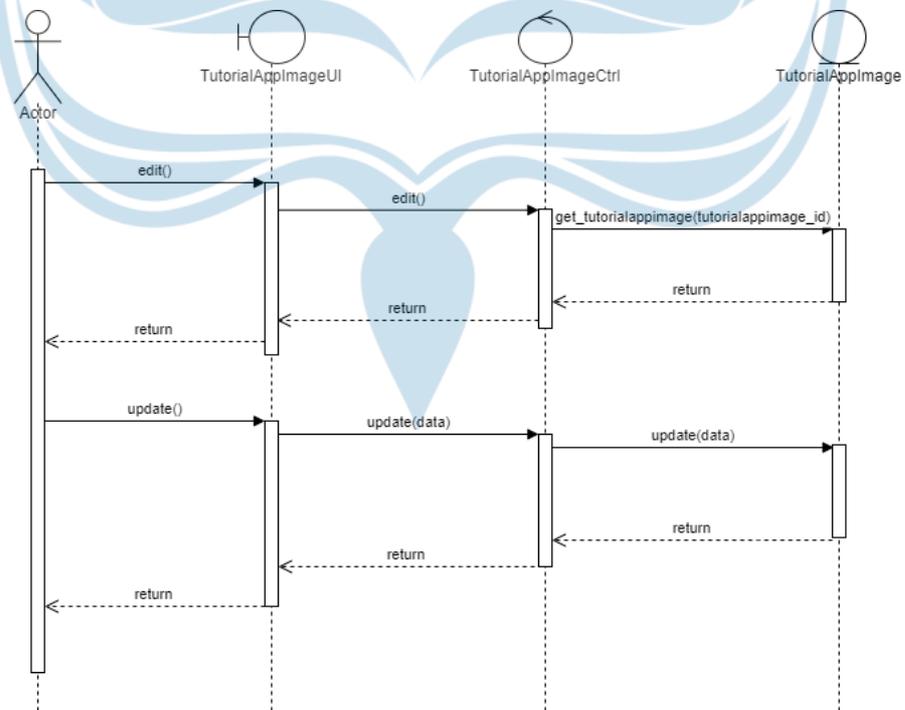
Lampiran 28. Sequence Diagram Edit Layanan



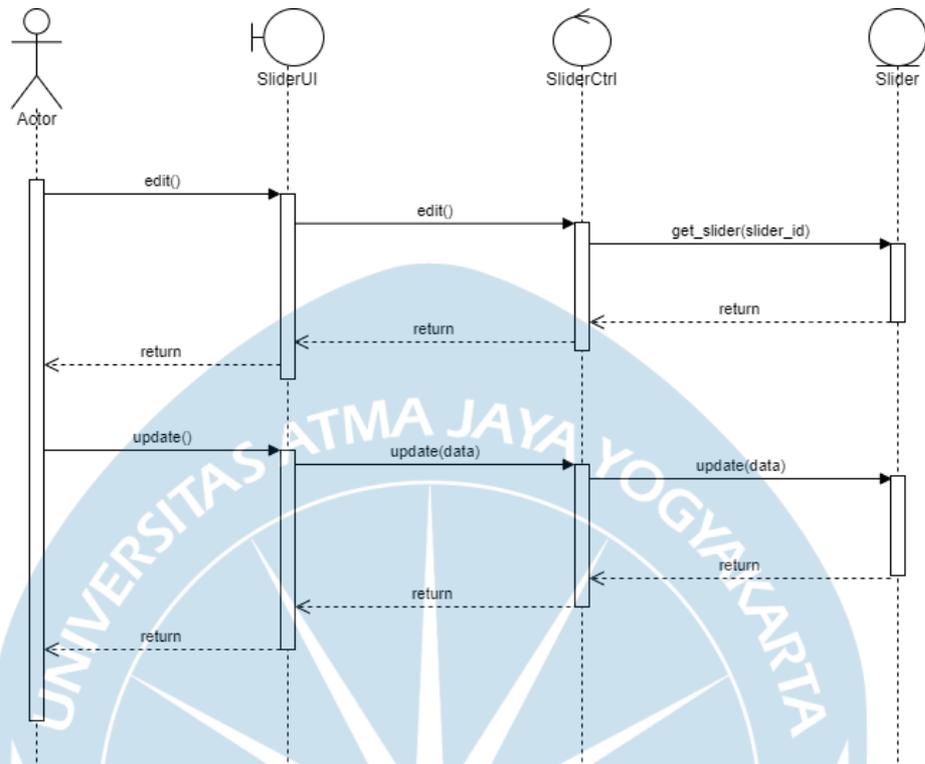
Lampiran 29. Sequence Diagram Edit Tutorial Web



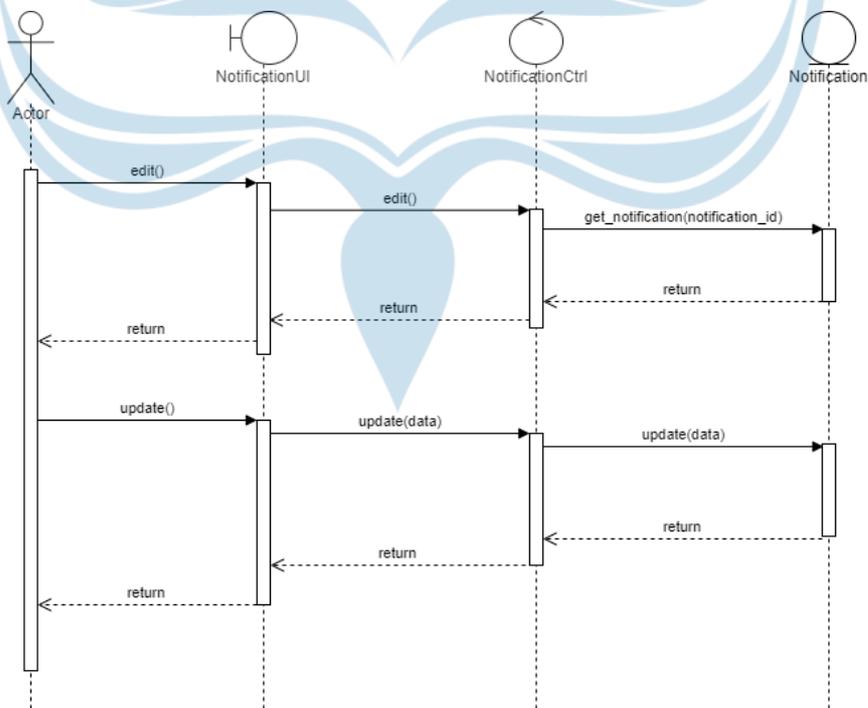
Lampiran 30. Sequence Diagram Edit Tutorial App



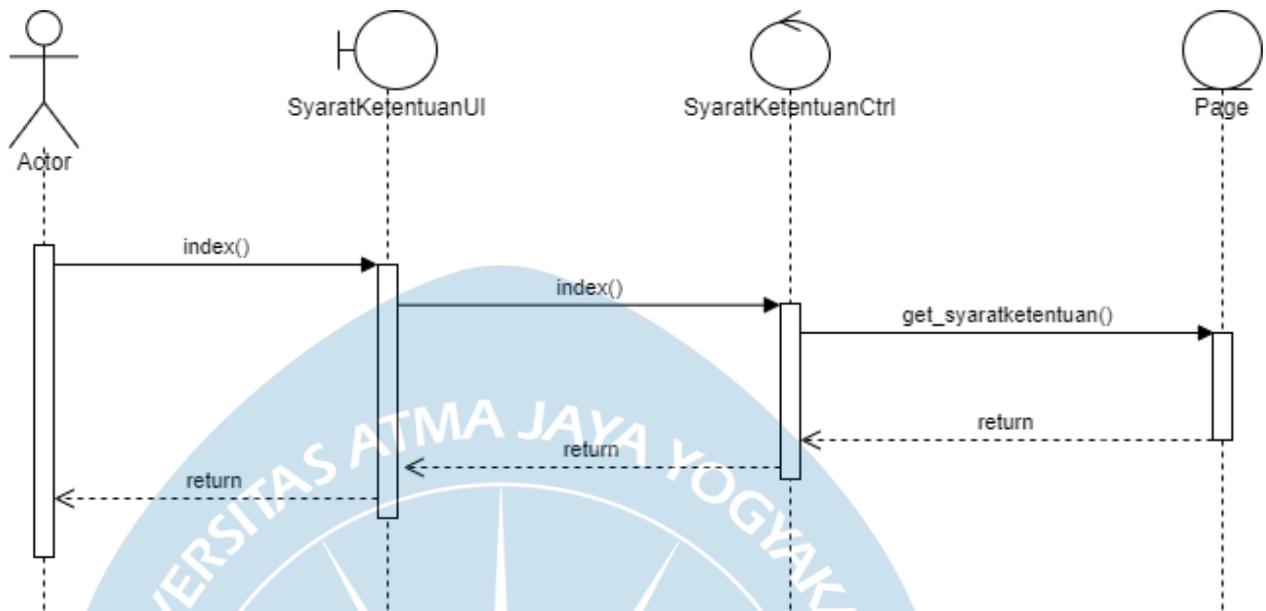
Lampiran 31. Sequence Diagram Edit Tutorial App Image



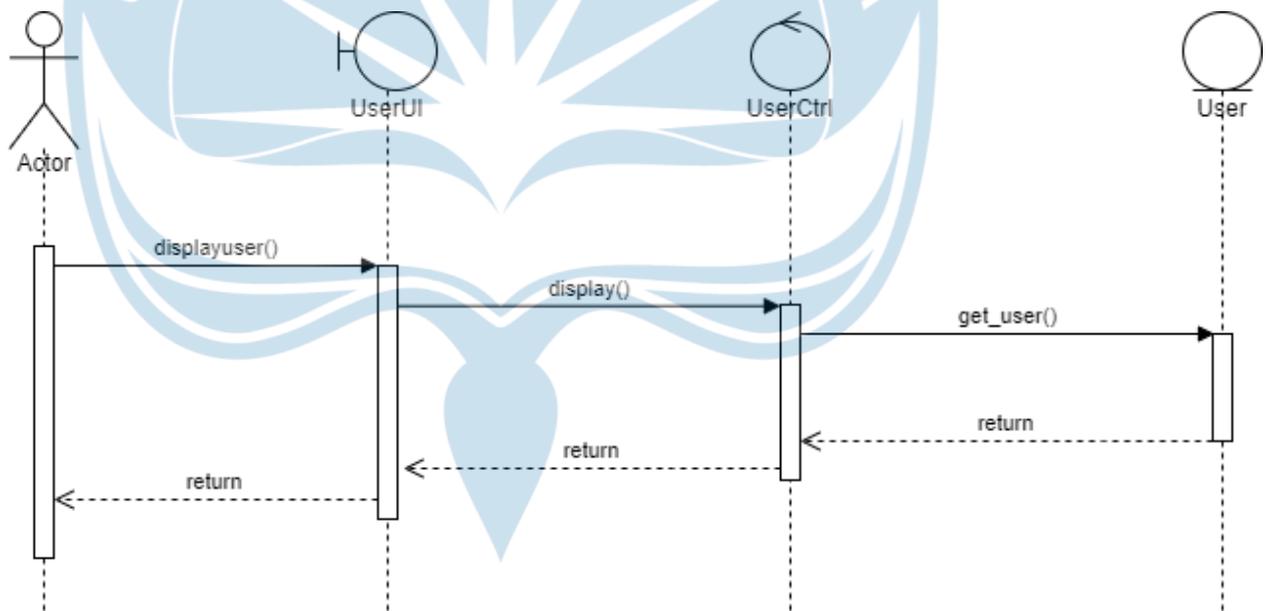
Lampiran 32. Sequence Diagram Edit Slider



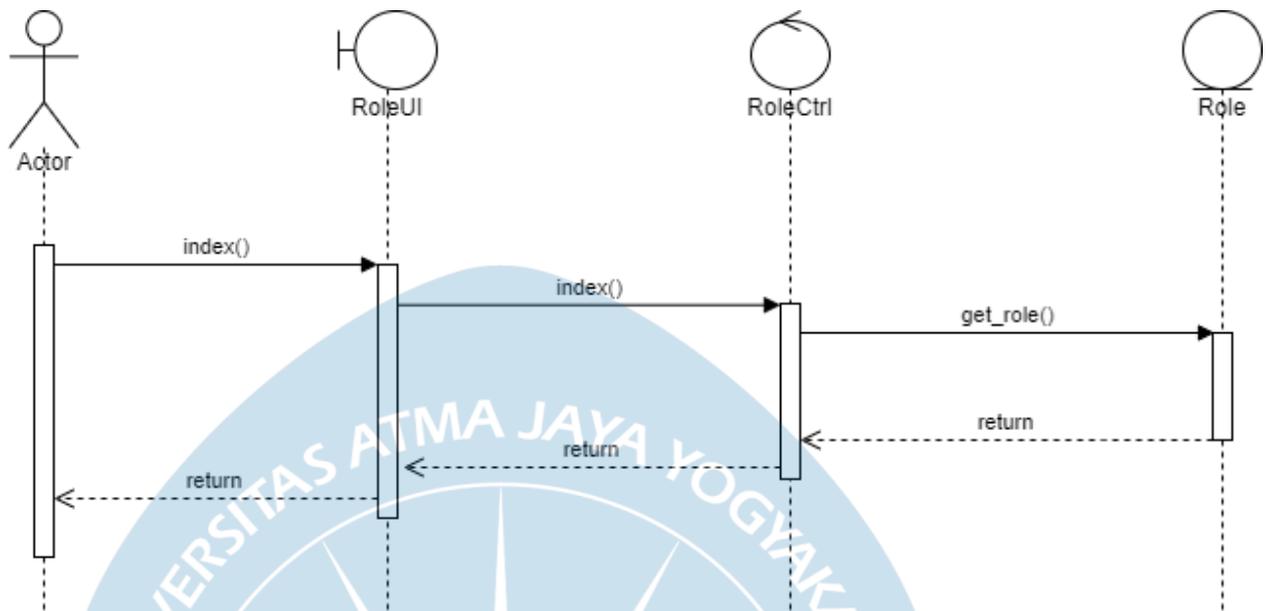
Lampiran 33. Sequence Diagram Edit Notification



