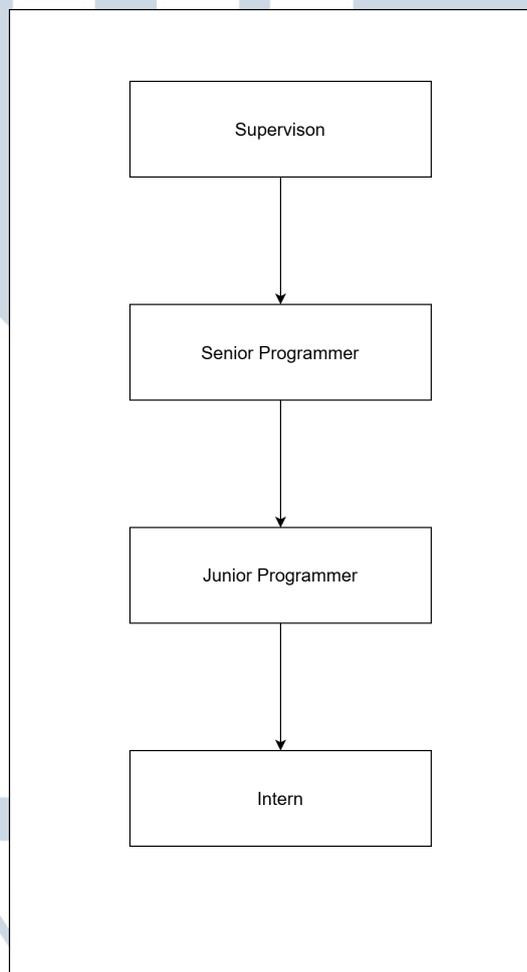


## BAB 3 PELAKSANAAN KERJA MAGANG

### 3.1 Kedudukan dan Koordinasi

*Internship* di PT Graha Karya Informasi sebagai divisi *IT Application* yang ditugaskan untuk membuat aplikasi *GTO System* pada bagian *back end*. Adapun tim ini terdiri dari *supervisor*, *senior programmer*, *junior programmer* dan *intern*.



Gambar 3.1. Divisi *IT Application*

Tim ini dipimpin oleh Bapak Thamiya Suherman selaku *supervisor* tim *IT Application*. Bapak Thamiya Suherman juga bertanggung jawab atas 3 orang dibawahnya. Selama praktik kerja magang, penugasan diberikan dan dipandu oleh Bapak Thamiya Suherman. Penugasan selalu diatur dengan *platform*

terdedikasi yaitu *Trello* terkait obyektif yang ingin dicapai pada penugasan tersebut. Dokumentasi *codebase* ataupun proses penugasan dilakukan melalui *platform Github*. Proyek ini selalu dikomunikasikan dan dikerjakan bersama *supervisor* dan *junior programmer* sehingga proyek yang dikerjakan selalu *direview* tiap minggu secara tatap muka dan melalui *Github*.

### 3.2 Tugas yang Dilakukan

Tugas yang dilakukan pada pelaksanaan kerja magang adalah sebagai berikut dapat dilihat pada tabel 3.1

Tabel 3.1. Tugas yang dilakukan selama pelaksanaan kerja magang

Minggu ke-	Pekerjaan yang dilakukan
1	Perkenalan dengan <i>supervisor</i> dan <i>mentor</i> serta latihan instalasi <i>library react</i> dan basik <i>javascript</i> .
2	<i>Meeting</i> latihan membuat tampilan <i>front end Skincare</i> dengan <i>react javascript</i> dan menghubungkannya ke <i>Github</i> .
3	Melanjutkan membuat tampilan <i>website Skincare</i> dan mengilas belajar dasar <i>javascript callback, asyn, dan promises</i> serta <i>meeting</i> untuk belajar dasar <i>javascript</i> .
4	Belajar tentang <i>UseState (ReactJS)</i> dan latihan menyelesaikan kasus sederhana <i>UseState</i> yaitu membuat kalkulator sederhana.
5	Latihan <i>MySQL</i> untuk memulai mengerjakan proyek <i>GTO System</i> .
6	membuat <i>database</i> menggunakan <i>tools SQLyog</i> dan belajar membuat <i>API</i> dengan <i>Query</i> dan mengisi data tersebut dengan data <i>dummy</i>

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

Minggu ke-	Pekerjaan yang dilakukan
7	Membuat <i>API</i> dengan nama <i>folder</i> yang berisi masing-masing <i>file course, trainer, dan participant</i> dengan menggunakan <i>function getHandler</i> yang berisi <i>Query</i> dan ditampilkan ke dalam masing-masing <i>page front end</i> .
8	Membuat <i>API</i> dengan <i>folder</i> yang berisi <i>file operation</i> dengan menggunakan <i>function getHandler</i> yang berisi <i>Query</i> dan ditampilkan ke dalam masing-masing <i>front end</i> .
9	Membuat komponen <i>form</i> dan menampilkan <i>form</i> tersebut dengan nama <i>folder</i> yang berisi masing-masing <i>file course, trainer, dan participant</i> . Komponen ini berfungsi untuk menginput data setiap <i>course, trainer, dan participant</i> .
10	Membuat komponen <i>form</i> dan menampilkan <i>form</i> tersebut dengan nama <i>folder</i> yang berisi <i>file operation</i> . Komponen ini berfungsi untuk mengisi data <i>operation</i> .
11	Membuat <i>function postHandler</i> pada <i>API course, trainer, dan participant</i> untuk menjalankan komponen <i>form</i> yang sudah dibuat dalam <i>folder</i> komponen <i>course, trainer, dan participant</i> .
12	Membuat <i>function postHandler</i> pada <i>API operation</i> untuk menjalankan komponen <i>form</i> yang sudah dibuat dalam <i>folder</i> komponen <i>operation</i> .
13	Membuat <i>function putHandler</i> pada <i>API course, trainer, dan participant</i> supaya dapat mengedit dan menghapus setiap data yang sudah dicantumkan. Kemudian membuat fitur <i>edit</i> untuk memperbaiki data dan <i>delete</i> untuk menghapus data yang sudah tidak diperlukan.
14	Membuat <i>function putHandler</i> pada <i>API operation</i> supaya dapat mengedit dan menghapus setiap data yang sudah dicantumkan. Kemudian membuat fitur <i>edit</i> untuk memperbaiki data dan <i>delete</i> untuk menghapus data yang sudah tidak diperlukan.

- Minggu pertama saat melakukan *internship* adalah perkenalan dengan *supervisor, mentor, dan* latihan instalasi *library react* dan dasar *javascript*.
- Pada minggu kedua *meeting* dilakukan untuk latihan membuat tampilan *front end Skincare* dengan *react javascript* dan menghubungkannya ke *Github*.

- Minggu ketiga, melanjutkan membuat tampilan *website Skincare* dan mengilias belajar dasar *javascript callback, asyn, dan promises* serta meeting untuk belajar dasar *javascript*.
- Minggu keempat, belajar tentang *UseState (ReactJS)* dan latihan menyelesaikan kasus sederhana, yaitu membuat kalkulator sederhana.
- Minggu kelima, latihan *MySQL* untuk memulai mengerjakan proyek *GTO System*.
- Minggu Keenam, membuat *database* menggunakan *tools SQLyog* dan belajar membuat *API* dengan *Query* dan mengisi data tersebut dengan data *dummy*.
- Minggu ketujuh, membuat *API* dengan nama folder yang berisi masing-masing *file course, trainer, dan participant* dengan menggunakan *function getHandler* yang berisi *Query* dan ditampilkan de dalam masing-masing *page front end*.
- Minggu kedelapan, membuat *API* dengan nama folder yang berisi *file operation* dengan menggunakan *function getHandler* yang berisi *Query* dan ditampilkan de dalam masing-masing *page front end*.
- Minggu kesembilan, membuat komponen *form* dan menampilkan *form* tersebut dengan nama *folder* yang berisi masing-masing *file course, trainer, dan participant*. Komponen ini berfungsi untuk mengisi data setiap *course, trainer, dan participant*.
- Minggu kesepuluh, membuat komponen *form* dan menampilkan *form* tersebut dengan nama *folder* yang berisi *file operation*. Komponen ini berfungsi untuk mengisi data setiap *operation*.
- Minggu kesebelas, membuat *function postHandler* pada *API course, trainer, dan participant* untuk menjalankan komponen *form* yang sudah dibuat dalam folder komponen *course, trainer, dan participant*.
- Minggu keduabelas, membuat *function postHandler* pada *API operation* untuk menjalankan komponen *form* yang sudah dibuat dalam *folder* komponen *operation*.
- Minggu ketigabelas, membuat *function putHandler* pada *API course, trainer, dan participant* supaya dapat mengedit dan menghapus setiap data yang sudah

dicantumkan. Kemudian membuat fitur *edit* untuk memperbarui data dan *delete* untuk menghapus data yang sudah tidak diperlukan.

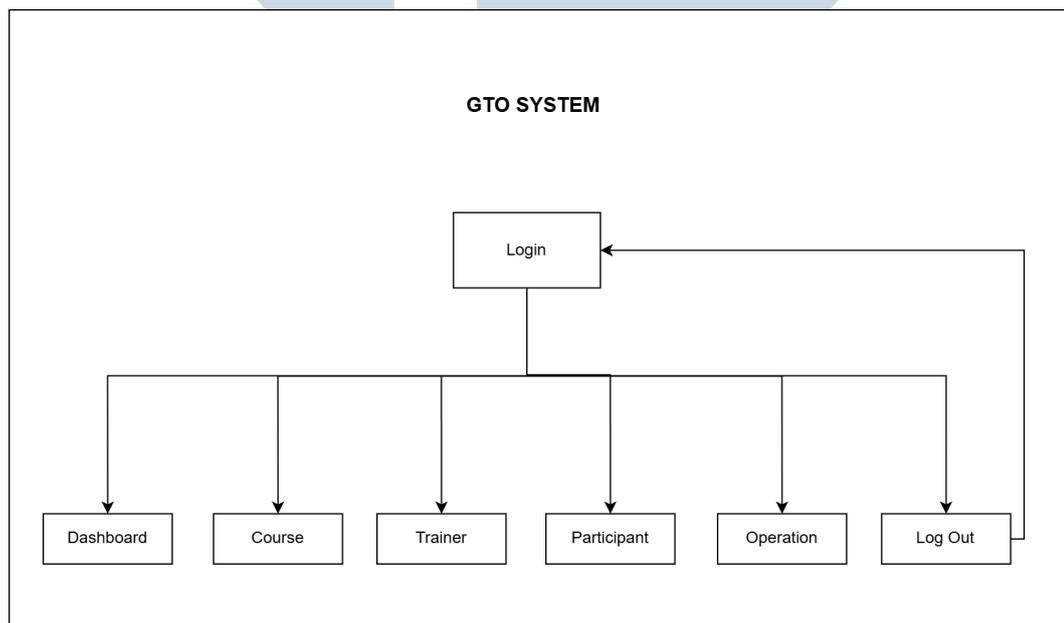
- Minggu keempatbelas, membuat *function putHandler* pada *API operation* supaya dapat mengedit dan menghapus setiap data yang sudah dicantumkan. Kemudian membuat fitur *edit* untuk memperbarui data dan *delete* untuk menghapus data yang sudah tidak diperlukan.

### 3.3 Uraian Pelaksanaan Magang

#### 3.3.1 Perancangan Sistem

##### A. Sitemap

*Sitemap* dibuat berdasarkan gambaran hubungan pada aplikasi *GTO System* yang sudah dirancang.

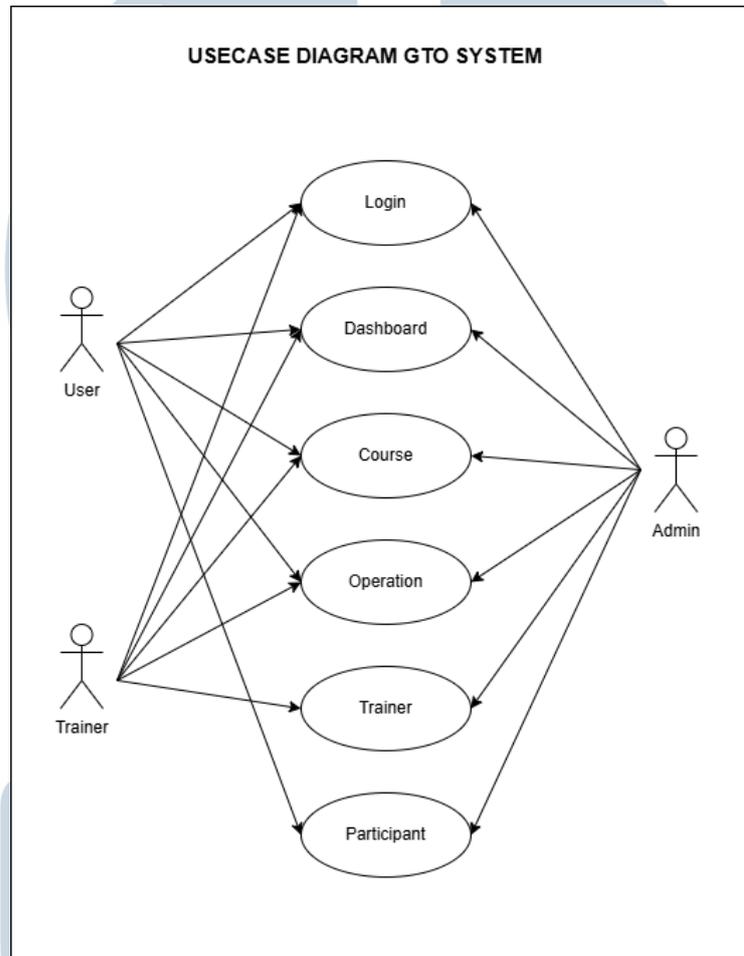


Gambar 3.2. Sitemap

Gambar 3.2 di atas menggambarkan fungsionalitas dan interaksi penggunaan aplikasi *GTO System*.

## B. Usecase Diagram

Aplikasi *GTO System* ini memiliki *Usecase* diagram supaya mempermudah dan memberikan pemahaman dalam penggunaannya.

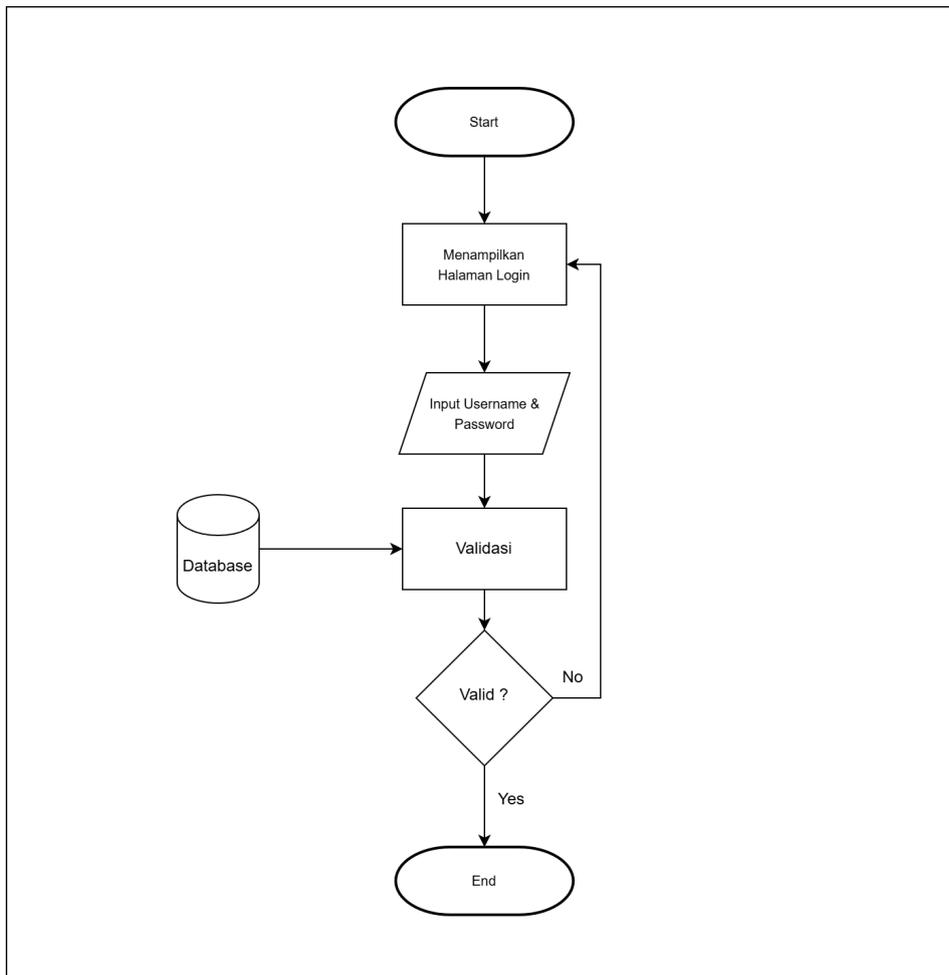


Gambar 3.3. *Usecase* Diagram *GTO System*

## C. Flowchart

Aplikasi *GTO System* ini memiliki *flowchart* per bagian dalam melakukan setiap sistemnya. Terdapat sistem *CRUD* dalam setiap halaman *website* kecuali halaman *Login* dan *Dashboard*.

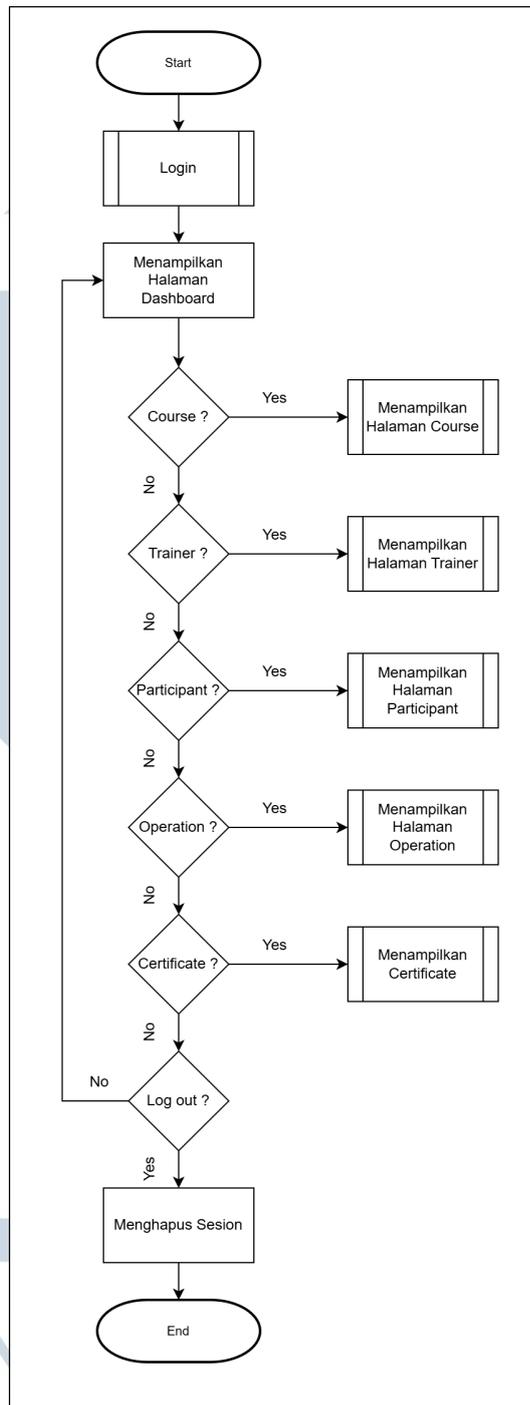
### 1. Flowchart Login



Gambar 3.4. Flowchart Login

Gambar 3.3 di atas merupakan flowchart *Login* yang digunakan sebagai pintu masuk ke dalam aplikasi *GTO System* dengan memasukkan *username* dan *password*. Setelah selesai melakukan *login*, maka tampilan akan berpindah ke laman *Dashboard*. Jika tidak melakukan login maka laman akan tetap pada halaman *login*.

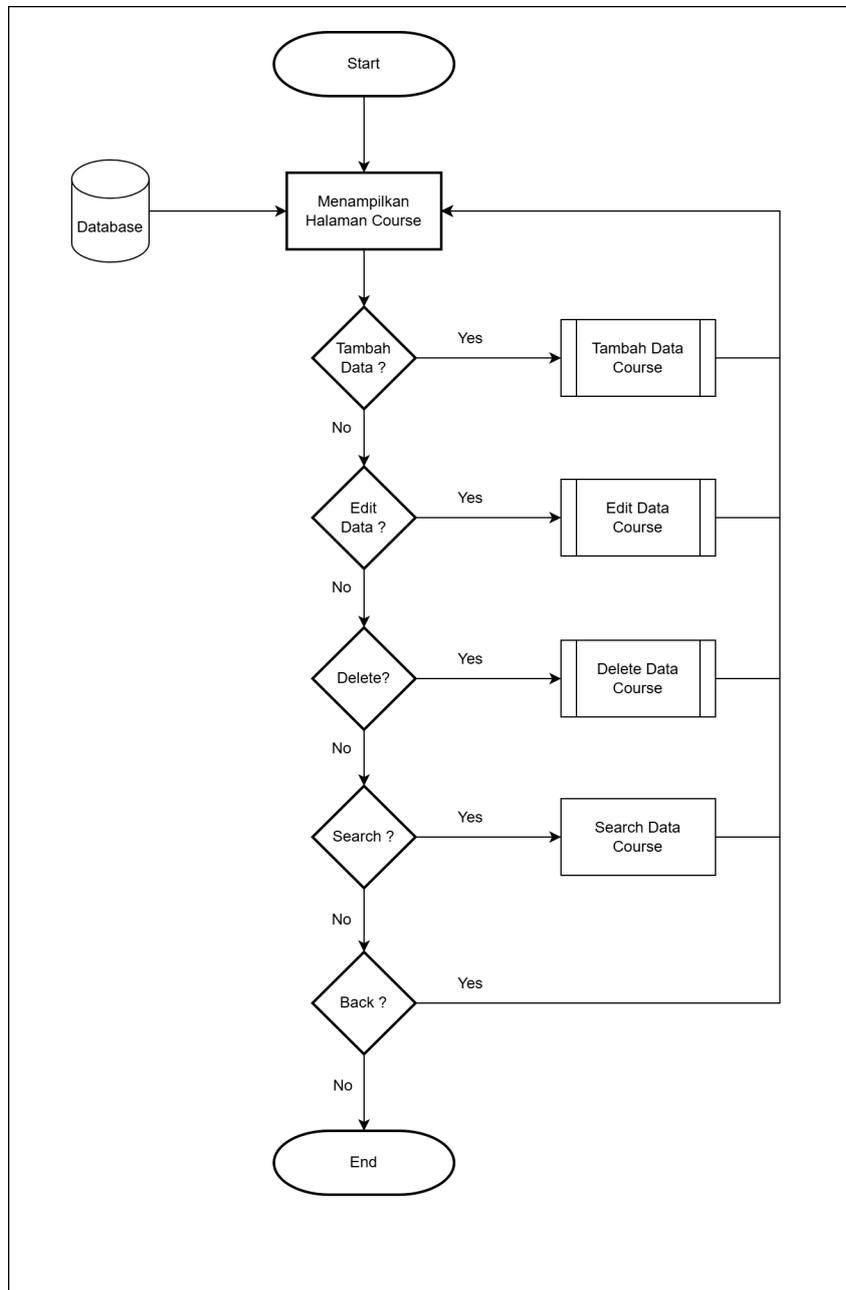
## 2. Flowchart Dashboard



Gambar 3.5. Flowchart Halaman Dashboard

Gambar 3.4 di atas merupakan flowchart halaman *Dashboard*. Halaman *Dashboard* menampilkan konten-konten yang ditampilkan dalam aplikasi *GTO System*. Menu *Course* pada *flowchart* tersebut menampilkan halaman *Course* yang berisi data-data berupa kolom *course* yang diisi oleh pengguna.