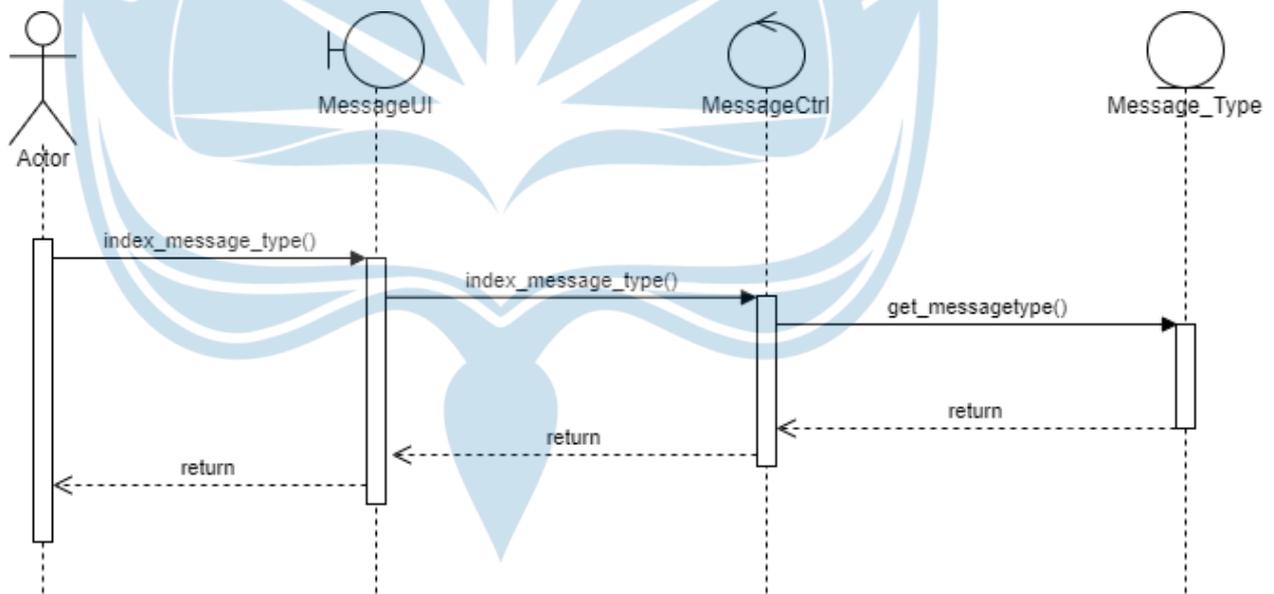
Lampiran 34. Sequence Diagram Display Syarat Ketentuan



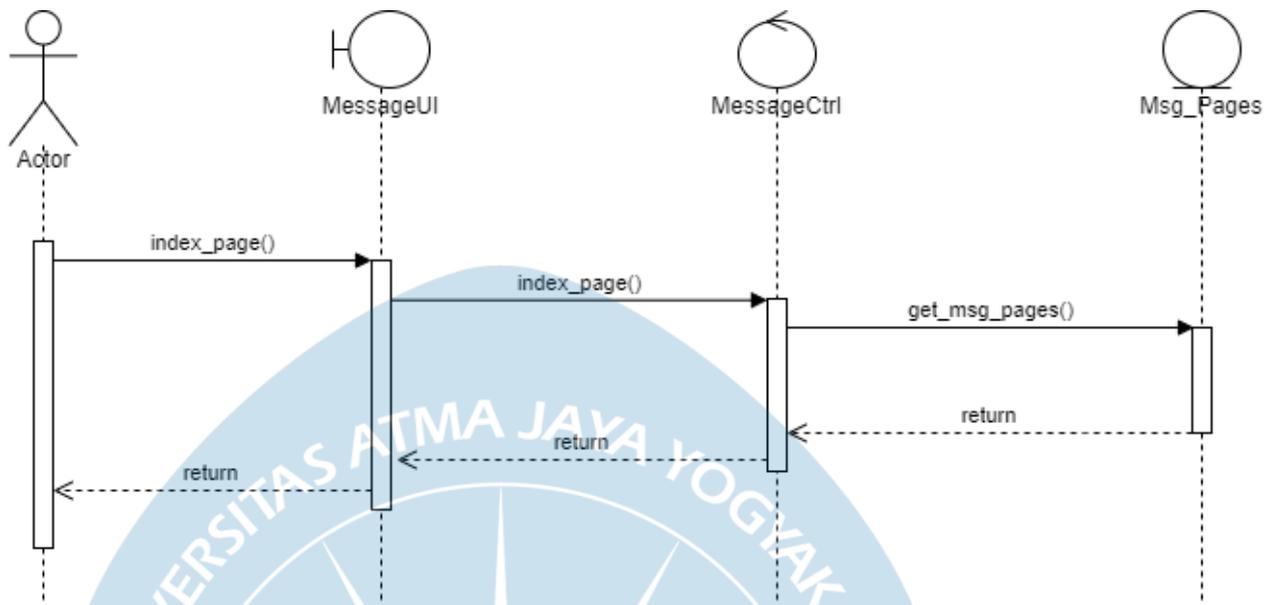
Lampiran 35. Sequence Diagram Display User



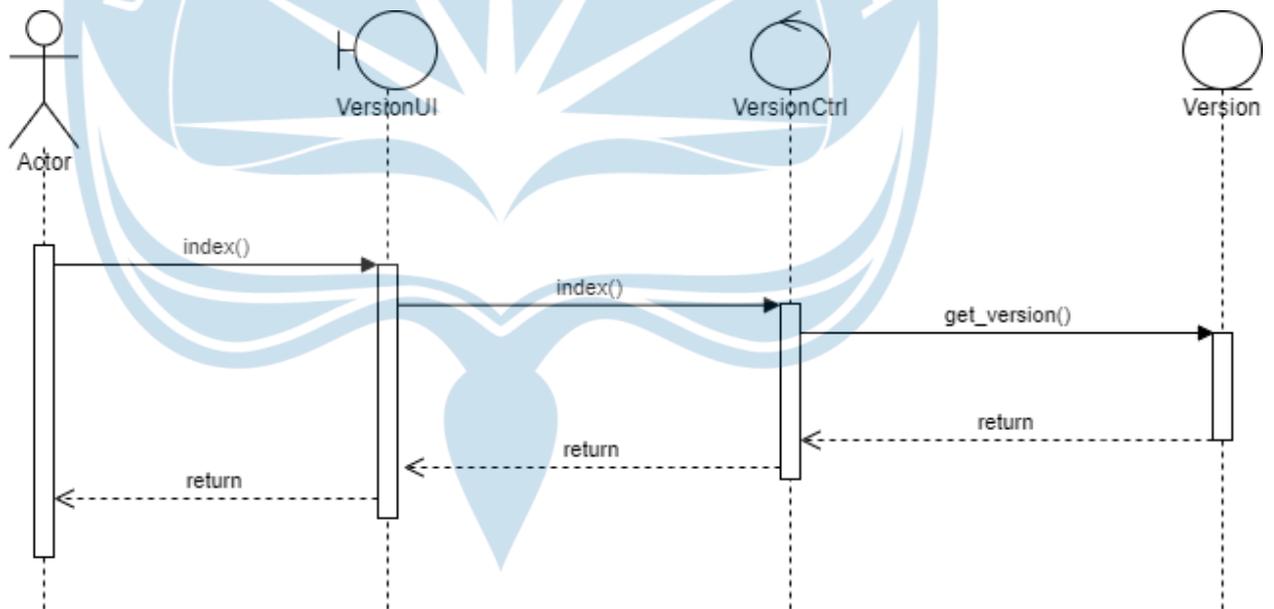
Lampiran 36. Sequence Diagram Display Role



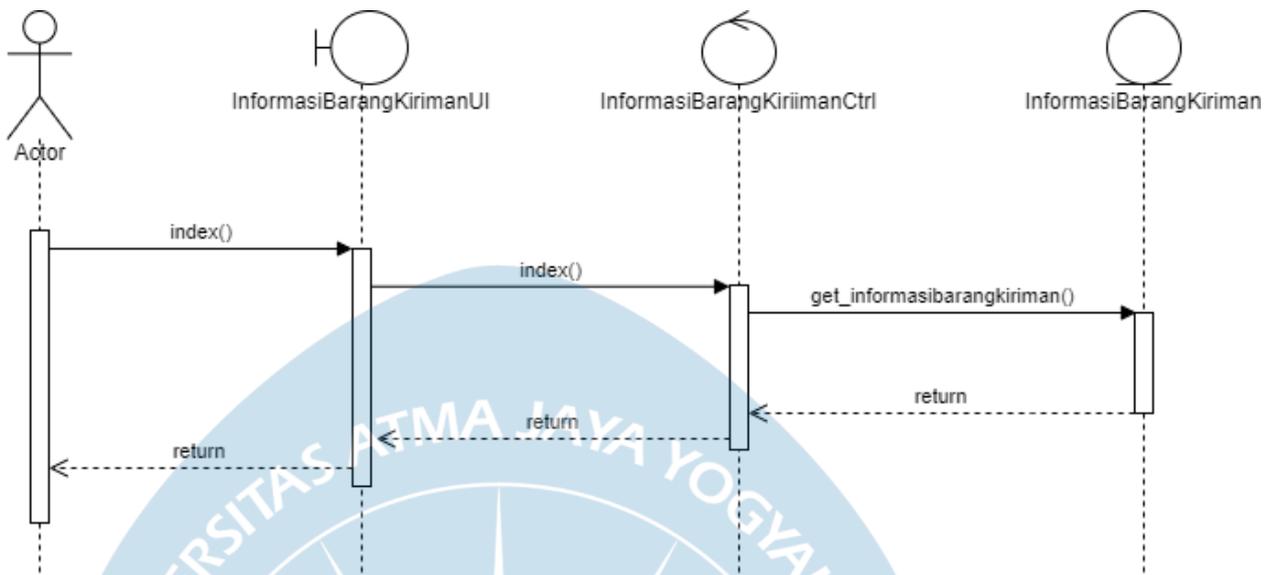
Lampiran 37. Sequence Diagram Display Message Type



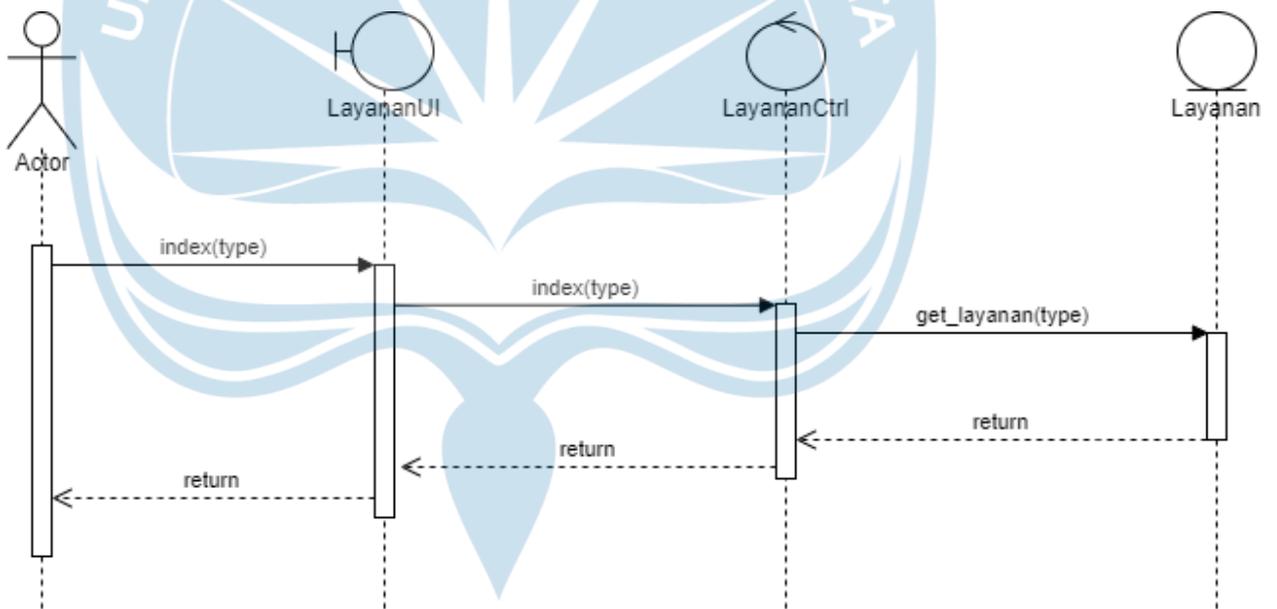
Lampiran 38. Sequence Diagram Display Halaman



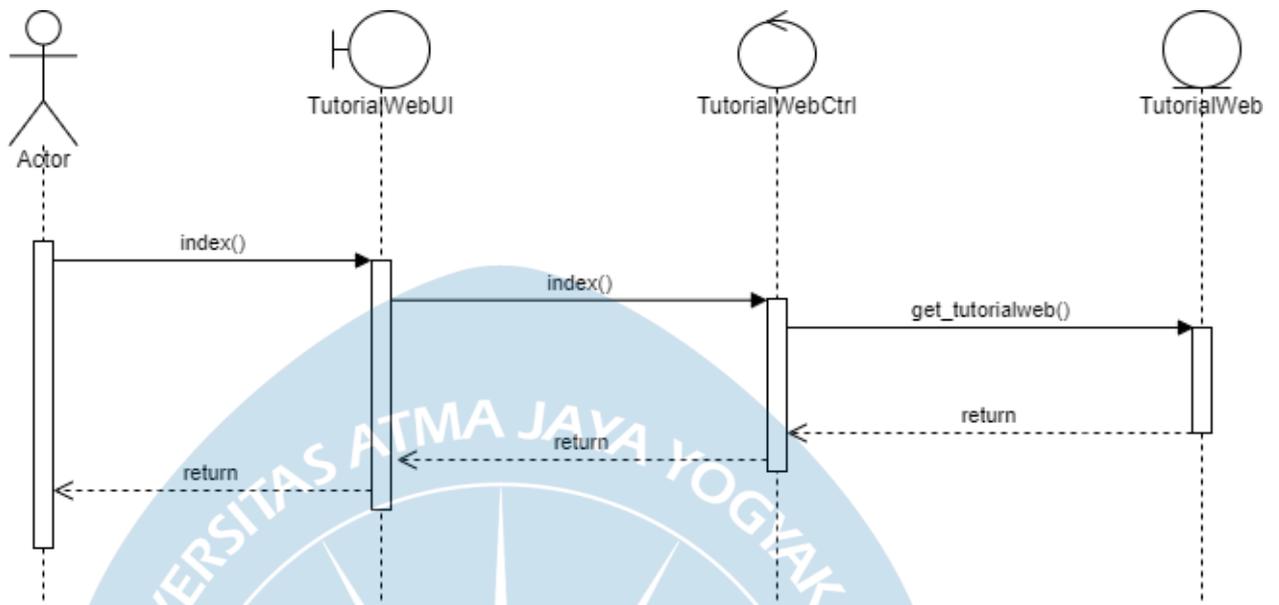
Lampiran 39. Sequence Diagram Display Version



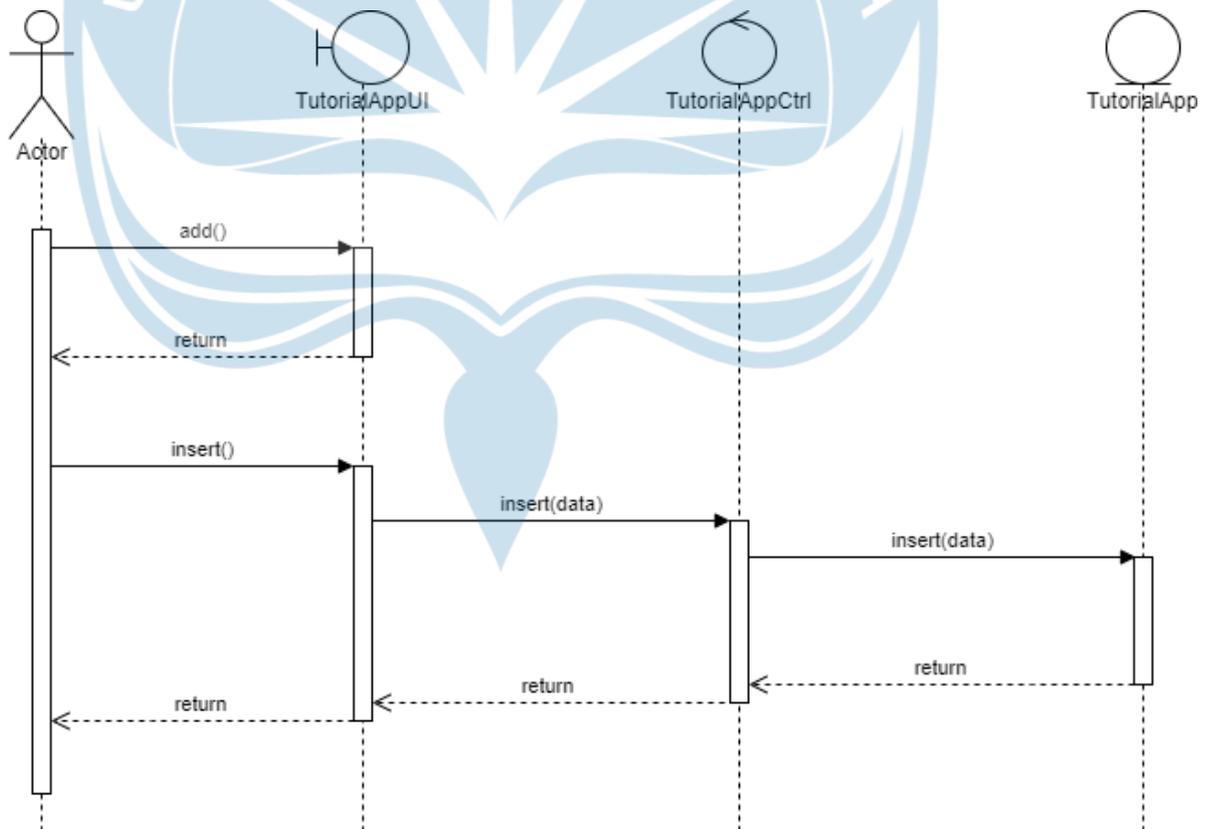
Lampiran 40. Sequence Diagram Display Informasi Barang Kiriman



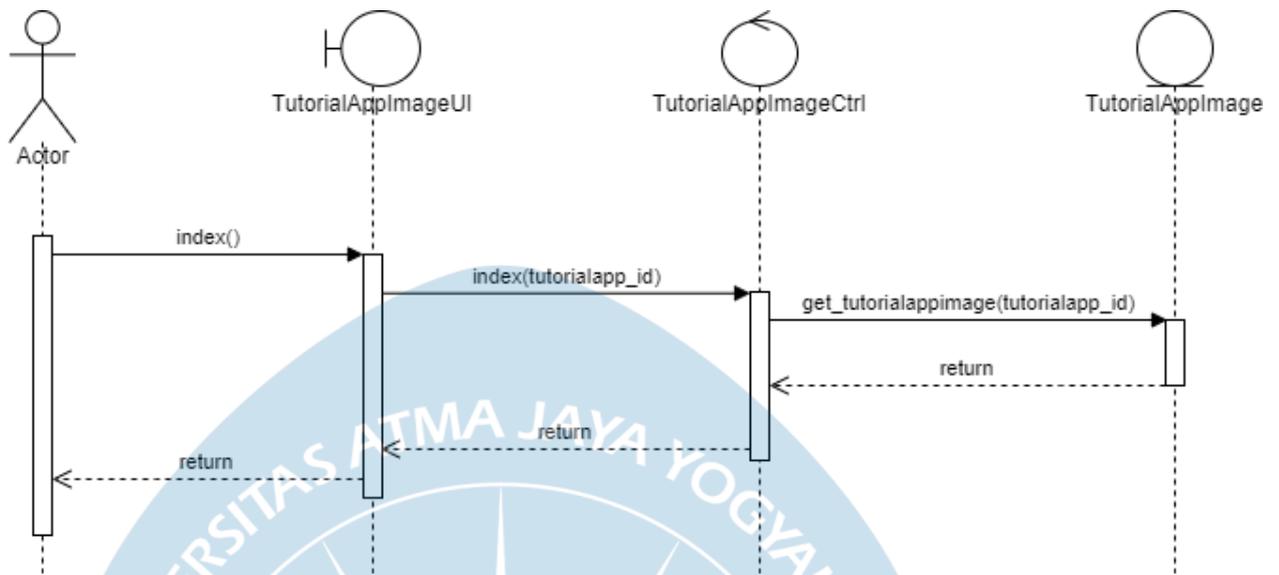
Lampiran 41. Sequence Diagram Display Layanan



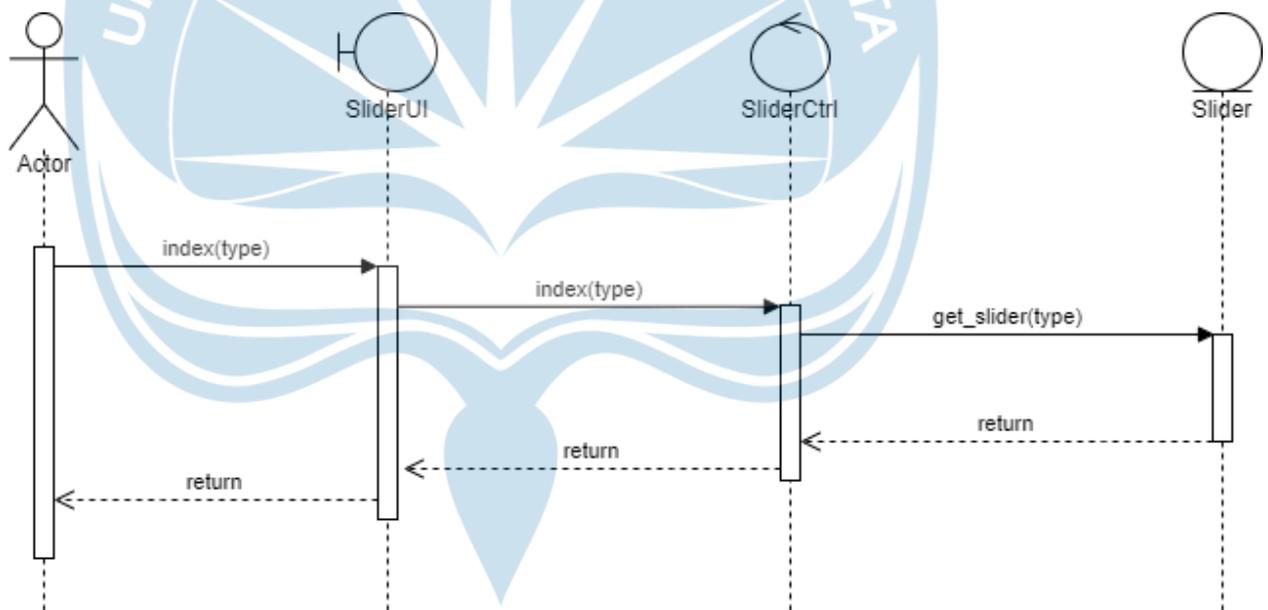
Lampiran 42. Sequence Diagram Display Tutorial Web



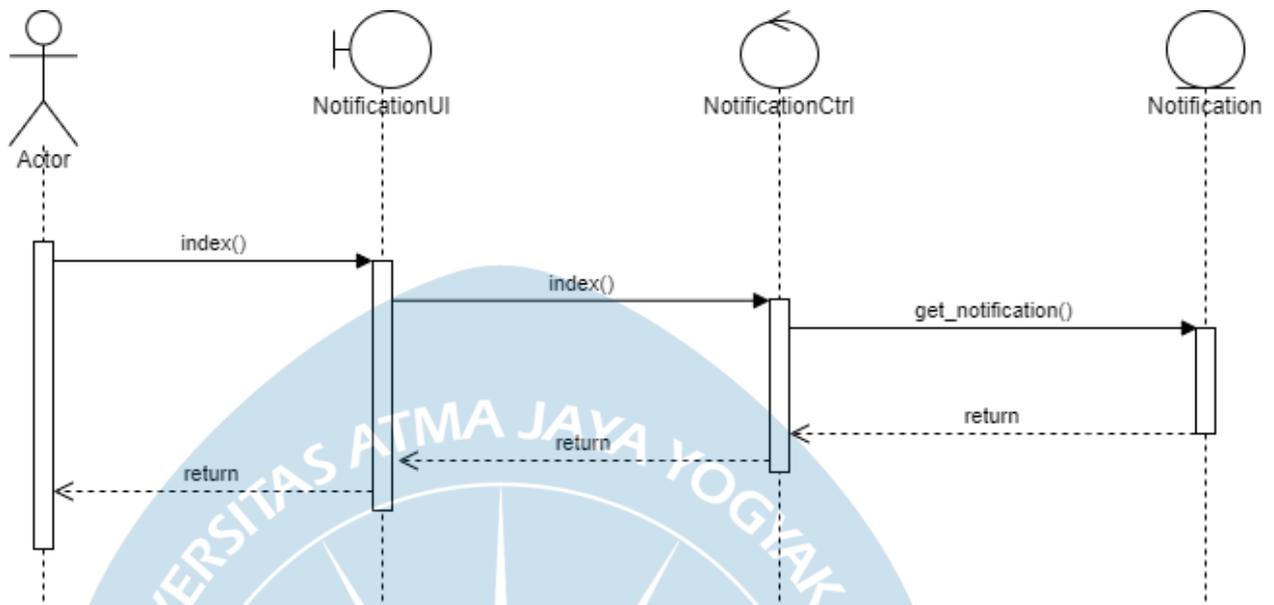
Lampiran 43. Sequence Diagram Display Tutorial App



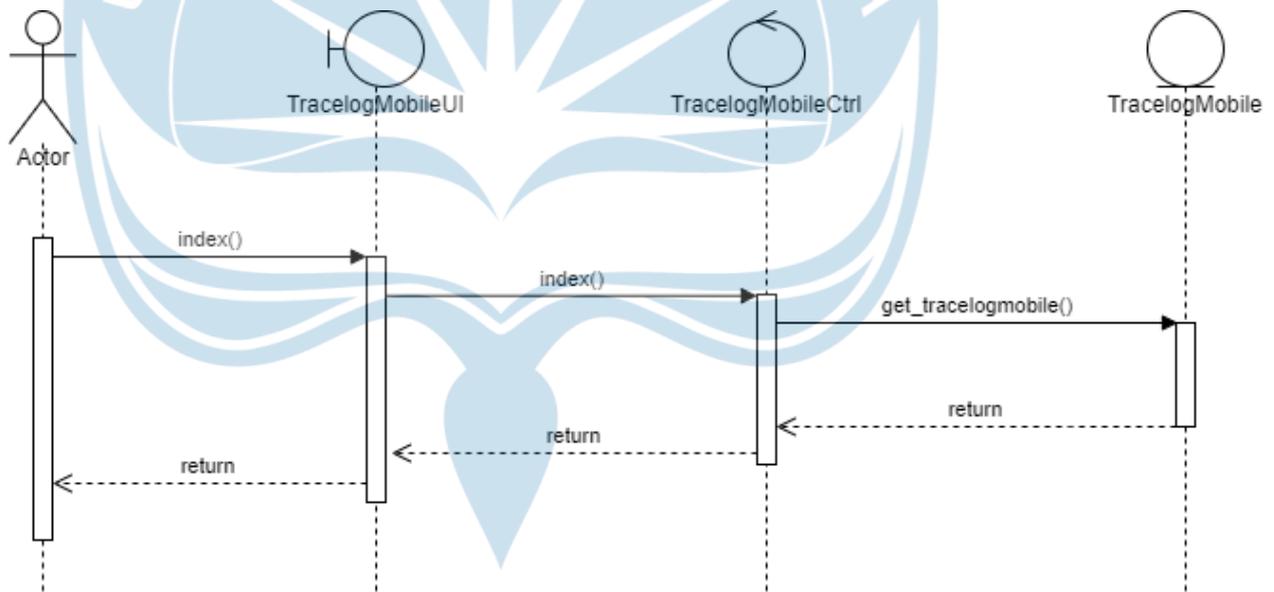
Lampiran 44. Sequence Diagram Display Tutorial App Image



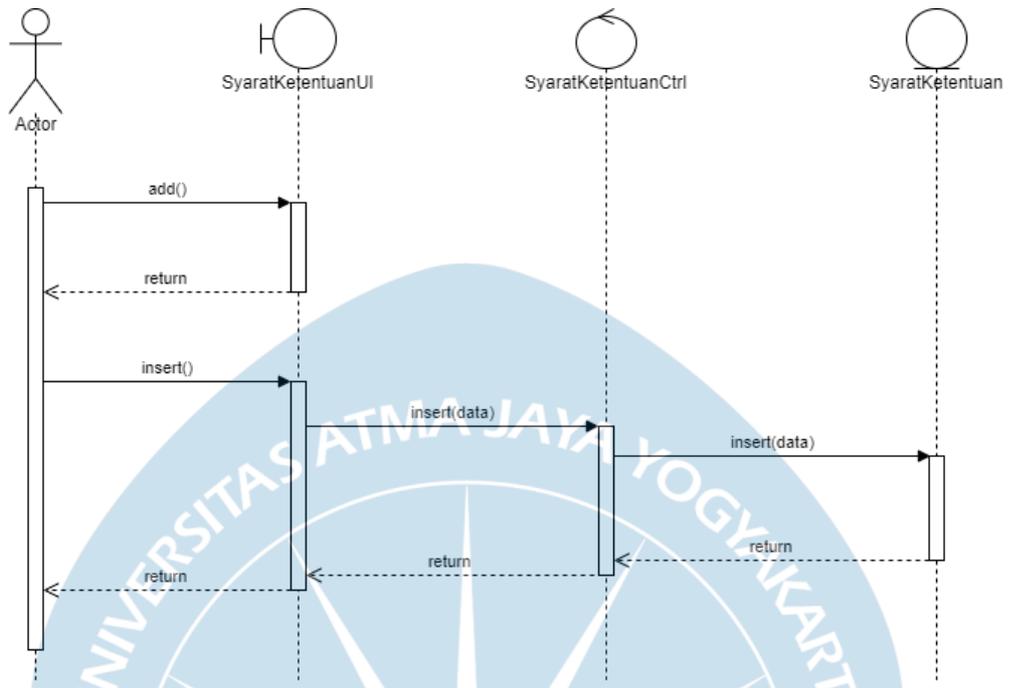
Lampiran 45. Sequence Diagram Display Slider