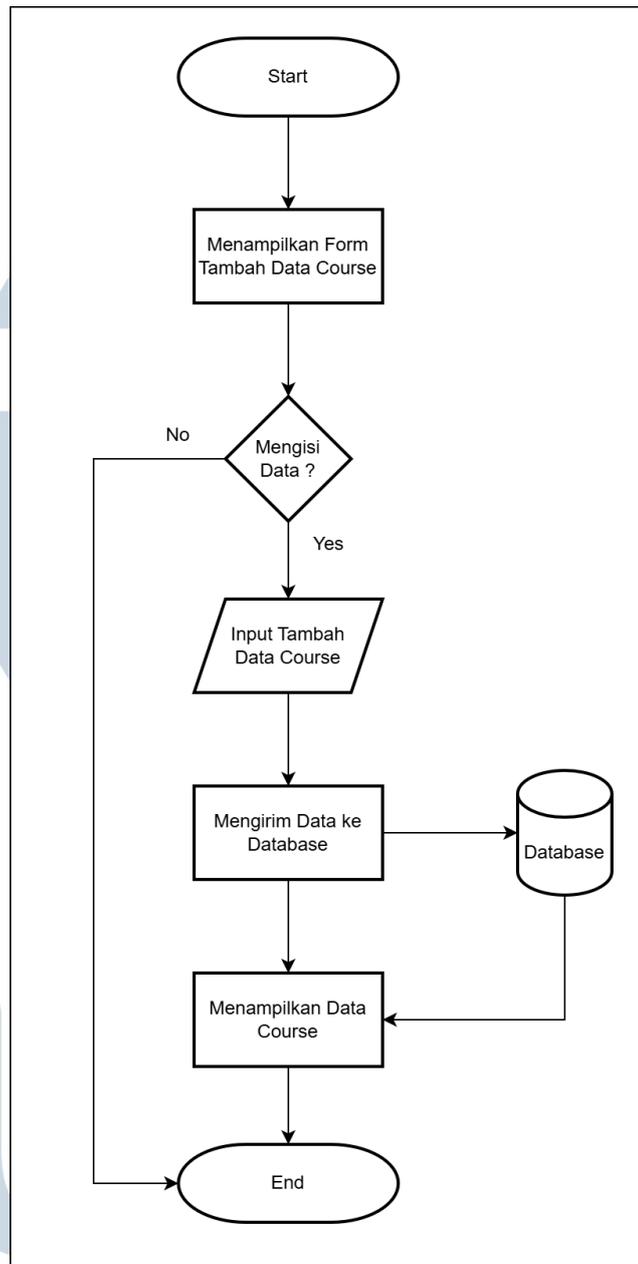
### 3. Flowchart Halaman Course



Gambar 3.6. Flowchart Halaman Course

Gambar 3.5 di atas merupakan *flowchart* halaman *course*. Halaman *course* menampilkan setiap beberapa fitur seperti menambah data, mengedit data, menghapus data, melihat data, dan mencari data.

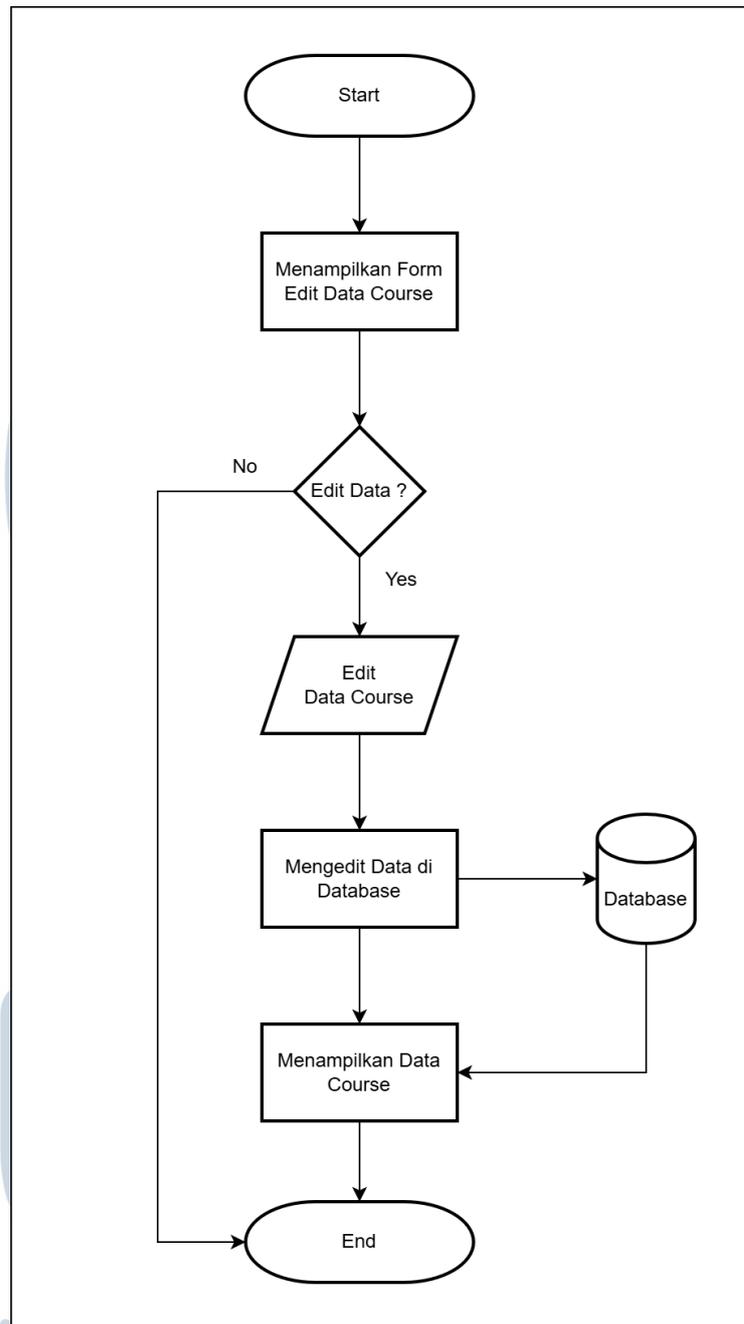
### 4. Flowchart Tambah Data Course



Gambar 3.7. Flowchart Tambah Data Course

Gambar 3.6 di atas merupakan *flowchart* tambah data pada halaman *course*. Penambahan data akan disuguhkan dengan tampilan form yang berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi supaya data terisi dengan lengkap.

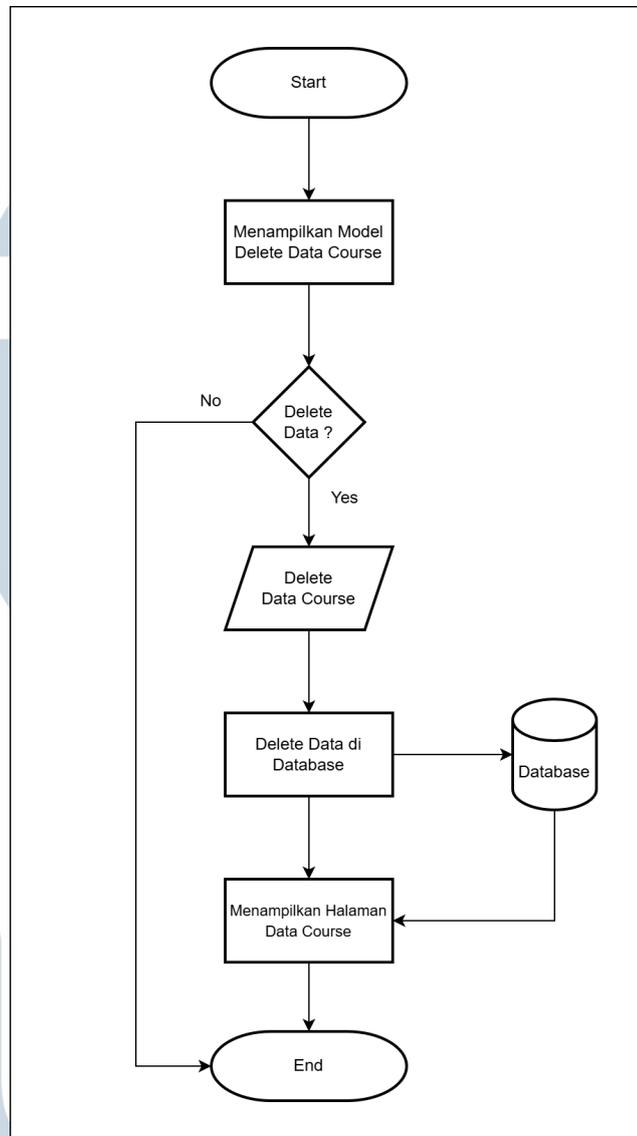
#### 5. Flowchart Edit Data Course



Gambar 3.8. Flowchart Edit Data Course

Gambar 3.7 di atas merupakan *flowchart* edit data pada halaman *course*. Pengeditan data akan disuguhkan dengan tampilan form yang sudah diisi dengan data lama pada saat penambahan data. Pengeditan data sama halnya berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi (diganti) supaya data terisi dengan lengkap,

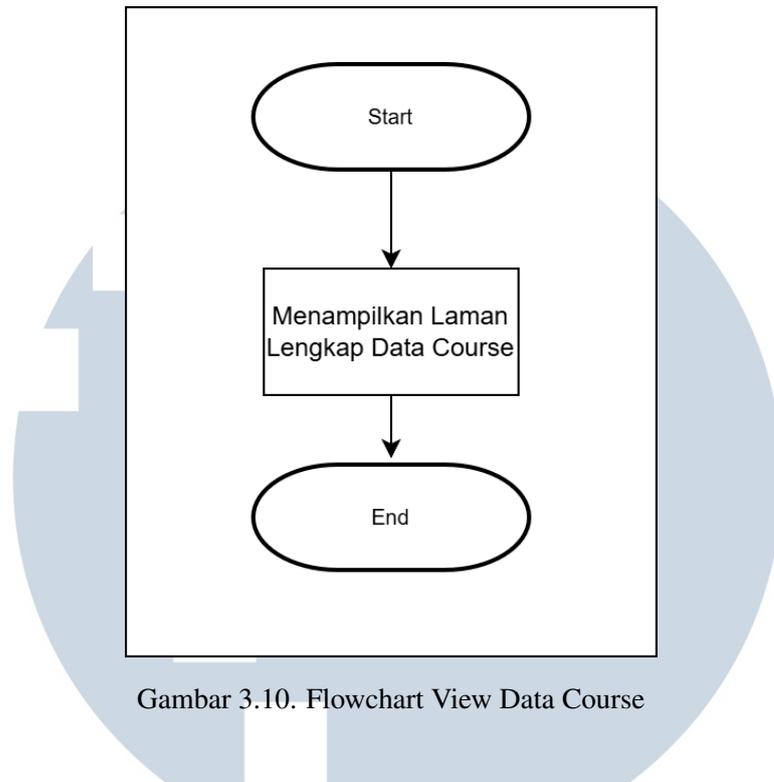
## 6. Flowchart Delete Data Course



Gambar 3.9. Flowchart Delete Data Course

Gambar 3.8 di atas merupakan *flowchart* penghapusan data pada halaman *course*. Penghapusan data oleh pengguna memiliki berbagai proses seperti pada gambar di atas. Ketika penghapusan data dilakukan maka data dalam database akan terhapus sehingga tampilan halaman *course* akan menampilkan data yang belum terhapus.