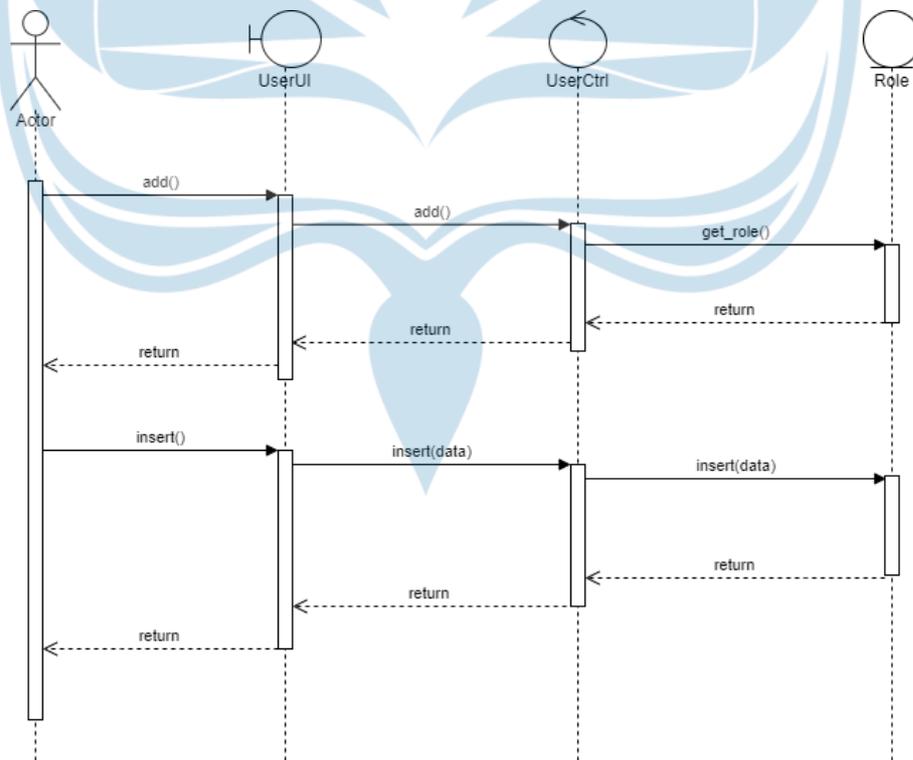
Lampiran 46. Sequence Diagram Display Notification



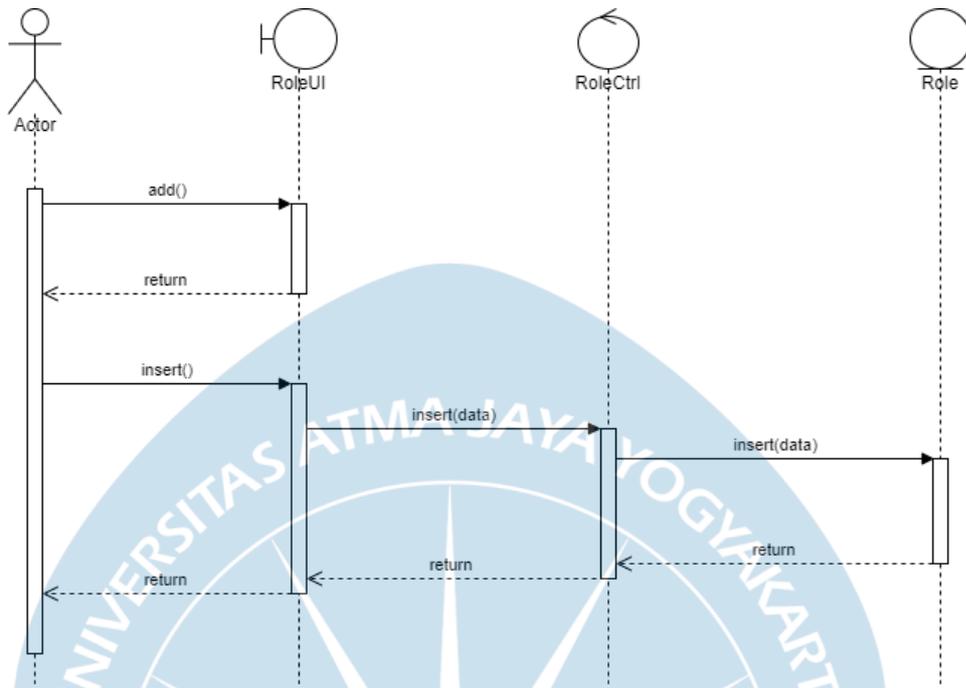
Lampiran 47. Sequence Diagram Display Tracelog Mobile



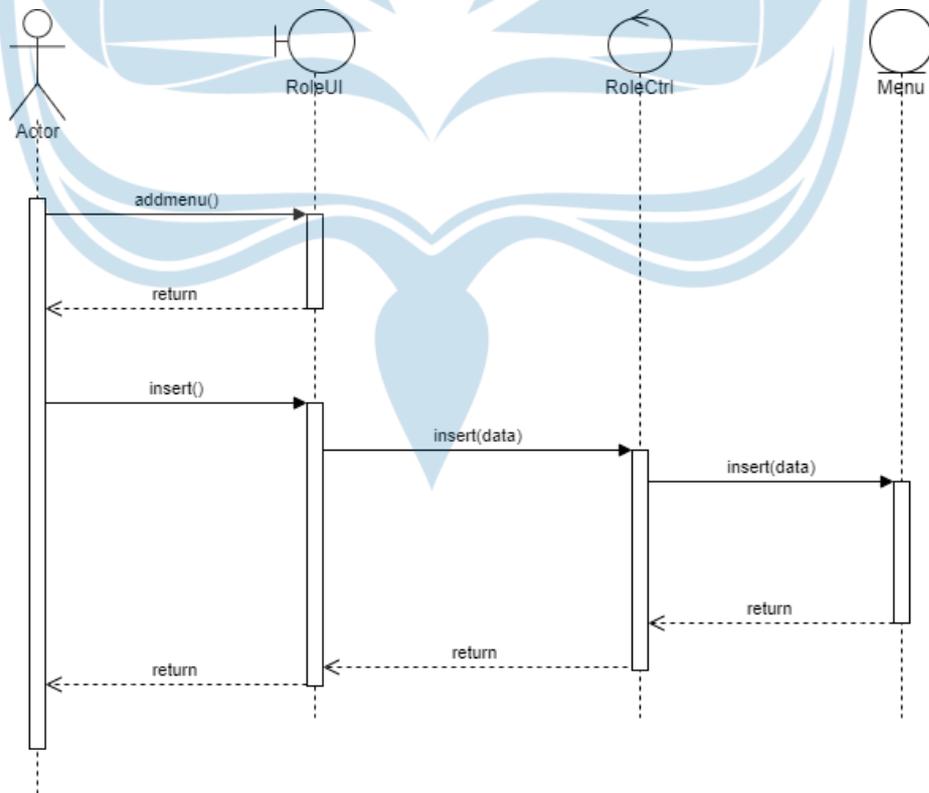
Lampiran 48. Sequence Diagram Add Syarat Ketentuan



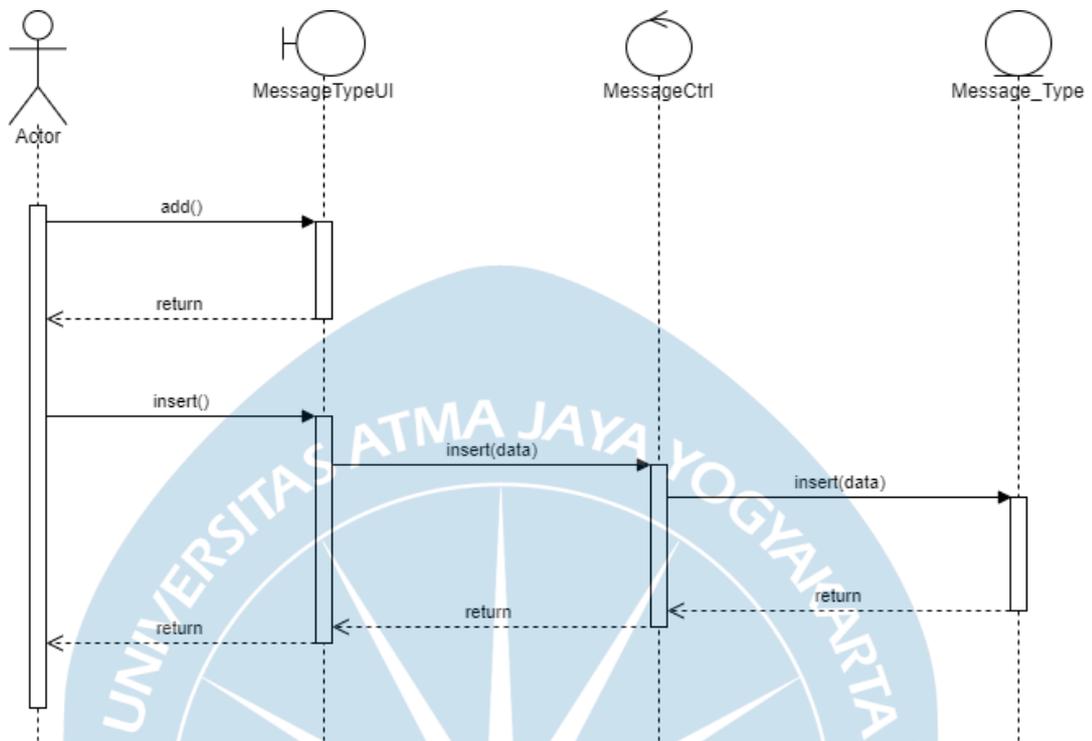
Lampiran 49. Sequence Diagram Add User



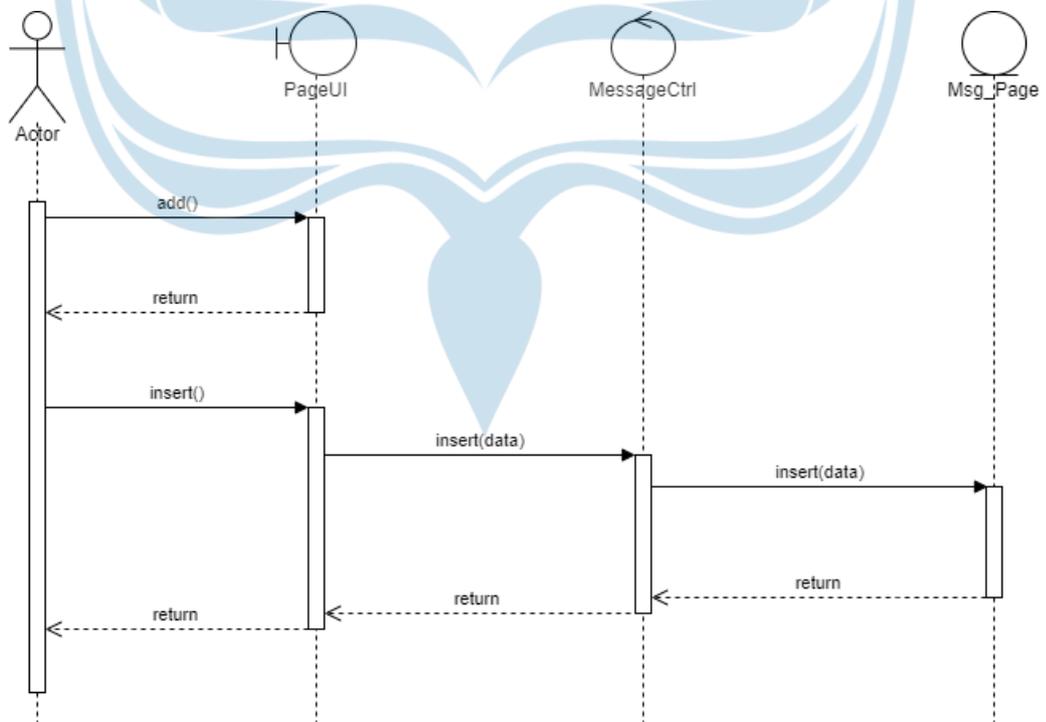
Lampiran 50. Sequence Diagram Add Role



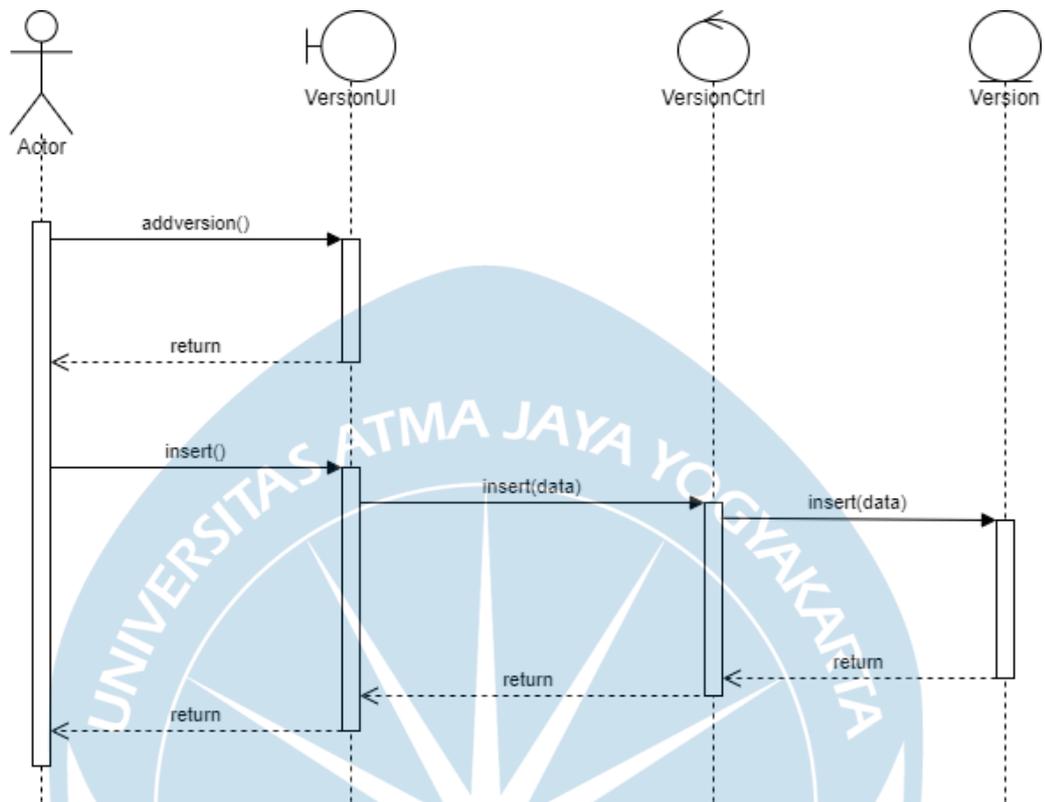
Lampiran 51. Sequence Diagram Add Menu/Submenu



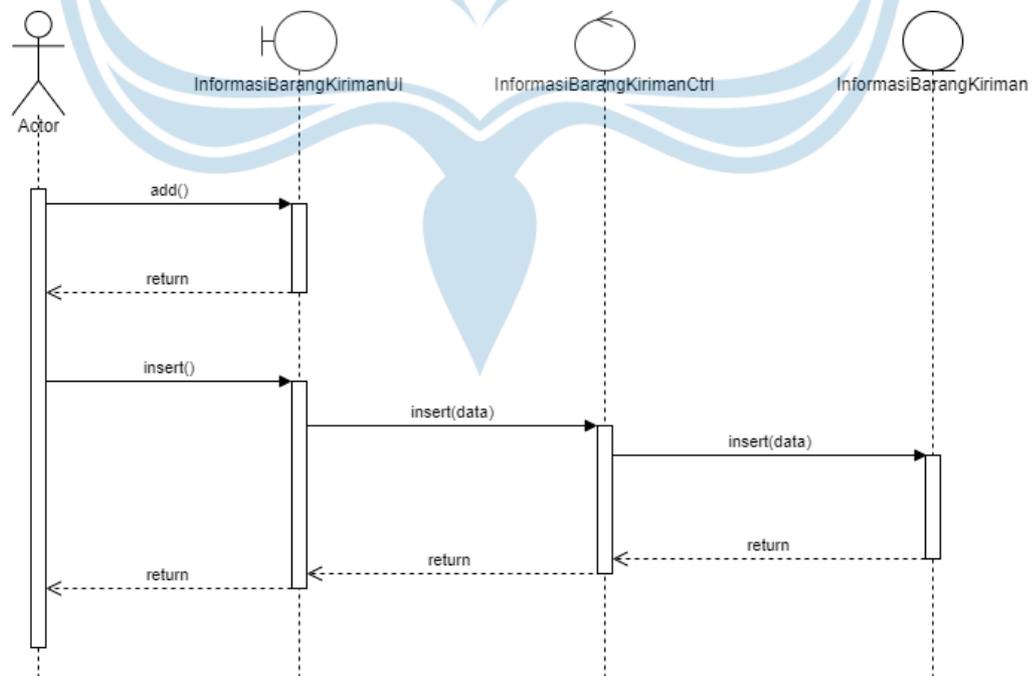
Lampiran 52. Sequence Diagram Add Message Type



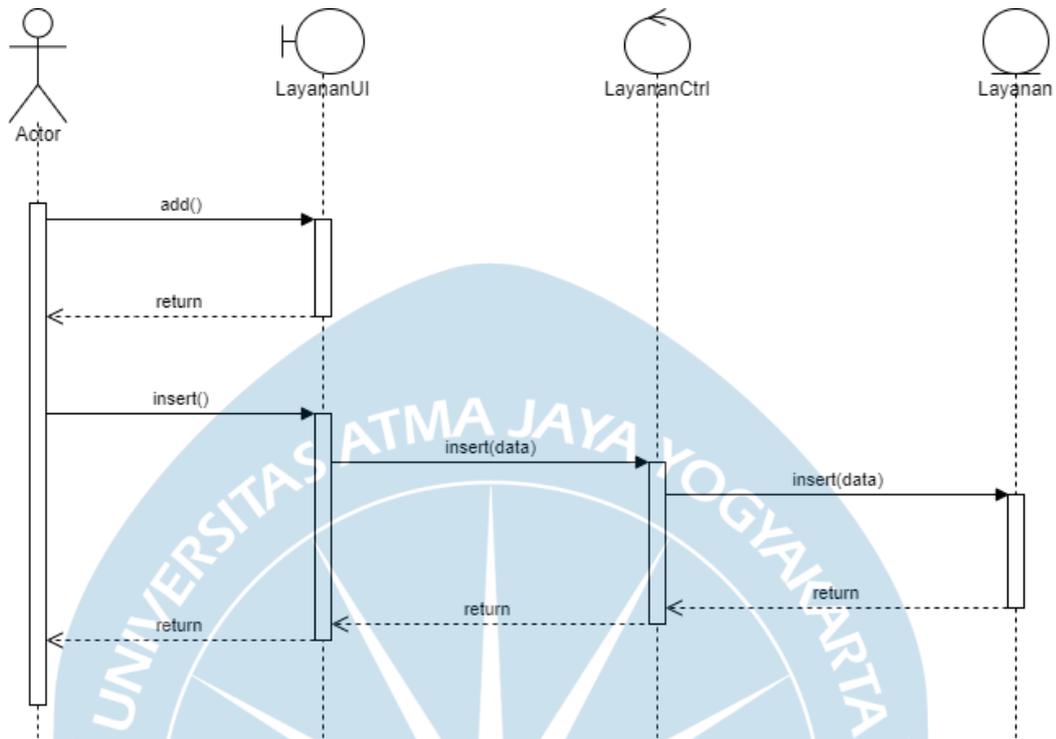
Lampiran 53. Sequence Diagram Add Halaman



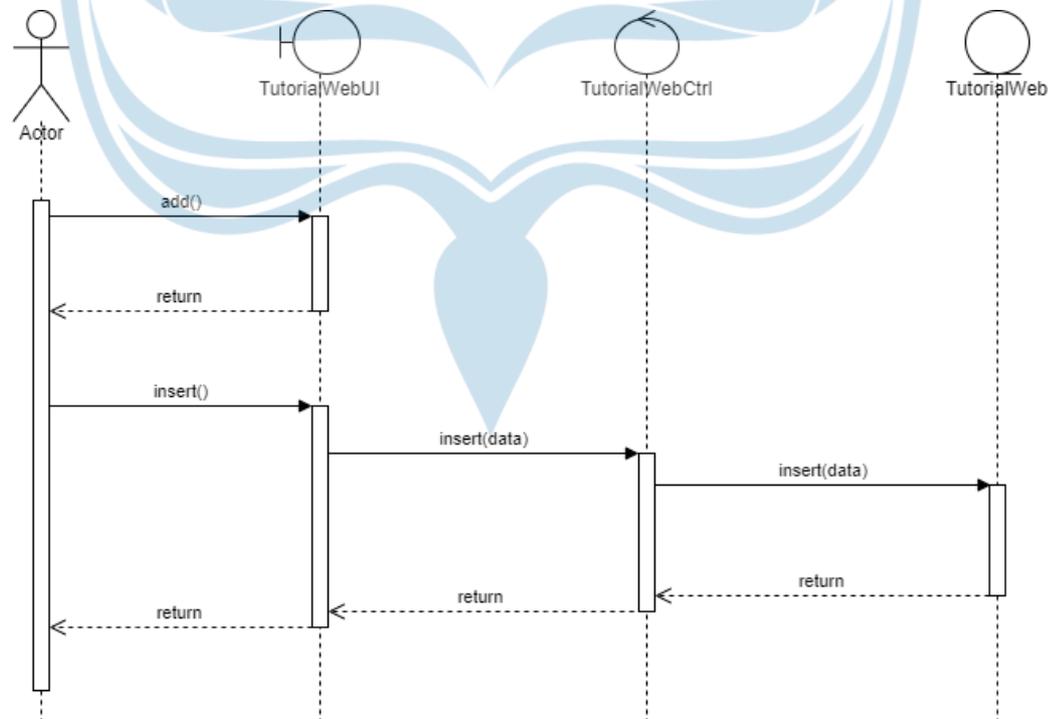
Lampiran 54. Sequence Diagram Add Version



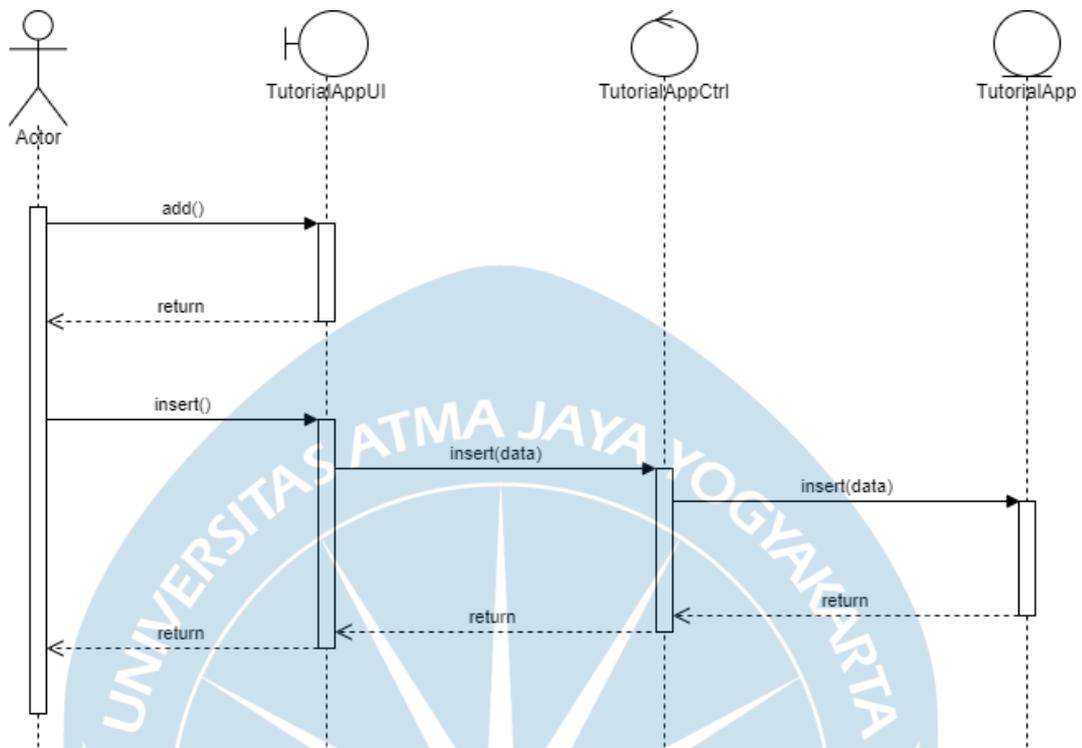
Lampiran 55. Sequence Diagram Add Informasi Barang Kiriman



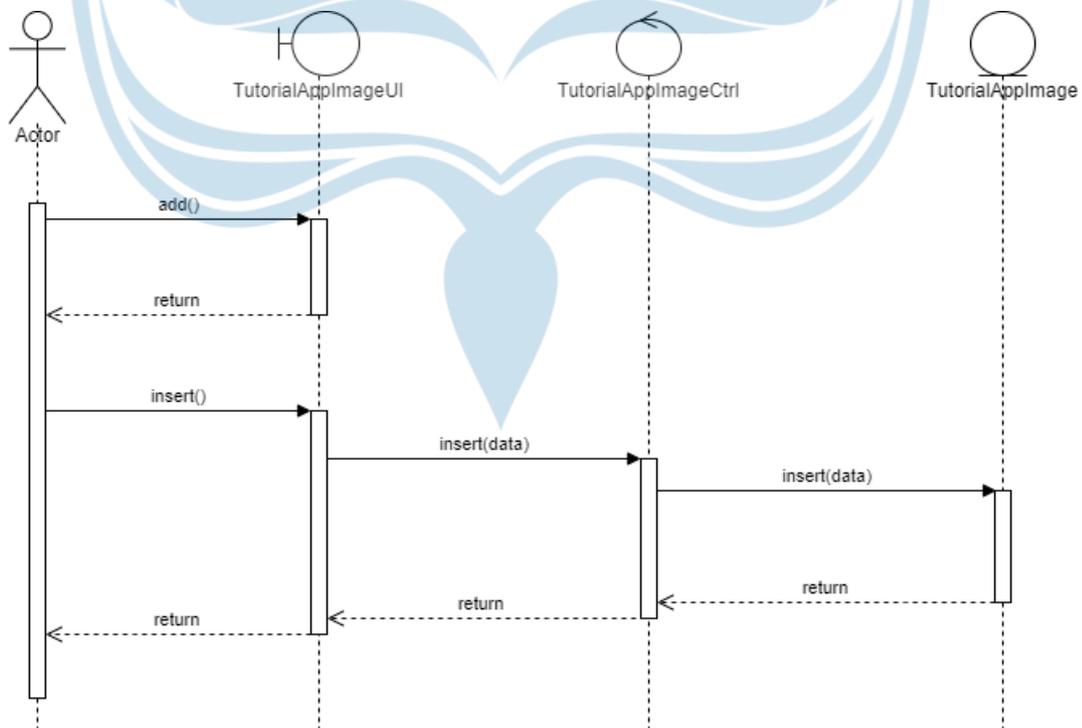
Lampiran 56. Sequence Diagram Add Layanan



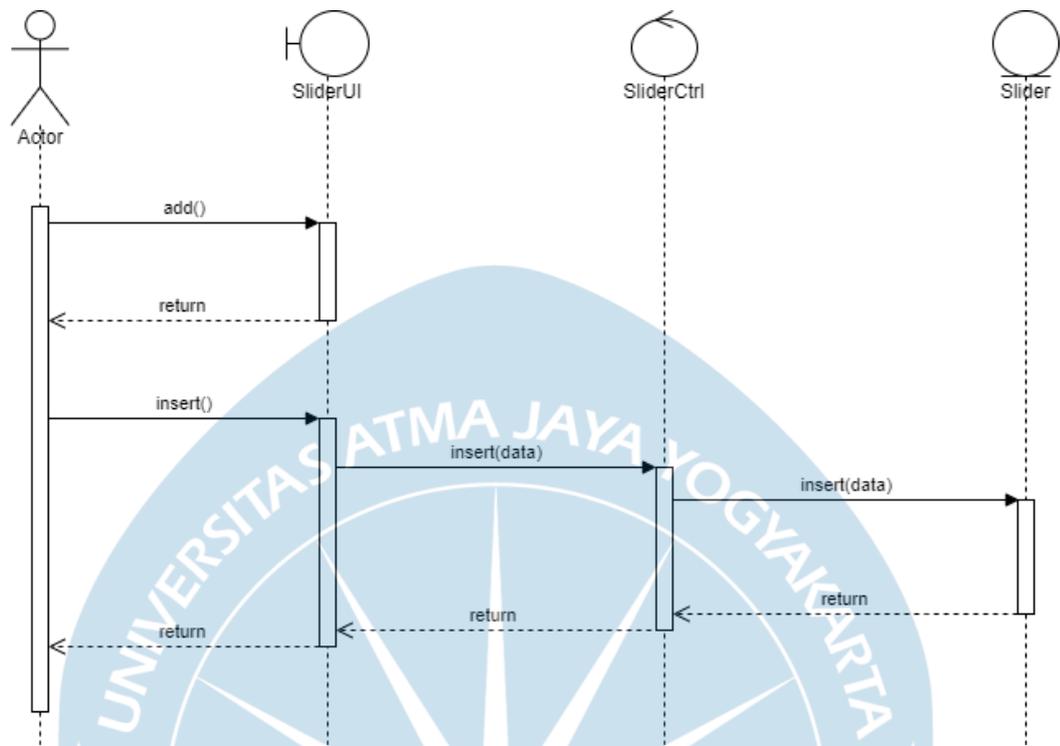
Lampiran 57. Sequence Diagram Add Tutorial Web