## 7. Flowchart View Data Course

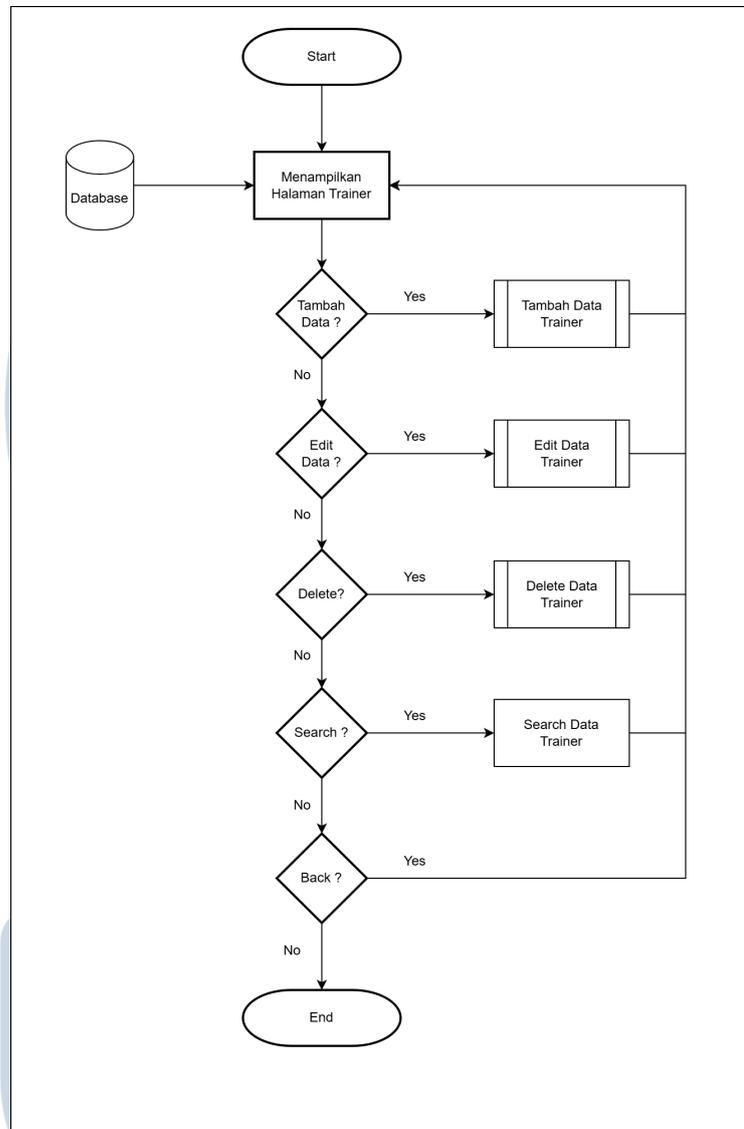


Gambar 3.10. Flowchart View Data Course

Gambar 3.9 di atas merupakan *flowchart view data* pada halaman *course*. *view data* hanya menampilkan laman data *course* secara lengkap sehingga informasi yang terlihat dapat dilihat dengan maksimal.

UMMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

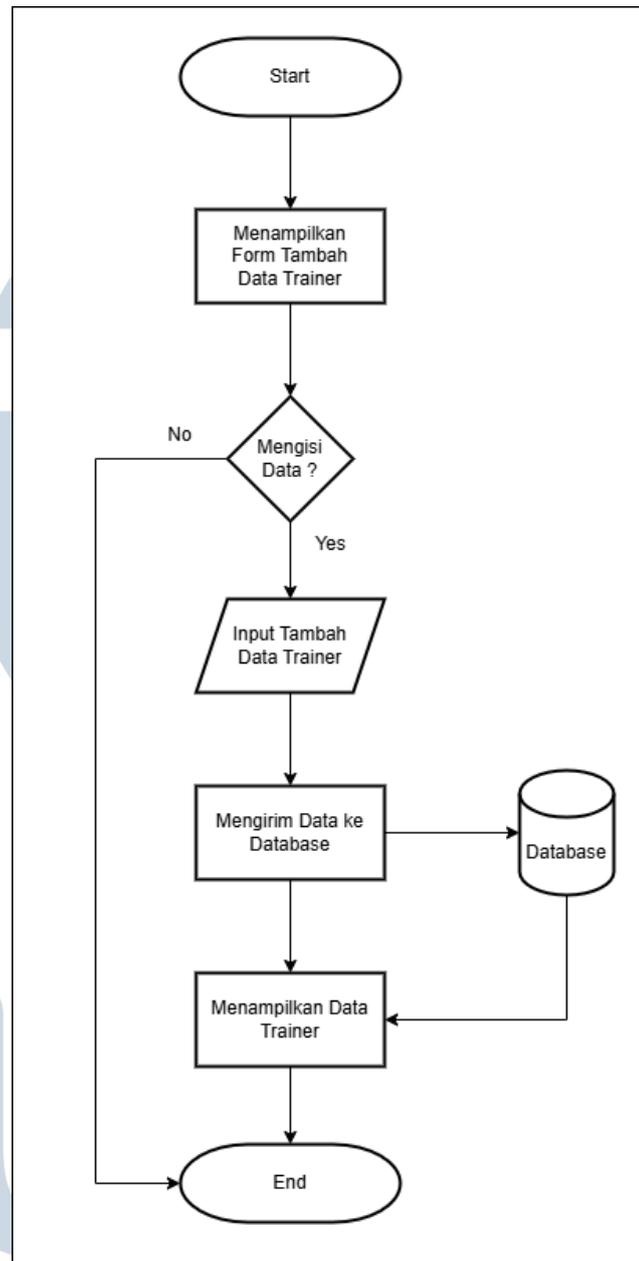
## 8. Flowchart Halaman Trainer



Gambar 3.11. Flowchart Halaman Trainer

Gambar 3.10 di atas merupakan *flowchart* halaman *trainer*. Berbeda dengan halaman *trainer*, halaman *trainer* hanya menampilkan beberapa fitur seperti menambah data, mengedit data, menghapus data, dan mencari data.

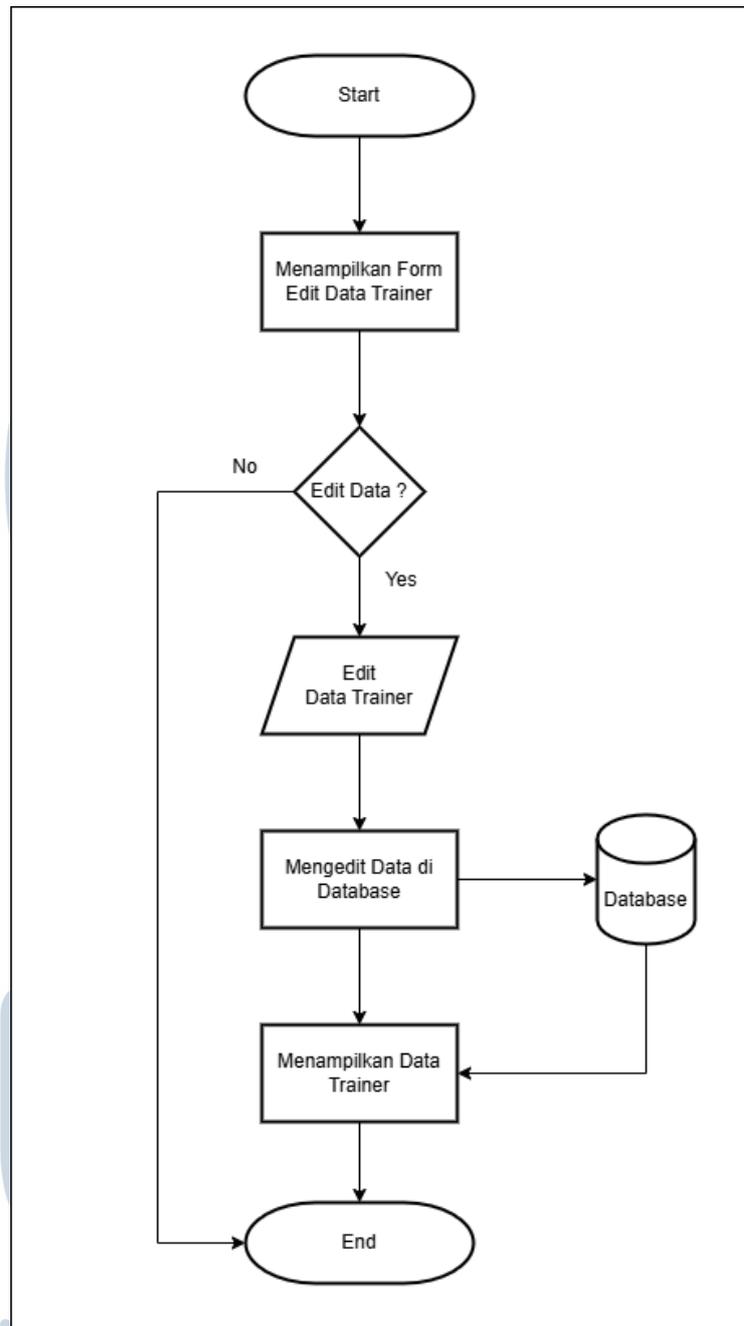
## 9. Flowchart Tambah Data Trainer



Gambar 3.12. Flowchart Tambah Data Trainer

Gambar 3.11 di atas merupakan *flowchart* tambah data pada halaman *trainer*. Penambahan data akan disuguhkan dengan tampilan form yang berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi supaya data terisi dengan lengkap.

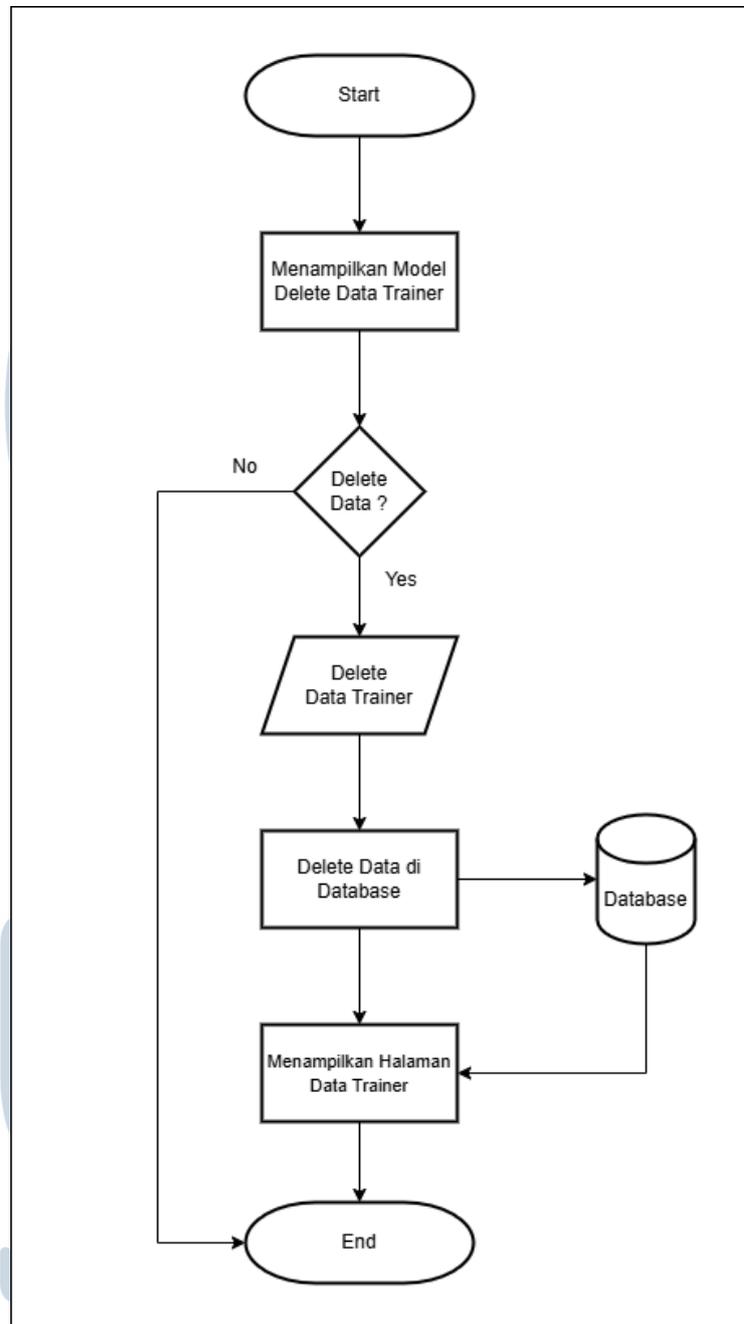
#### 10. Flowchart Edit Data Trainer



Gambar 3.13. Flowchart Edit Data Trainer

Gambar 3.12 di atas merupakan *flowchart edit data* pada halaman *trainer*. Pengeditan data akan disuguhkan dengan tampilan form yang sudah diisi dengan data lama pada saat penambahan data. Pengeditan data sama halnya berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi (diganti) supaya data terisi dengan lengkap.