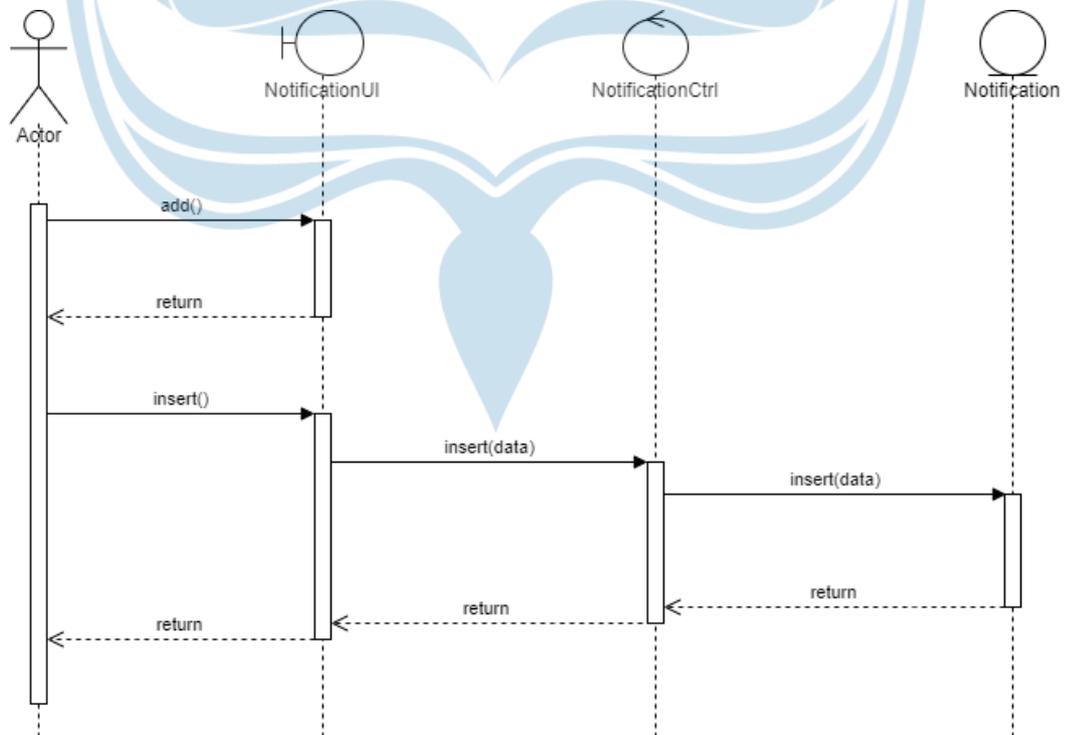
Lampiran 58. Sequence Diagram Add Tutorial App



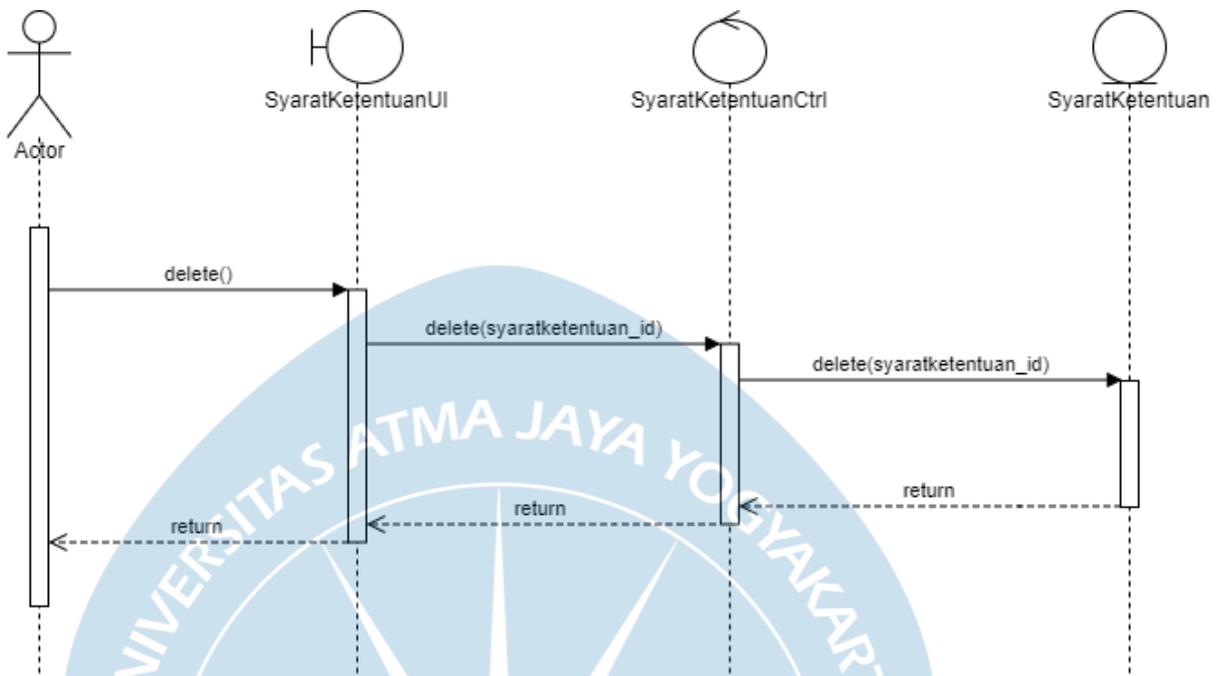
Lampiran 59. Sequence Diagram Add Tutorial App Image



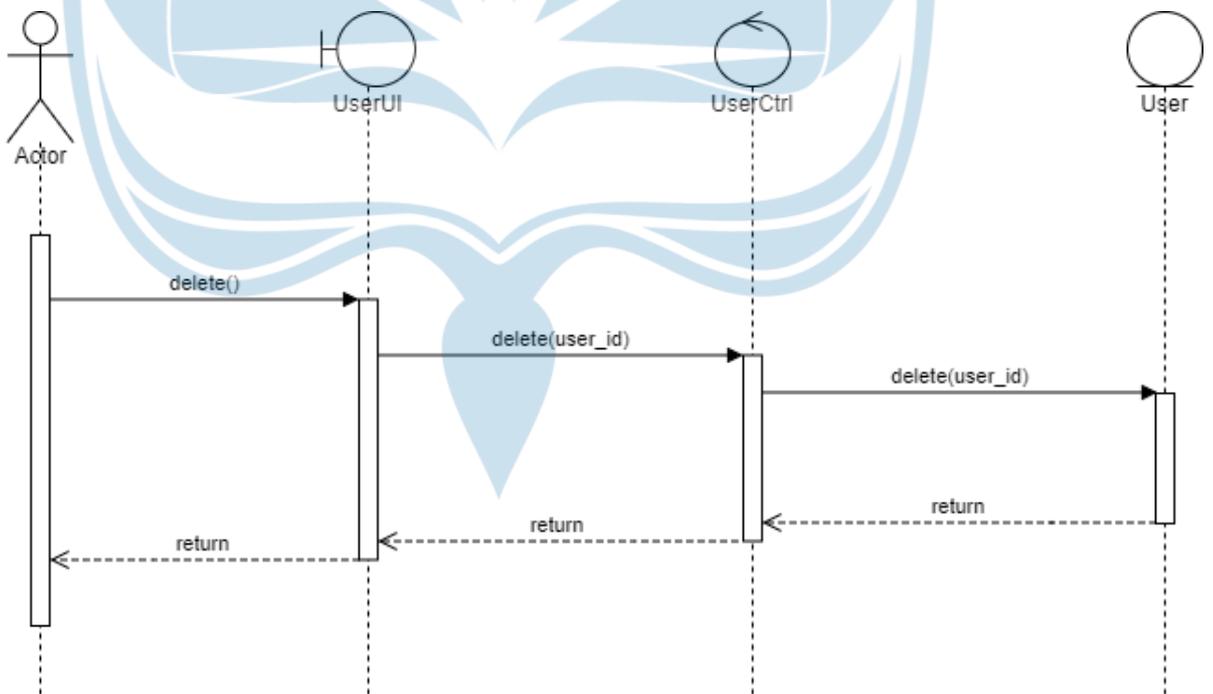
Lampiran 60. Sequence Diagram Add Slider



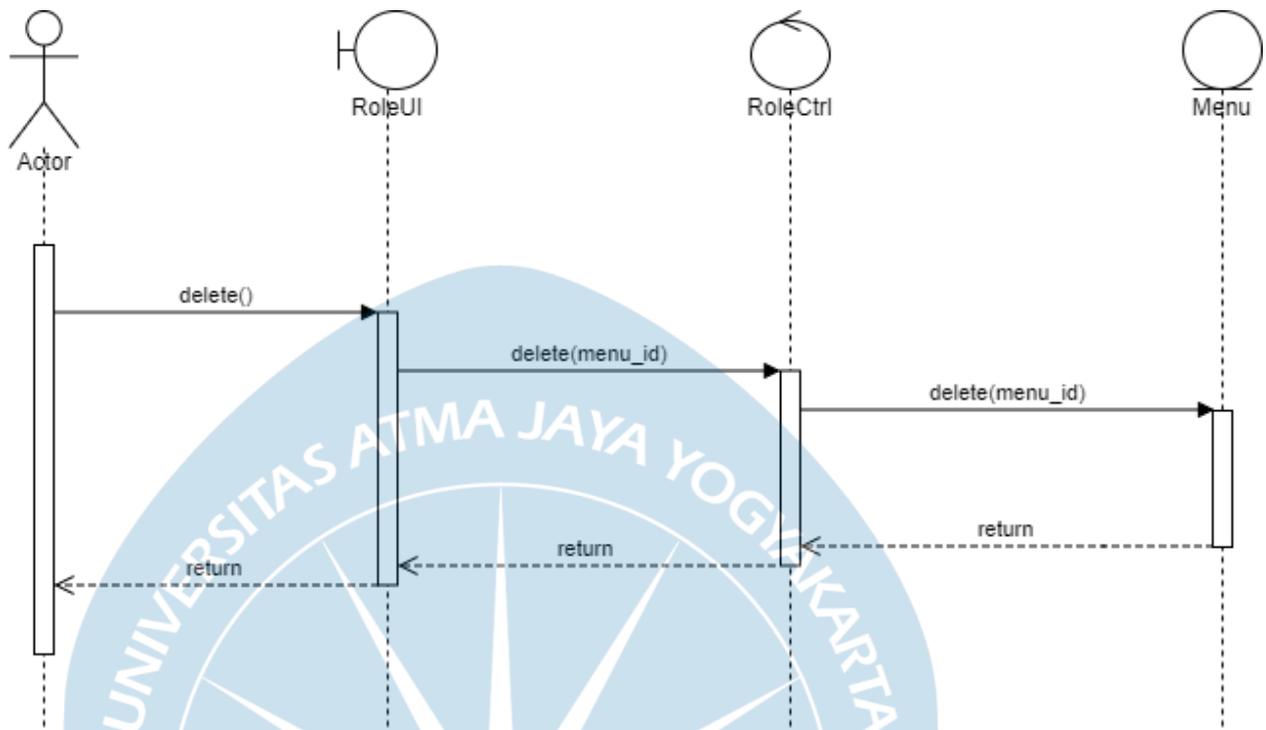
Lampiran 61. Sequence Diagram Add Notification



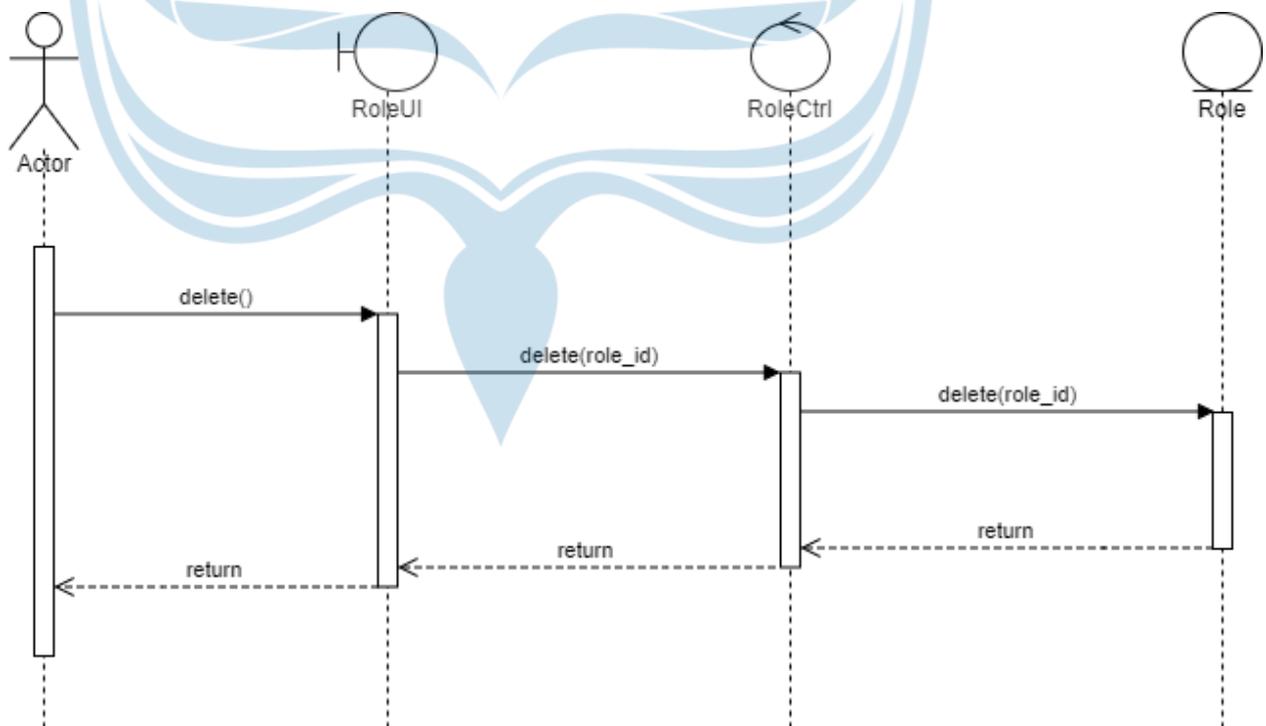
Lampiran 62. Sequence Diagram Delete Syarat Ketentuan



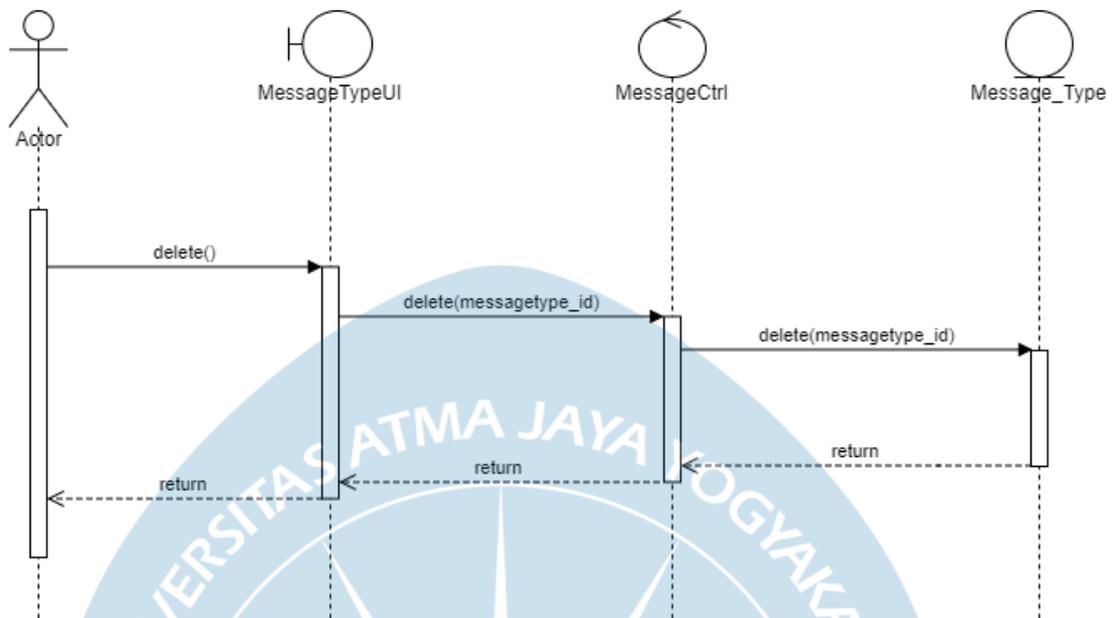
Lampiran 63. Sequence Diagram Delete User



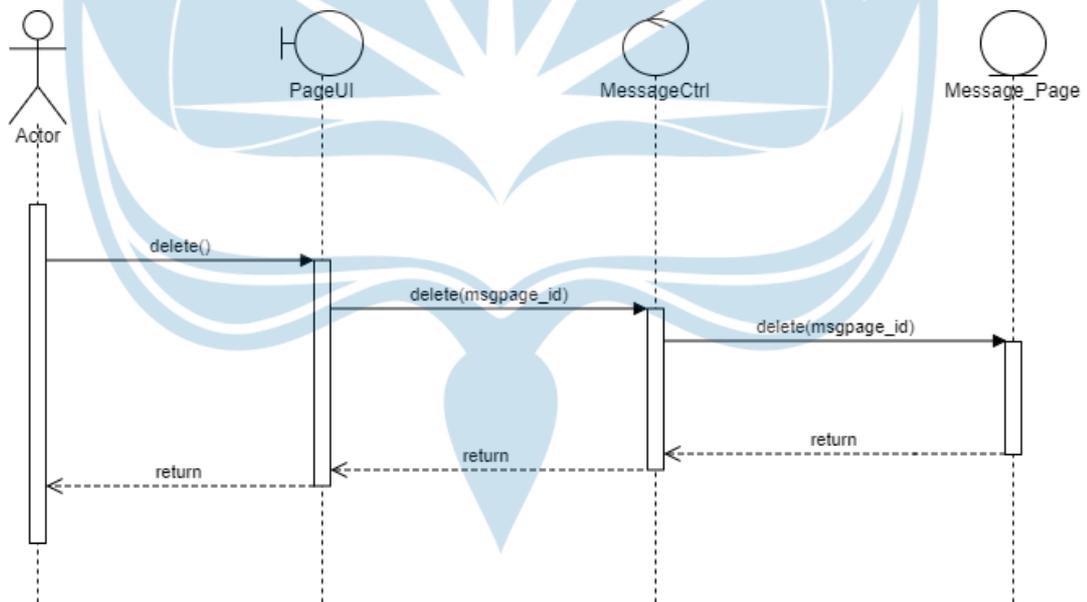
Lampiran 64. Sequence Diagram Delete Menu



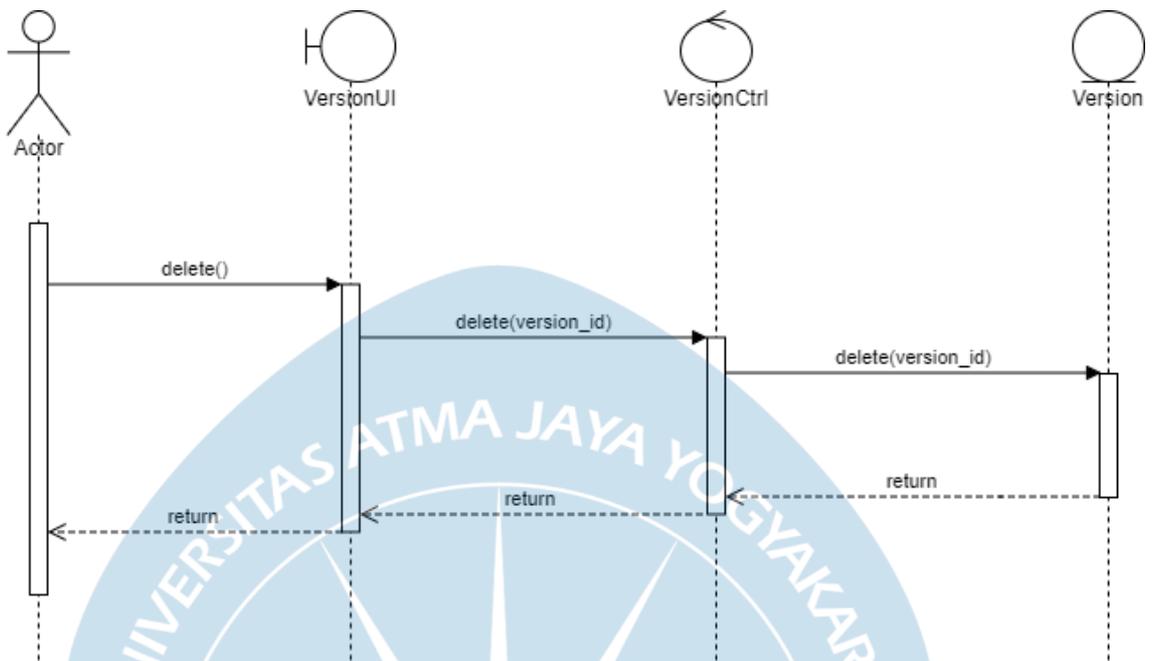
Lampiran 65. Sequence Diagram Delete Role



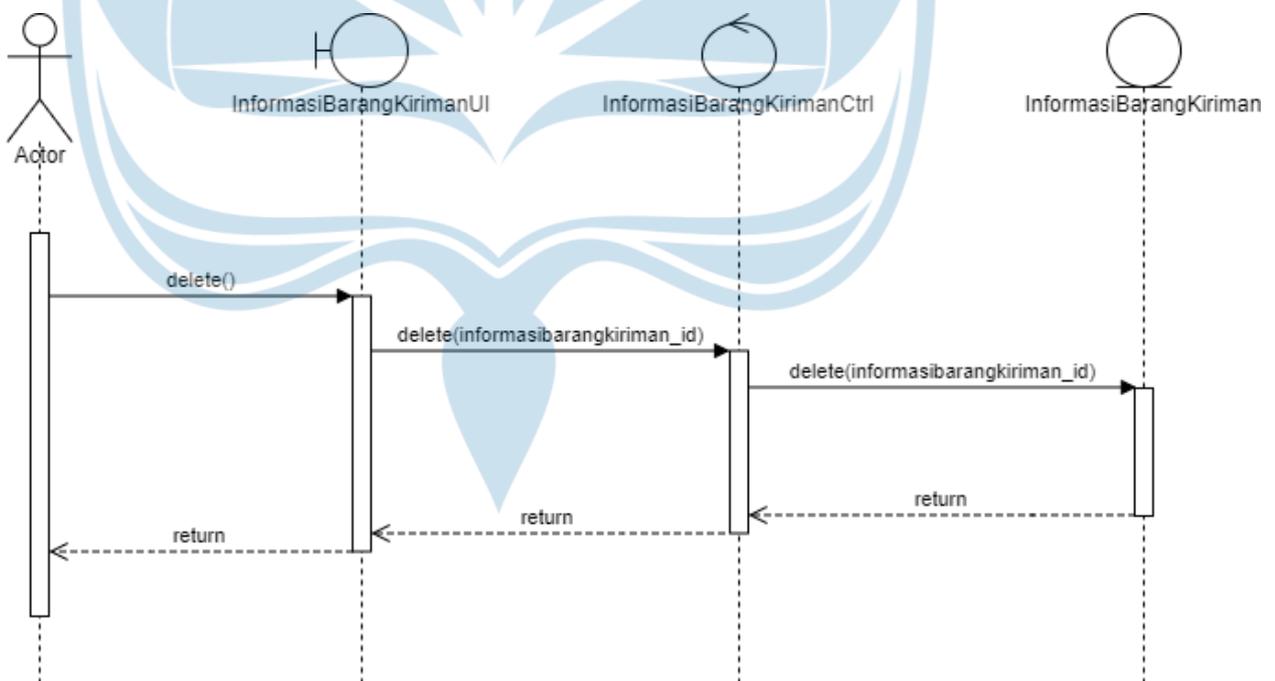
Lampiran 66. Sequence Diagram Delete Message Type



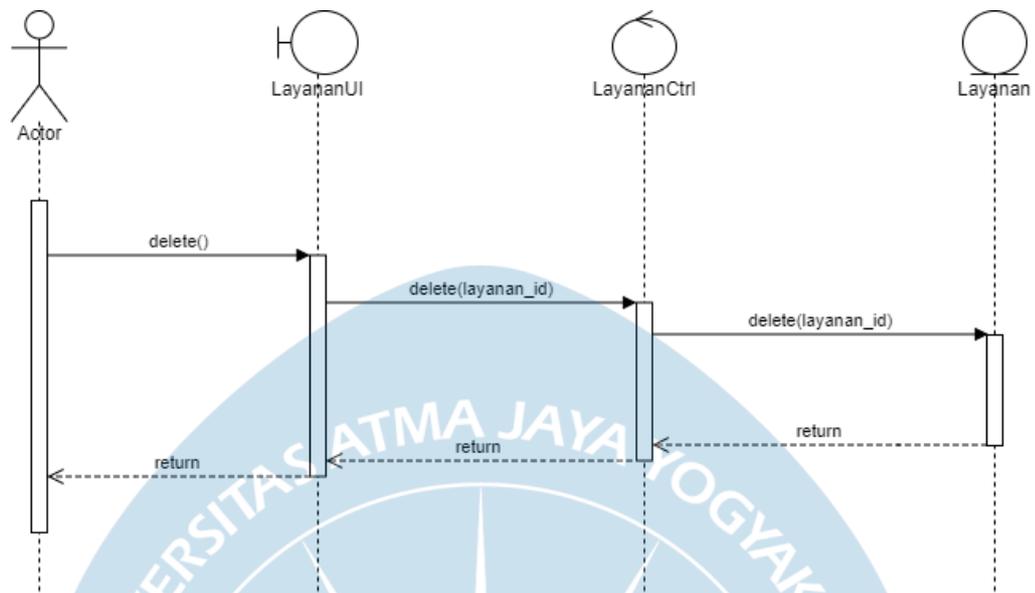
Lampiran 67. Sequence Diagram Delete Halaman



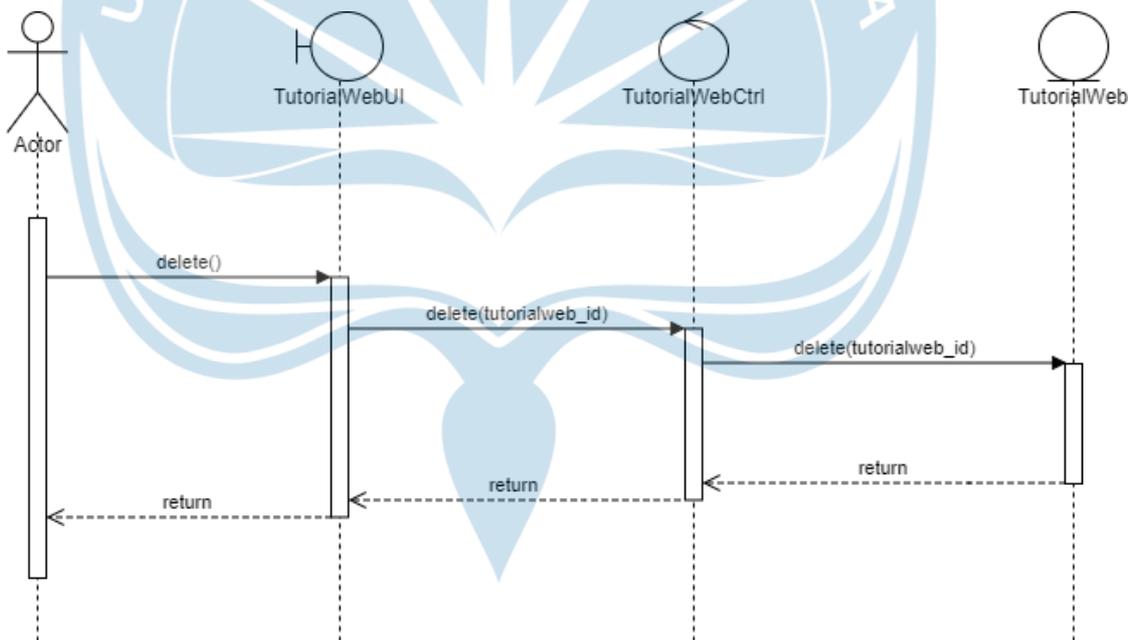
Lampiran 68. Sequence Diagram Delete Version



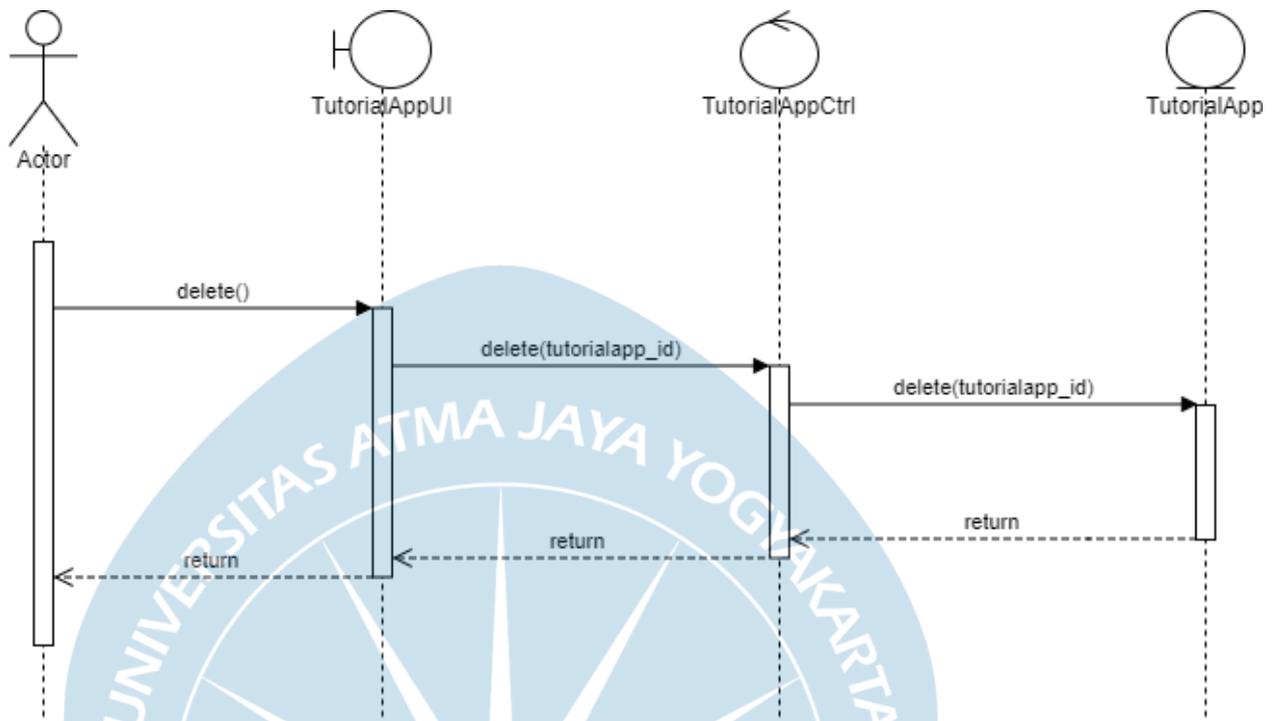
Lampiran 69. Sequence Diagram Delete Informasi Barang Kiriman