## 11. Flowchart Delete Data Trainer

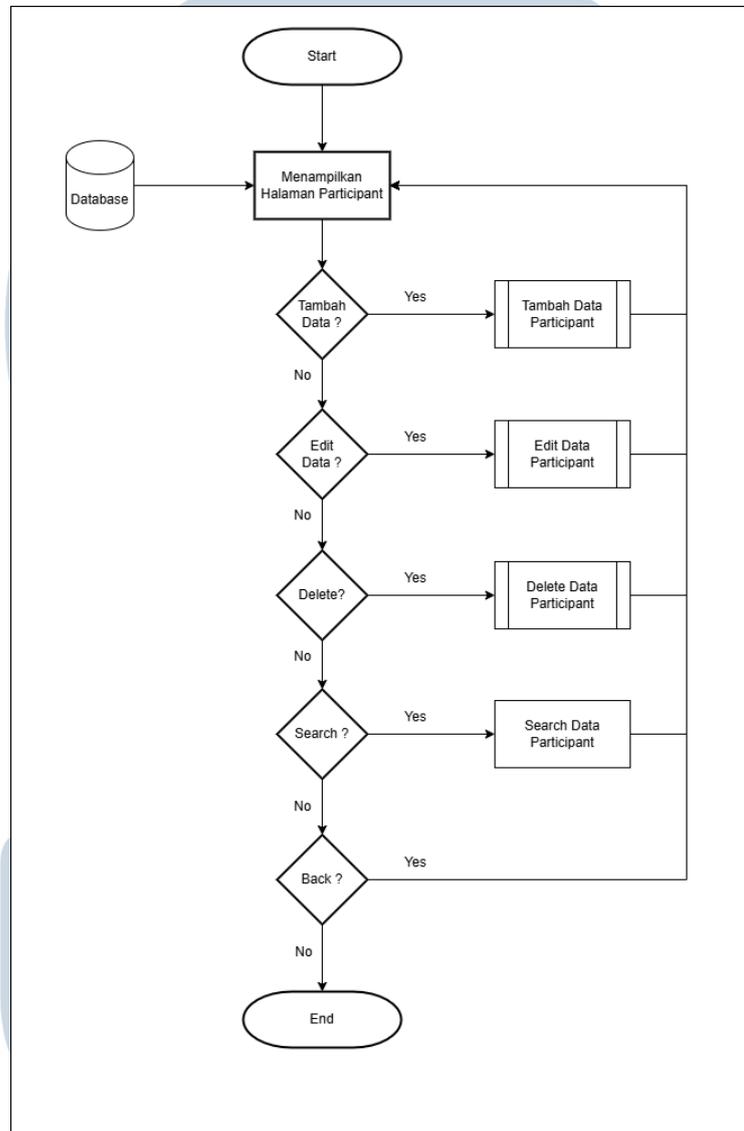


Gambar 3.14. Flowchart Delete Data Trainer

Gambar 3.13 di atas merupakan *flowchart* penghapusan data pada halaman *trainer*. Penghapusan data oleh pengguna memiliki berbagai proses seperti pada gambar di atas. Ketika penghapusan data dilakukan maka data dalam database akan terhapus sehingga tampilan halaman *trainer* akan

menampilkan data yang belum terhapus.

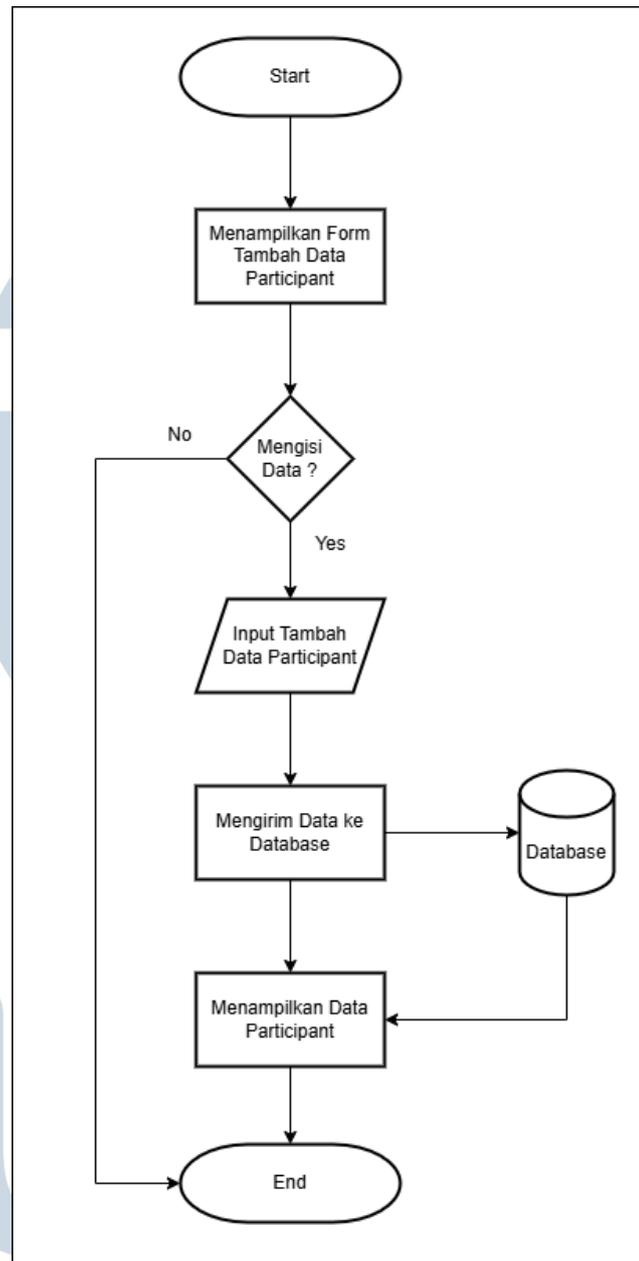
## 12. Flowchart Halaman Participant



Gambar 3.15. Flowchart Halaman Participant

Gambar 3.14 di atas merupakan *flowchart* halaman *participant*. Sama halnya dengan halaman *trainer*, halaman *participant* juga menampilkan beberapa fitur seperti menambah data, mengedit data, menghapus data, dan mencari data.

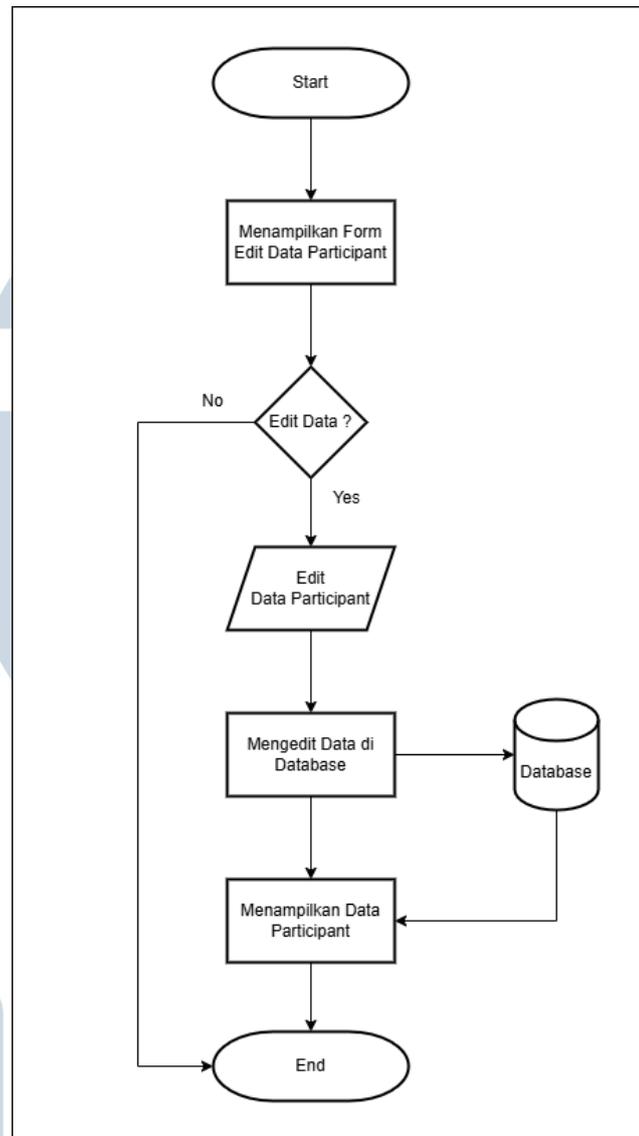
## 13. Flowchart Tambah Data Participant



Gambar 3.16. Flowchart Tambah Data Participant

Gambar 3.15 di atas merupakan *flowchart* tambah data pada halaman *participant*. Sama halnya dengan *flowchart* tambah data *trainer*, penambahan data akan disuguhkan dengan tampilan form yang berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi supaya data terisi dengan lengkap.

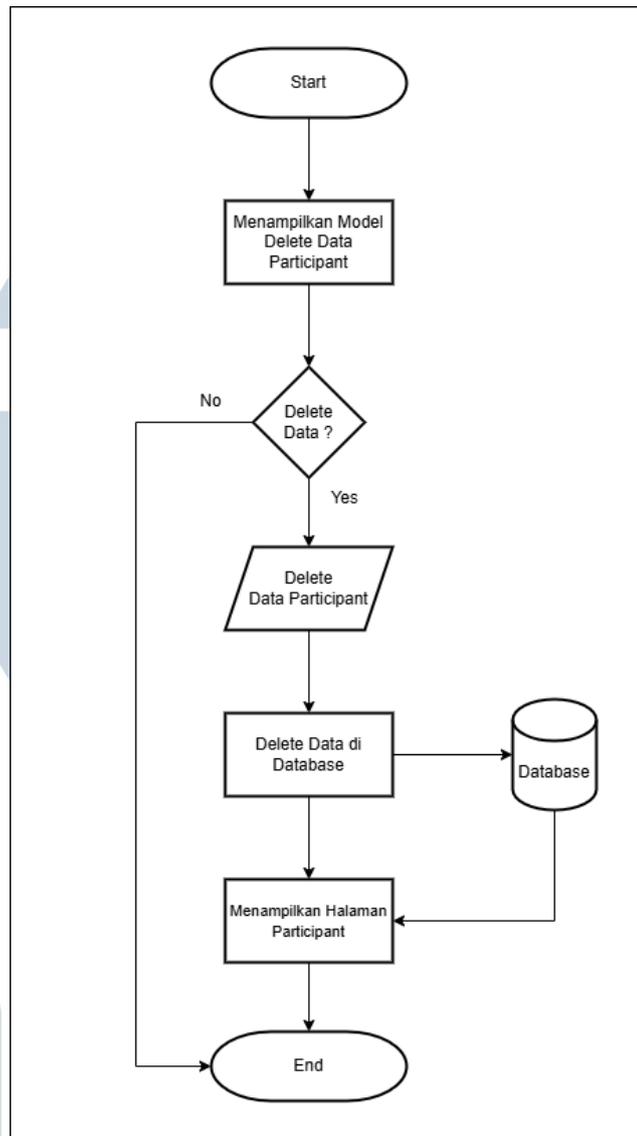
#### 14. Flowchart Edit Data Participant



Gambar 3.17. Flowchart Edit Data Participant

Gambar 3.16 di atas merupakan *flowchart edit data* pada halaman *participant*. Sama halnya dengan *flowchart trainer*, pengeditan data akan disuguhkan dengan tampilan form yang sudah diisi dengan data lama pada saat penambahan data. Pengeditan data sama halnya berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi (diganti) supaya data terisi dengan lengkap.