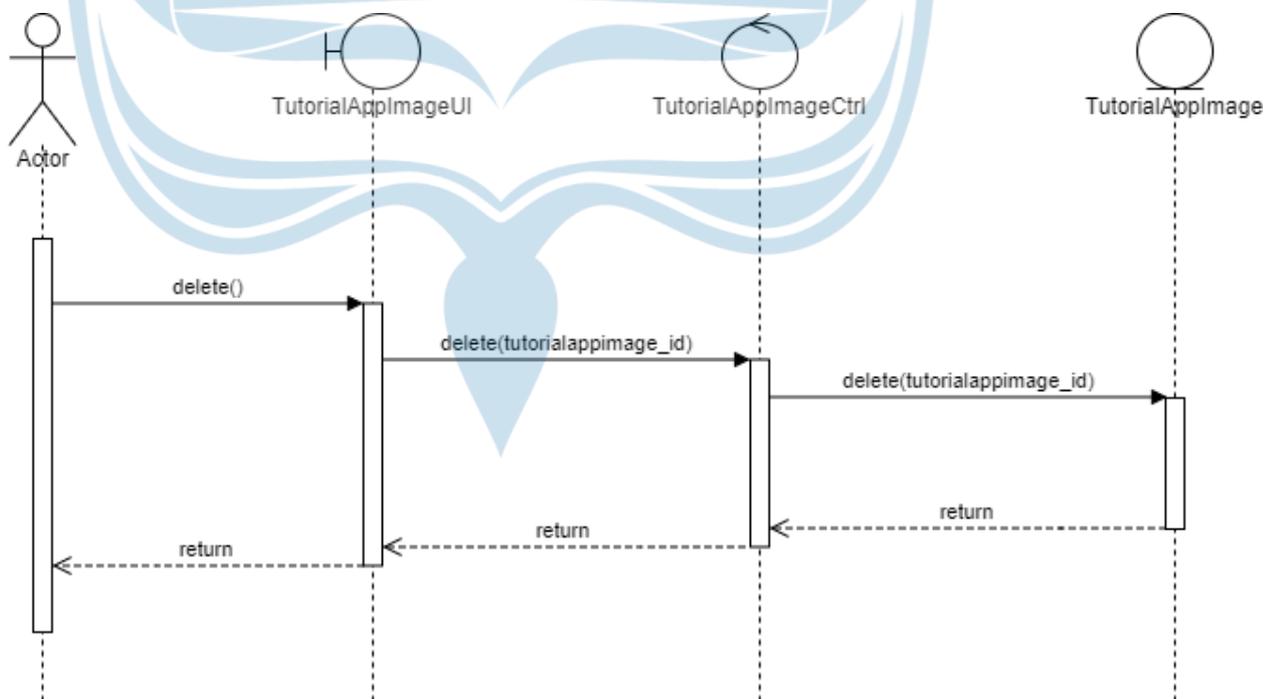
Lampiran 70. Sequence Diagram Delete Layanan



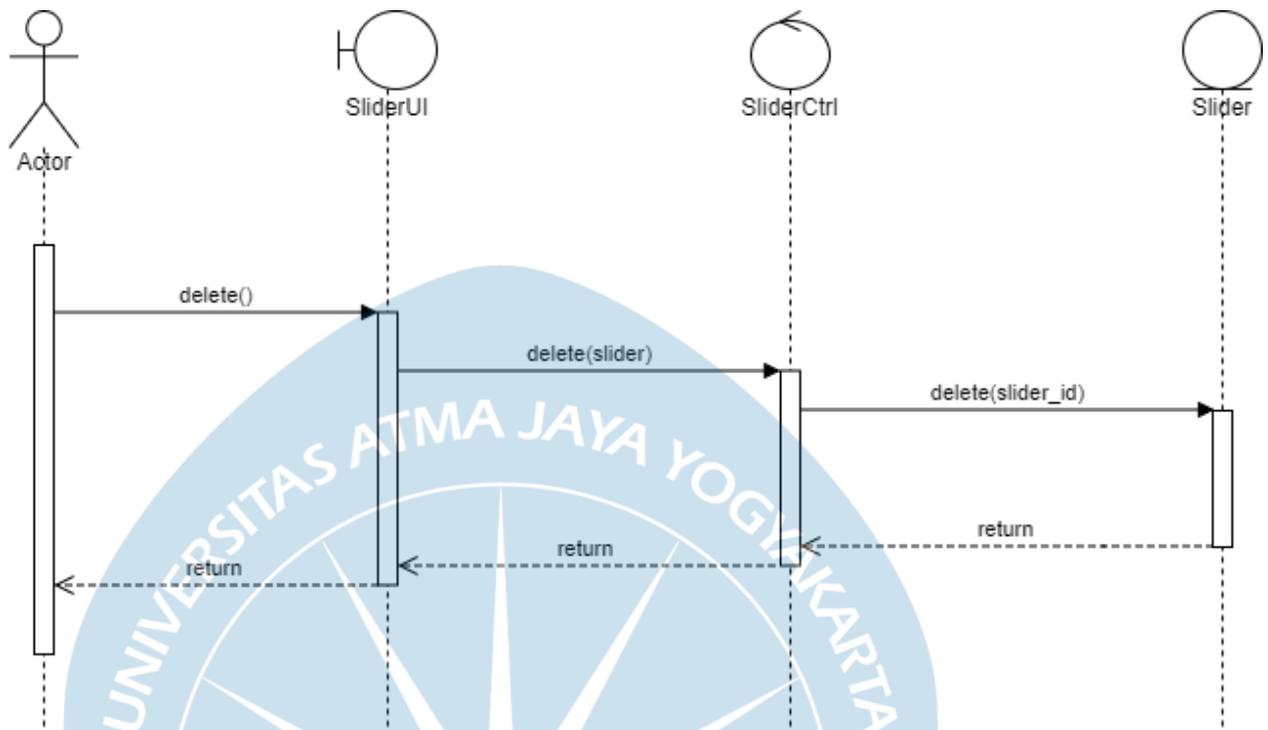
Lampiran 71. Sequence Diagram Delete Tutorial Web



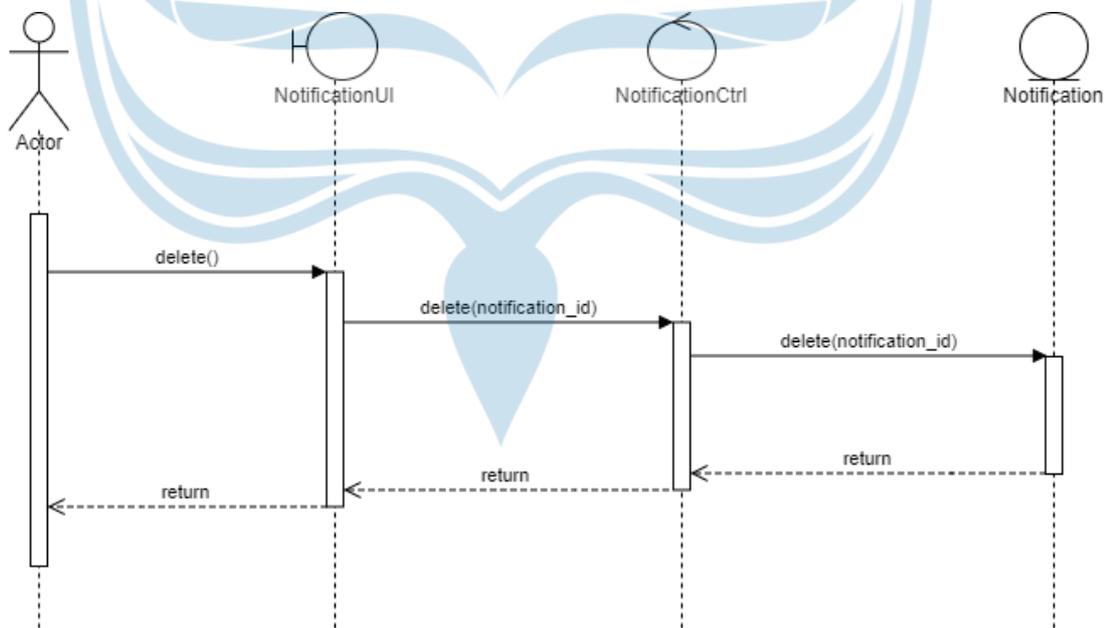
Lampiran 72. Sequence Diagram Delete Tutorial App



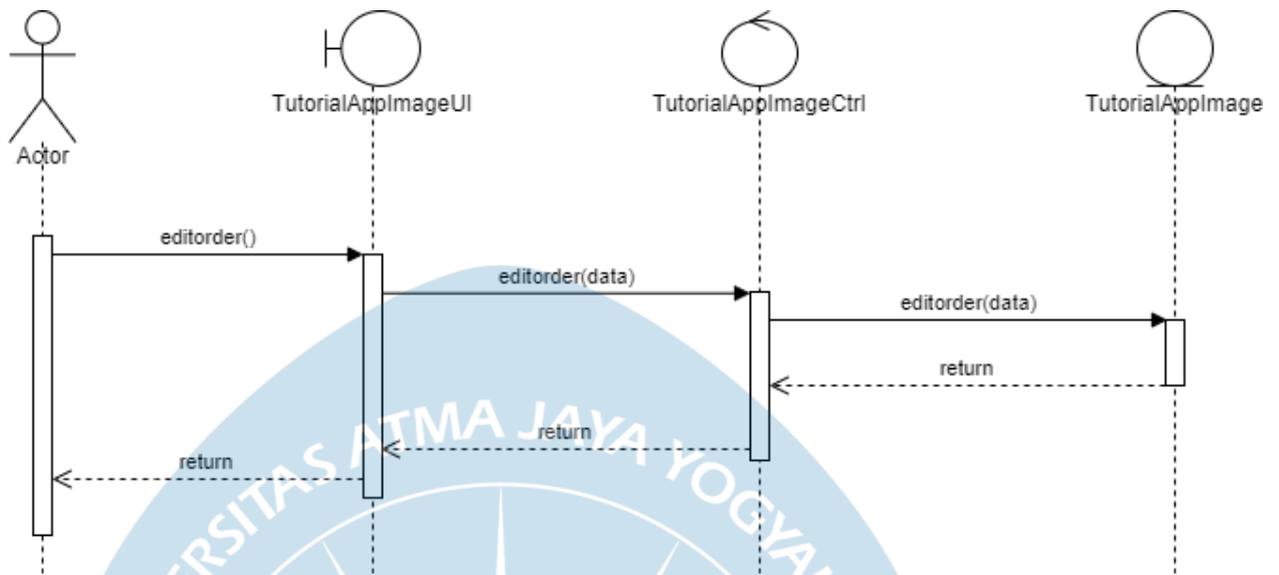
Lampiran 73. Sequence Diagram Delete Tutorial App Image



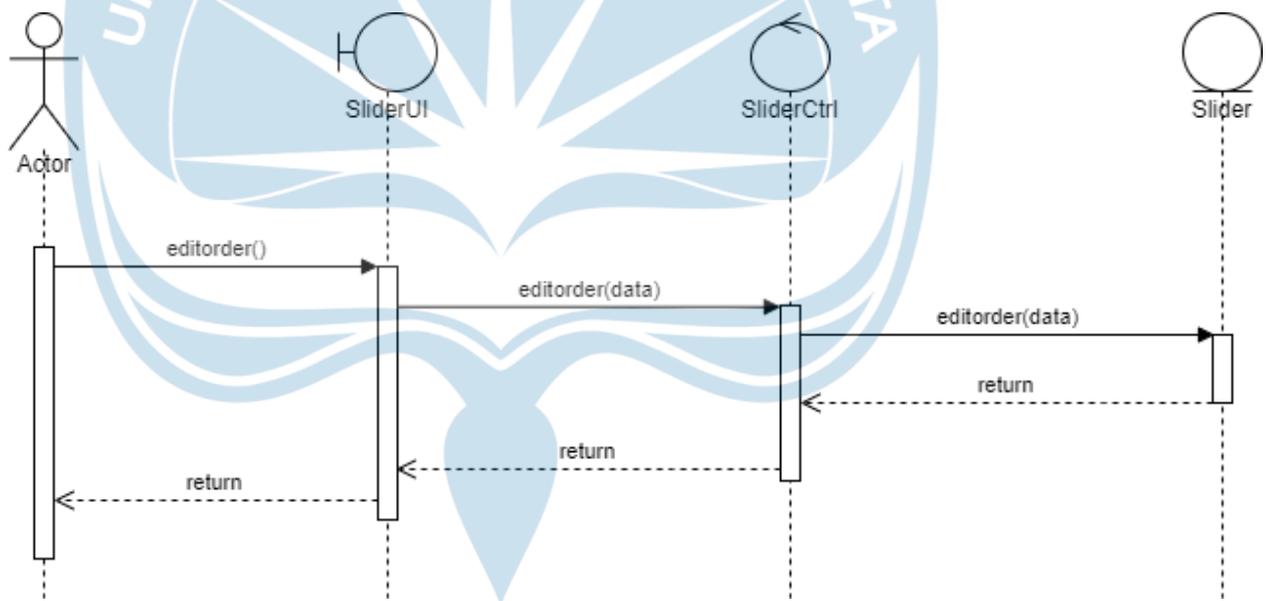
Lampiran 74. Sequence Diagram Delete Slider



Lampiran 75. Sequence Diagram Delete Notification



Lampiran 76. Sequence Diagram Edit Order Tutorial App Image



Lampiran 77. Sequence Diagram Edit Order Slider

Tabel Revisi

No	Tugas Revisi	Halaman Revisi
1.	Perubahan judul.	Mengubah kata “Pembangunan” menjadi “Perancangan”. (Halaman judul)  Mengubah kata “Sistem Manajemen Konten” menjadi “Sistem Pengelolaan Web & App”. (Halaman judul)
2.	Perubahan batasan masalah	Menghilangkan pembangunan dari batasan masalah. (Halaman 5)
3.	Perbaiki metode pengembangan.	Memindahkan teori pengembangan dari Bab 3 ke Bab 2. (Halaman 15-17)
4.	Perubahan citasi.	Mengubah format citasi dari APA menjadi IEEE. (Halaman 147-150)
5.	Penambahan referensi pada gambar kutipan	Menambahkan referensi pada gambar yang dikutip. (Halaman 2)
6.	Perubahan gambar dengan resolusi rendah.	Mengubah gambar SDLC model waterfall dengan gambar baru dengan kualitas lebih tinggi. (Halaman 16)
7.	Penambahan <i>stakeholder</i> dan <i>requirement</i> dari perusahaan.	Menambahkan <i>stakeholder</i> , daftar fungsi sistem lama perusahaan, dan daftar fungsi baru dari <i>request</i> perusahaan sebagai justifikasi perancangan kebutuhan. (Halaman 23)
8.	Penambahan penjelasan	Menambahkan penjelasan untuk masing-

	masing masing gambar.	masing gambar pada <i>use case diagram</i> dan <i>sequence diagram</i> . (Halaman 45-62)
9.	Perbaiki gambar diagram.	Memindahkan gambar <i>use case diagram</i> dan <i>sequence diagram</i> yang mempunyai kemiripan ke lampiran. (Halaman 158-195)
10.	Penambahan bukti alasan pengembangan sistem.	Menambahkan skrip wawancara pada lampiran. (Halaman 156-157)