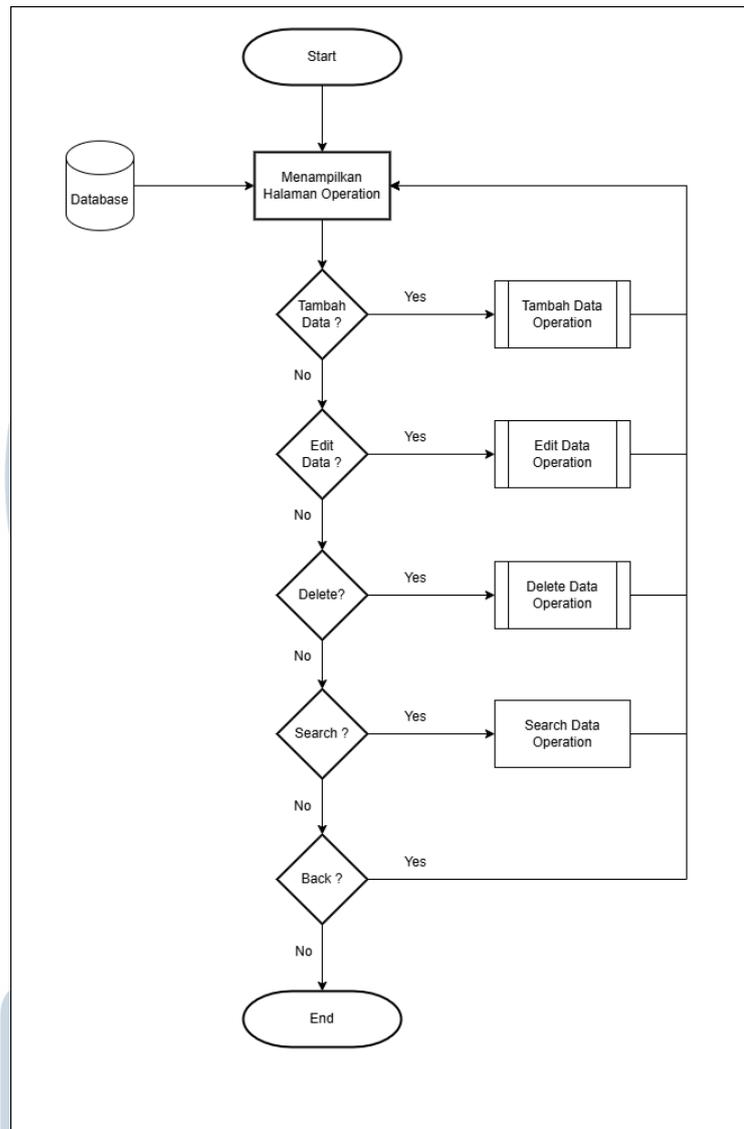
#### 15. Flowchart Delete Data Participant



Gambar 3.18. Flowchart Delete Data Participant

Gambar 3.17 di atas merupakan *flowchart* penghapusan data pada halaman *participant*. Sama halnya dengan *flowchart delete trainer*, penghapusan data oleh pengguna memiliki berbagai proses seperti pada gambar di atas. Ketika penghapusan data dilakukan maka data dalam database akan terhapus sehingga tampilan halaman *participant* akan menampilkan data yang belum terhapus.

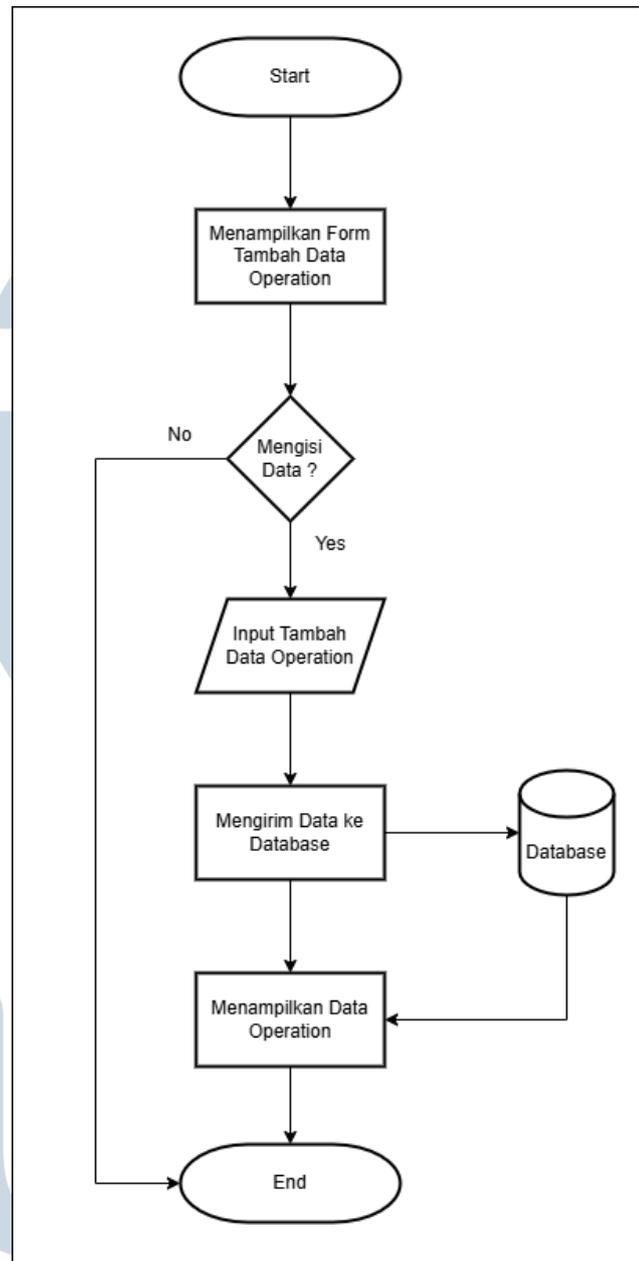
#### 16. Flowchart Halaman Operation



Gambar 3.19. Flowchart Halaman Operation

Gambar 3.18 di atas merupakan *flowchart* halaman *operation*. Sama halnya dengan halaman *trainer* dan *participant*, halaman *operation* juga menampilkan beberapa fitur seperti menambah data, mengedit data, menghapus data, dan mencari data.

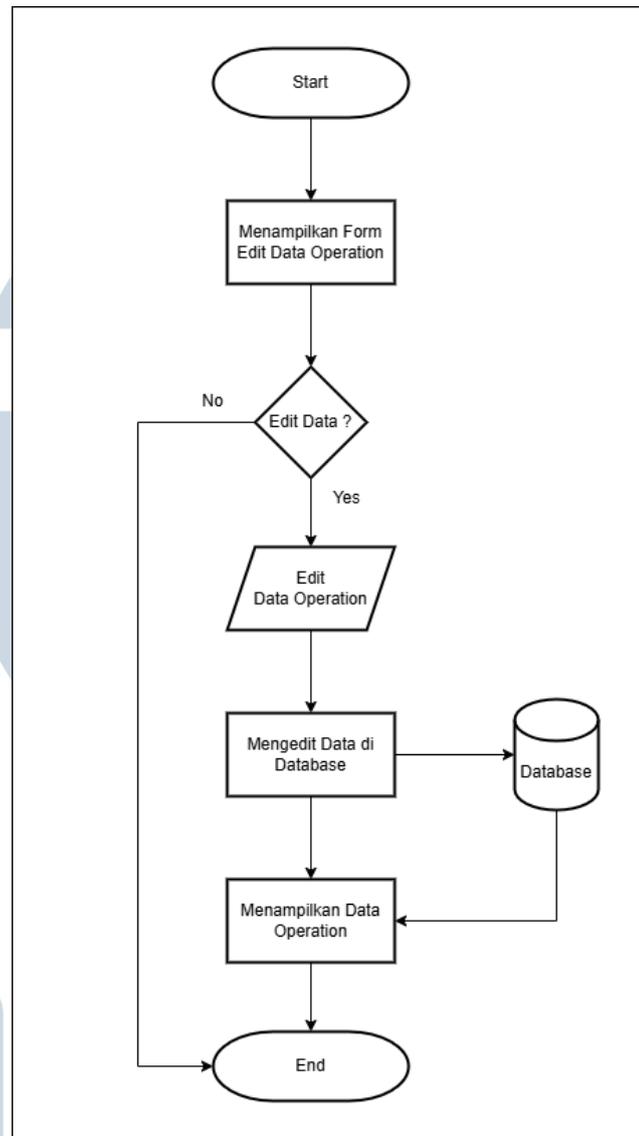
#### 17. Flowchart Tambah Data Operation



Gambar 3.20. Flowchart Tambah Data Oporteaion

Gambar 3.19 di atas merupakan *flowchart* tambah data pada halaman *operation*. Sama halnya dengan *flowchart* tambah data *trainer* dan *participant*, penambahan data akan disuguhkan dengan tampilan form yang berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi supaya data terisi dengan lengkap.

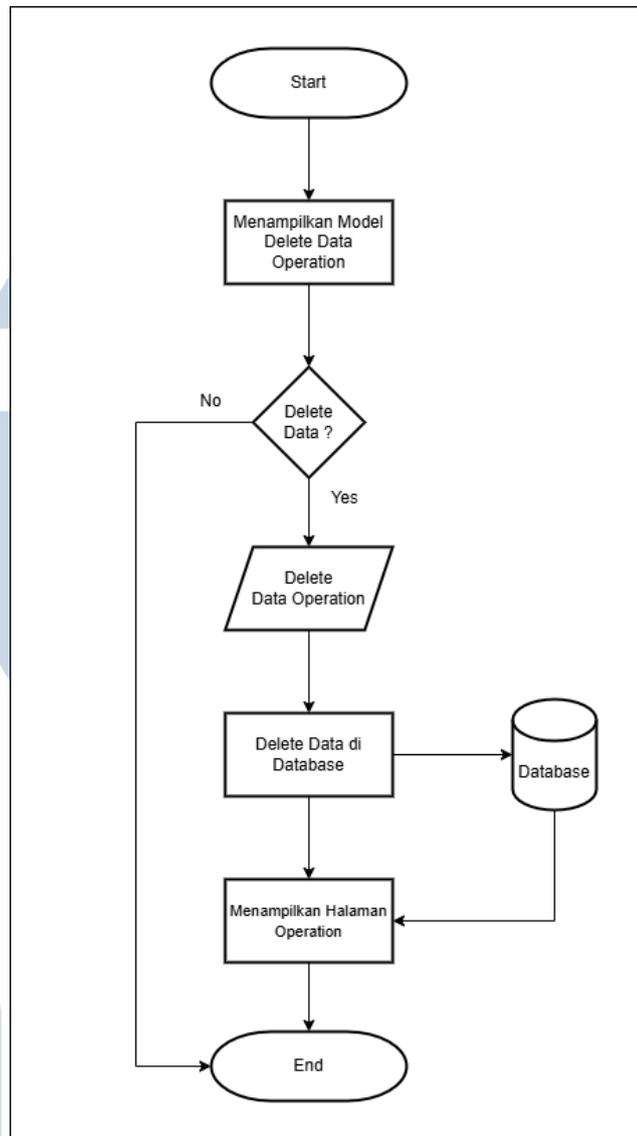
#### 18. Flowchart Edit Data Operation



Gambar 3.21. Flowchart Edit Data Operation

Gambar 3.20 di atas merupakan *flowchart edit data* pada halaman *operation*. Sama halnya dengan *flowchart trainer* dan *participant*, pengeditan data akan disuguhkan dengan tampilan form yang sudah diisi dengan data lama pada saat penambahan data. Pengeditan data sama halnya berisi berbagai proses yang harus diisi oleh pengguna. Data yang ada di dalam *form* wajib diisi (diganti) supaya data terisi dengan lengkap.

#### 19. Flowchart Delete Data Operation

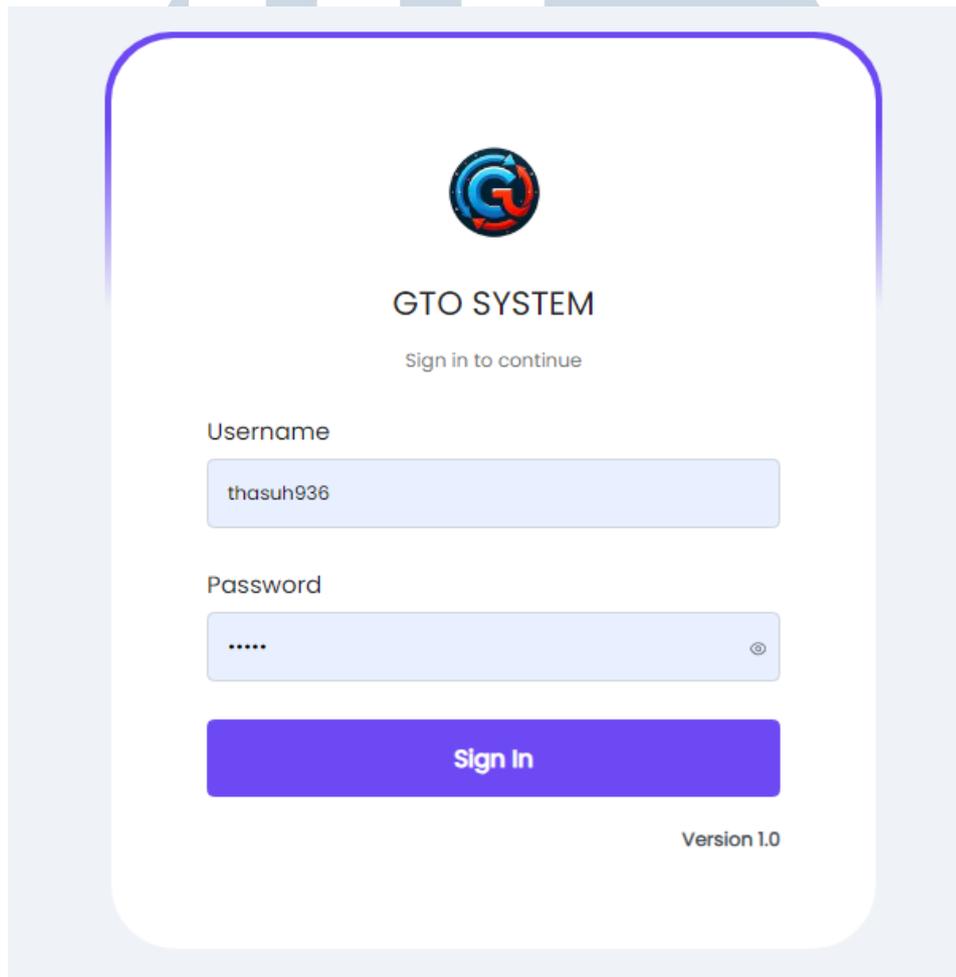


Gambar 3.22. Flowchart Delete Data Operation

Gambar 3.21 di atas merupakan *flowchart* penghapusan data pada halaman *operation*. Sama halnya dengan *flowchart delete trainer* dan *participant*, penghapusan data oleh pengguna memiliki berbagai proses seperti pada gambar di atas. Ketika penghapusan data dilakukan maka data dalam database akan terhapus sehingga tampilan halaman *operation* akan menampilkan data yang belum terhapus.

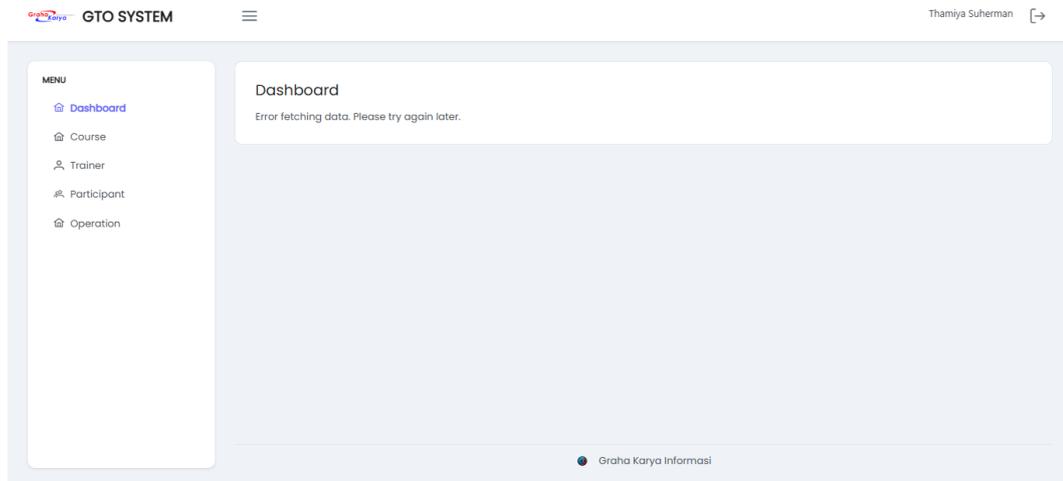
### 3.4 Aplikasi GTO System

Halaman pertama dari *website* ini adalah halaman *login*. Dengan memasukan *username* dan *password*, admin dapat masuk ke dalam halaman *dashboard* apabila proses *login* berhasil.



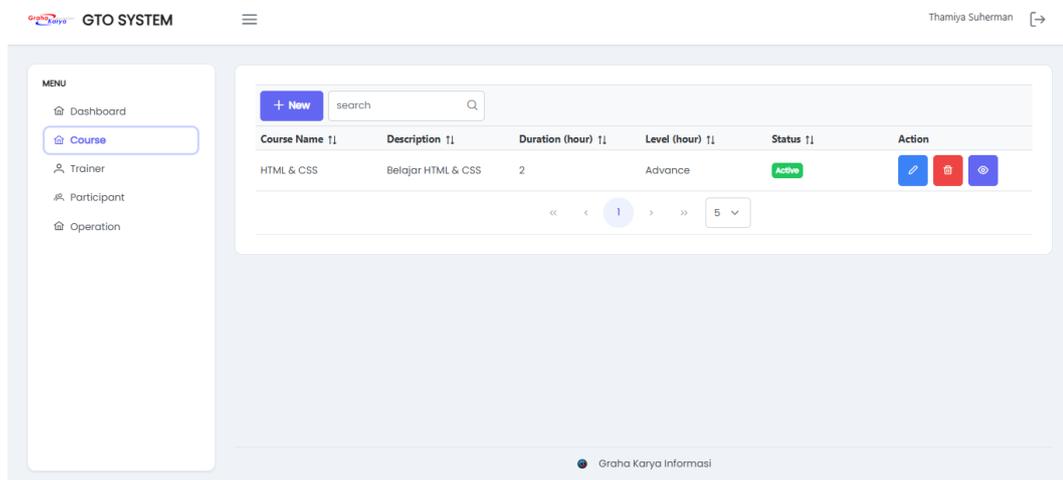
Gambar 3.23. Login GTO System

Selanjutnya adalah tampilan *dashboard*. Tampilan *dashboard* ini hanya memberikan tampilan setiap konten yang sudah dilakukan.



Gambar 3.24. Dashboard GTO System

Selanjutnya adalah tampilan halaman *course*. Halaman *course* menampilkan setiap kolom yang sudah diisi data oleh admin. Dalam halaman ini terdapat tombol "+New" yang berfungsi untuk memasukan data, ikon pensil berwarna biru yang berfungsi untuk memperbarui data, ikon tempat sampah berwarna merah yang berfungsi untuk menghapus data, dan ikon mata berwarna ungu yang berfungsi untuk melihat keseluruhan data.



Gambar 3.25. Course GTO System

Berikutnya adalah tampilan *form add course* yang muncul ketika menekan tombol "+New". Admin dapat memasukan setiap data yang dibutuhkan ke dalam *form* tersebut.

The image shows a modal window titled "Add Course" with a close button (X) in the top right corner. The form contains the following fields and controls:

- Course Name:** A text input field.
- Course Description:** A text input field.
- Duration:** A dropdown menu currently showing "1 hour(s)" with up and down arrow icons.
- Level:** A dropdown menu with the text "Select Level" and a downward arrow.
- Prerequisite:** A text input field.
- Syllabus:** A text input field.
- Materials:** A text input field.

At the bottom of the form, there are two buttons: "Cancel" (with an X icon) and "Submit" (with a checkmark icon).

Gambar 3.26. Form Add Course GTO System

Selanjutnya adalah tampilan *form edit course* yang muncul ketika menekan tombol ikon pensil. Admin dapat memperbarui setiap data yang dibutuhkan ke dalam *form* tersebut.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

The image shows a modal window titled "Edit Course" with a close button (X) in the top right corner. The form contains the following fields:

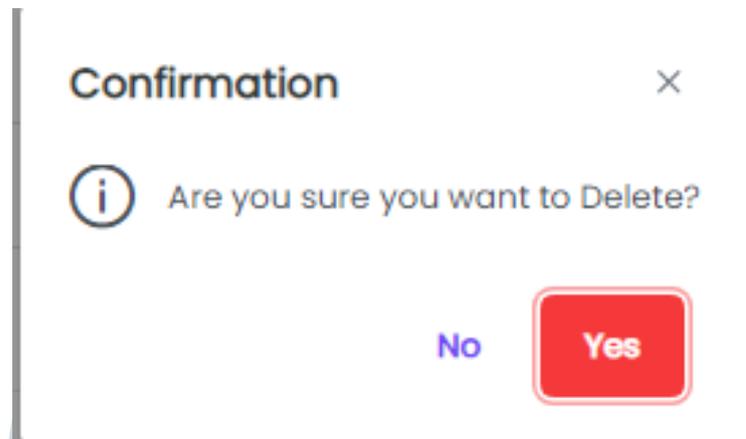
- Course Name:** Ms. Excel
- Course Description:** Microsoft
- Duration:** 4 hour(s) (with up/down arrows)
- Level:** Basic (with a dropdown arrow)
- Prerequisite:** Windows
- Syllabus:** Document
- Materials:** PDF

At the bottom right, there are two buttons: "Cancel" (with an X icon) and "Submit" (with a checkmark icon).

Gambar 3.27. Form Edit Course GTO System

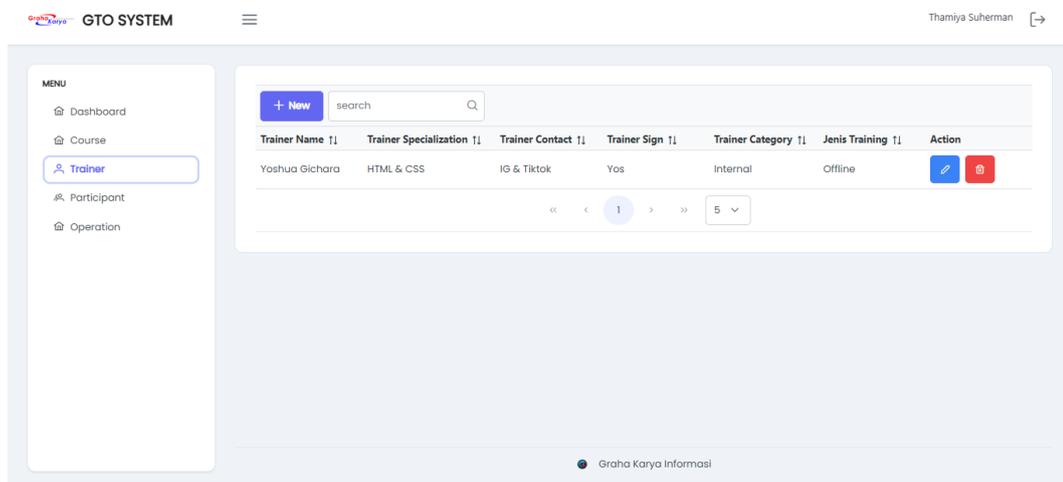
Berikutnya adalah tampilan *delete course* yang muncul ketika menekan tombol ikon tempat sampah. Admin dapat menghapus setiap data yang tampil di dalam halaman *course*.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.28. Delete Course GTO System

Gambar 3.28 adalah tampilan halaman *Trainer*. Halaman *trainer* menampilkan data *trainer* yang sudah diisi oleh admin. Dalam halaman ini juga terdapat tombol "+New" yang berfungsi untuk menambah data *trainer*, ikon pensil berwarna biru yang berfungsi untuk memperbarui data, dan ikon tempat sampah berwarna merah yang berfungsi untuk menghapus data *trainer*.



Gambar 3.29. Trainer GTO System

Gambar 3.29 adalah tampilan *form add trainer* yang muncul ketika menekan tombol "+New". Admin dapat memasukan setiap data *trainer* yang dibutuhkan ke dalam *form* tersebut.

The image shows a web form titled "Add Trainer" with a close button (X) in the top right corner. The form contains the following fields:

- Trainer Name: A text input field.
- Trainer Specialization: A text input field.
- Trainer Contact: A text input field.
- Trainer Sign: A text input field.
- Trainer Category: A dropdown menu with the text "Select Category" and a downward arrow.

At the bottom of the form, there are two buttons: "Cancel" (with a red X icon) and "Submit" (with a green checkmark icon).

Gambar 3.30. Form Add Trainer GTO System

Gambar 3.30 adalah tampilan *form edit trainer* yang muncul ketika menekan tombol ikon pensil. Admin dapat memperbarui setiap data *trainer* yang dibutuhkan ke dalam *form* tersebut.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

**Edit Trainer** ×

**Trainer Name**  
Refsi

**Trainer Specialization**  
Front End

**Trainer Contact**  
08123543

**Trainer Sign**  
RefDef

**Trainer Category**  
Internal ▾

× Cancel ✓ Submit

Gambar 3.31. Form Edit Trainer GTO System

Berikutnya adalah tampilan *delete trainer* yang muncul ketika menekan tombol ikon tempat sampah. Admin dapat menghapus setiap data yang tampil di dalam halaman *trainer*.

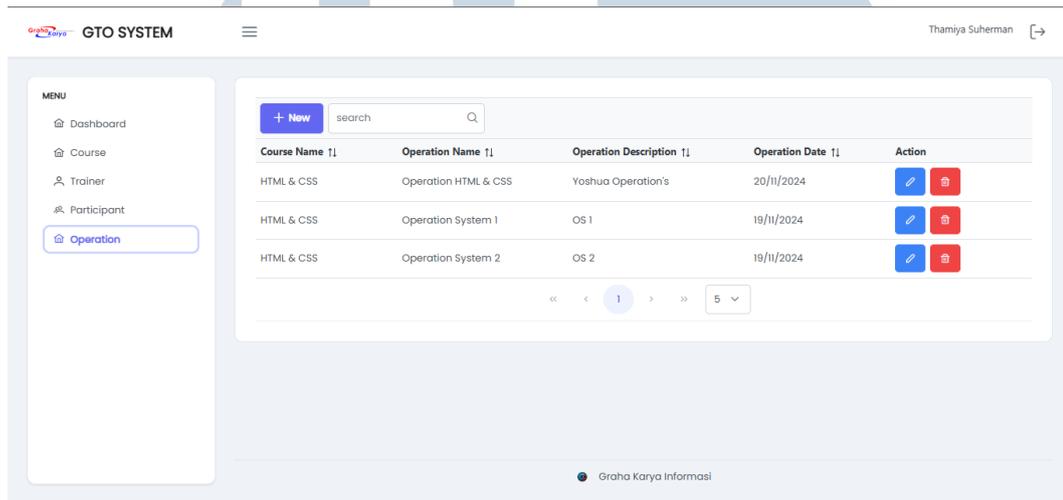
**Confirmation** ×

ⓘ Are you sure you want to Delete?

No Yes

Gambar 3.32. Delete Trainer GTO System

Gambar 3.32 adalah tampilan halaman *Operation*. Halaman *operation* menampilkan setiap kolom yang sudah diisi data oleh admin. Dalam halaman ini terdapat tombol "+New" yang berfungsi untuk memasukan data, ikon pensil berwarna biru yang berfungsi untuk memperbarui data dan ikon tempat sampah berwarna merah yang berfungsi untuk menghapus data *operation*.



Gambar 3.33. Operation GTO System

Gambar 3.33 adalah tampilan *form add operation* yang muncul ketika menekan tombol "+New". Admin dapat memasukan setiap data *operation* yang dibutuhkan ke dalam *form* tersebut.



Add Operation

Operation Name

Operation Description

Operation Date

Course Name

Select Level

Cancel Submit

Gambar 3.34. Form Add Operation GTO System

Gambar 3.34 adalah tampilan *form edit operation* yang muncul ketika menekan tombol ikon pensil. Admin dapat memperbarui setiap data *operation* yang dibutuhkan ke dalam *form* tersebut.

UMMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

The image shows a modal window titled "Edit Operation" with a close button (X) in the top right corner. The form inside has four sections: "Operation Name" with a text input containing "Developer 2"; "Operation Description" with a text input containing "Developer Web"; "Operation Date" with a date input containing "21/05/2024" and a calendar icon; and "Course Name" with a dropdown menu showing "Select Level". At the bottom of the modal, there are two buttons: a "Cancel" button with an X icon and a "Submit" button with a checkmark icon.

Gambar 3.35. Form Edit Operation GTO System

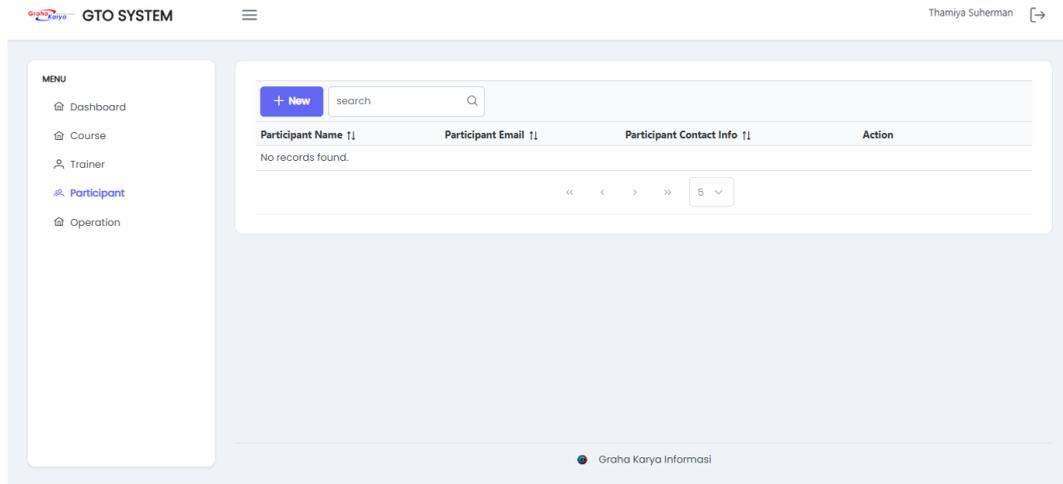
Gambar 3.35 adalah tampilan *delete operation* yang muncul ketika menekan tombol ikon tempat sampah. Admin dapat menghapus setiap data yang tampil di dalam halaman *operation*.

The image shows a modal window titled "Confirmation" with a close button (X) in the top right corner. It contains an information icon (i) and the text "Are you sure you want to Delete?". At the bottom, there are two buttons: a "No" button and a "Yes" button.

Gambar 3.36. Delete Operation GTO System

Gambar 3.36 adalah tampilan halaman *Participant*. Halaman *participant*

menampilkan data *participant* yang sudah diisi oleh admin. Dalam halaman ini, terdapat tombol "+New" yang berfungsi untuk menambah data *participant*, ikon pensil berwarna biru yang berfungsi untuk memperbarui data, dan ikon tempat sampah berwarna merah yang berfungsi untuk menghapus data *participant*.

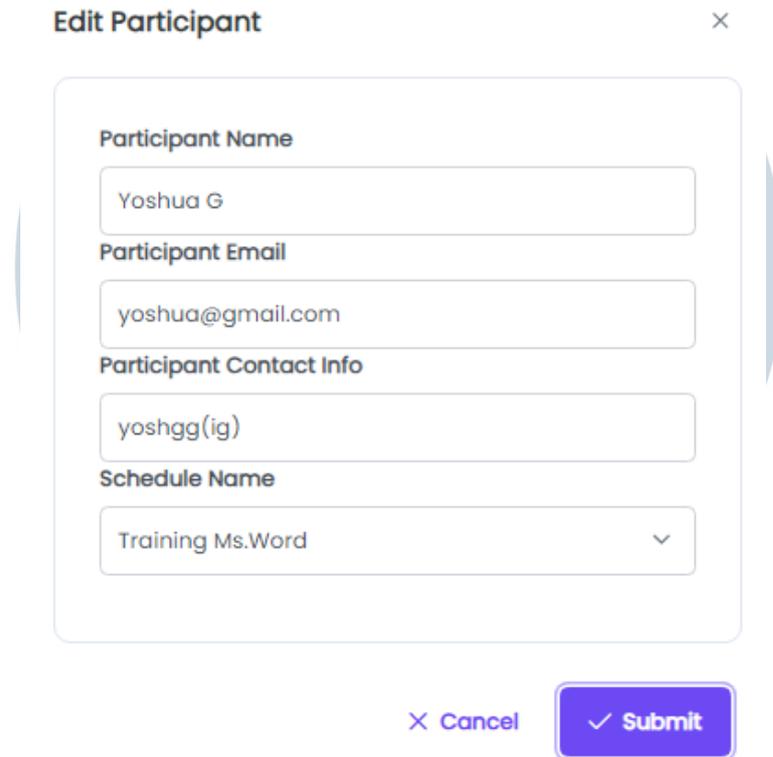


Gambar 3.37. Participant GTO System

Gambar 3.37 adalah tampilan *form add participant* yang muncul ketika menekan tombol "+New". Admin dapat memasukkan setiap data *participant* yang dibutuhkan ke dalam *form* tersebut.

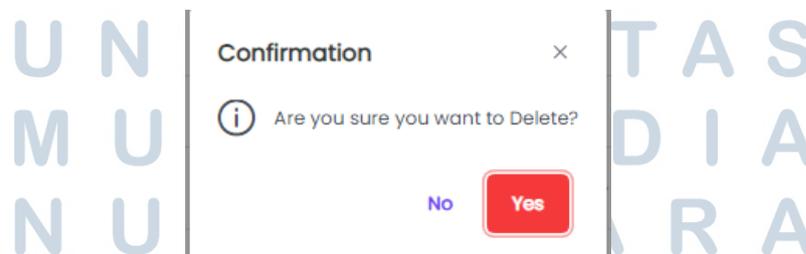
Gambar 3.38. Form Add Participant GTO System

Gambar 3.38 adalah tampilan *form edit participant* yang muncul ketika menekan tombol ikon pensil. Admin dapat memperbarui setiap data *participant* yang dibutuhkan ke dalam *form* tersebut.



Gambar 3.39. Form Edit Participant GTO System

Gambar 3.39 adalah tampilan *delete participant* yang muncul ketika menekan tombol ikon tempat sampah. Admin dapat menghapus setiap data yang tampil di dalam halaman *participant*.



Gambar 3.40. Delete Participant GTO System

### 3.5 Kendala yang Ditemukan

Kendala yang dihadapi untuk merancang *website* GTO System yaitu:

- (a) *Flow website* tidak jelas. Hal ini dikarenakan *website* ini tidak ada dulunya, jadi diharuskan membuat dari awal dan harus banyak bertanya kepada *supervisor*.
- (b) Pemahaman yang cukup sulit dalam pengerjaan proyek. Adanya materi yang belum dipelajari saat digunakan dalam pengerjaan proyek. Sehingga membutuhkan waktu yang cukup lama untuk memahami materi tersebut.
- (c) Hal yang tidak asing untuk ditemui jika sedang membuat *website* adalah *bug* dan *error*. Ini terjadi karena kurangnya komunikasi antara *development* dan pihak profesional perusahaan.

### 3.6 Solusi Atas Kendala yang Ditemukan

Banyaknya kendala tidak menghambat pembuatan *website* ini berhenti, berikut adalah solusi atas segala kendala yang terjadi:

- (a) Memberikan *flow* secara jelas agar pengerjaan lebih mudah dan tidak terlalu memakan banyak waktu.
- (b) Belajar dan latihan dengan tekun supaya mencakup materi proyek yang lebih luas sehingga tidak mengulang kesalahan yang sama.
- (c) Membagi waktu dalam mengerjakan pembuatan *website*. Sebisa mungkin menyelami internet untuk melakukan pembelajaran mandiri.
- (d) Melakukan instropeksi diri juga terkait *bug* dan *error* yang sama jika terjadi lagi dan meminta bantuan juga kepada teman kampus mengenai masalah yang terjadi.