

**PENGEMBANGAN BACK-END E-COURSE BERBASIS LARAVEL DAN  
REACT DENGAN PAYMENT GATEWAY XENDIT**

**TUGAS AKHIR**



**ANDIKA SURYA RAHMAD**

**PROGRAM STUDI D3 SISTEM INFORMASI  
DEPARTEMEN TEKNIK  
FAKULTAS VOKASI  
UNIVERSITAS AIRLANGGA  
2023**

**PENGEMBANGAN BACK-END E-COURSE BERBASIS LARAVEL DAN  
REACT DENGAN PAYMENT GATEWAY XENDIT**

**HALAMAN JUDUL**



**ANDIKA SURYA RAHMAD**

**152011513034**

**PROGRAM STUDI D3 SISTEM INFORMASI**

**DEPARTEMEN TEKNIK**

**FAKULTAS VOKASI**

**UNIVERSITAS AIRLANGGA**

**2023**

## HALAMAN PERNYATAAN TIDAK MELAKUKAN PLAGIAT

Bagian atau keseluruhan isi Tugas Akhir ini tidak pernah diajukan untuk mendapatkan gelar akademis pada bidang studi dan / atau Universitas lain dan tidak pernah dipublikasikan / ditulis oleh individu selain penyusun, kecuali bila dituliskan dengan format dalam isi Tugas Akhir.

Apabila ditemukan bukti bahwa pernyataan saya tidak benar, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku di Universitas Airlangga.

Surabaya, 27 Juli 2023

Penulis



Andika Surya Rahmad  
NIM. 152011513034

**LEMBAR PERSETUJUAN TUGAS AKHIR**

**PENGEMBANGAN BACK-END E-COURSE BERBASIS LARAVEL DAN  
REACT DENGAN PAYMENT GATEWAY XENDIT**

**TUGAS AKHIR**

Sebagai Salah Satu Syarat Untuk Dinyatakan Lulus

Program Studi D3 Sistem Informasi pada Fakultas Vokasi

Universitas Airlangga

Oleh:

Andika Surya Rahmad

NIM. 152011513034

Disetujui oleh:

Dosen Pembimbing



Tesa Eranti Putri, S.Kom., M.Kom

NIP. 198904232016033201

**LEMBAR PENGESAHAN NASKAH  
TUGAS AKHIR**

Judul : Pengembangan Back-End E-Course Berbasis Laravel dan  
Rect Dengan Payment Gateway XENDIT

Penulis : Andika Surya Rahmad

NIM : 152011513034

Pembimbing : Tesa Eranti Putri, S.Kom., M.Kom.

Tanggal Ujian : 20 Juli 2023

Disetujui Oleh,

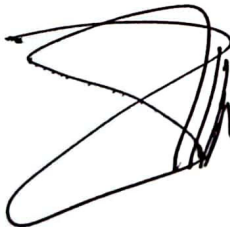
Dosen Pembimbing,



Tesa Eranti Putri, S.Kom., M.Kom.

NIP. 198904232016033201

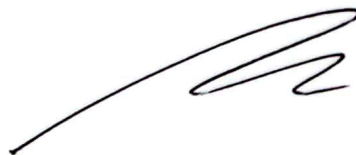
Ketua Departmen Teknik,



Drs. Eto Wuryanto, DEA.

NIP. 196609281991021001

Koordinator Program Studi D3  
Sistem Informasi,



Fitri Retrialisca, S.Kom., M.Kom.

NIP. 199303302018083201

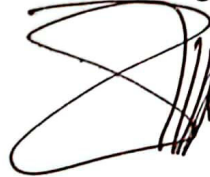
**LEMBAR PENGESAHAN PANITIA PENGUJI**

Tugas Akhir Ini Telah Diujikan dan Disahkan Dihadapan Komisi Penguji

Program Studi : D-III Sistem Informasi  
Departemen : Teknik  
Fakultas : Vokasi Universitas Airlangga  
Hari / Tanggal : Kamis / 20 Juli 2023  
Pukul : 13.00 - 14.30 WIB  
Tempat : Ruang 208, Fakultas Vokasi Kampus B Universitas  
Airlangga.

Panitia Penguji terdiri dari:

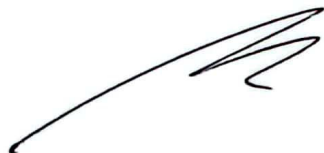
Ketua Penguji



Drs. Eto Wuryanto, DEA.

NIP. 196609281991021001

Penguji I



Fitri Retrialisca, S.Kom., M.Kom.

NIP. 199303302018083201

Penguji II



Tesa Eranti Putri, S. Kom., M. Kom.

NIP. 198904232016033201

## **PEDOMAN PENGGUNAAN TUGAS AKHIR**

Naskah Tugas Akhir ini tidak dipublikasikan, namun tersedia di perpustakaan dalam lingkungan Universitas Airlangga. Diperkenankan untuk dipakai sebagai referensi kepustakaan, tetapi pengutipan harus seijin penulis dan harus menyebutkan sumbernya sesuai kebiasaan ilmiah.

**Dokumen Tugas Akhir ini merupakan hak milik Universitas Airlangga.**

## KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “E-Course”.

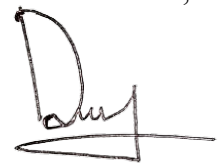
Dalam pembuatan tugas akhir ini penulis tidak lupa untuk mengucapkan terima kasih kepada seluruh pihak yang ikut andil dalam menyelesaikan Tugas Akhir ini, diantaranya adalah :

Penulis menyadari sepenuhnya bahwa penulisan tugas akhir ini masih jauh dari kesempurnaan. Oleh sebab itu, segala kritik, saran dan koreksi yang membangun sangat diharapkan penulis demi kesempurnaan penelitian selanjutnya.

Akhir kata penulis berharap semoga tugas akhir ini bermanfaat terutama bagi penyusun sendiri dan bagi mitra PT. Otak Kanan khususnya serta bagi para pembaca agar dapat digunakan untuk menambah referensi dan dapat digunakan dengan sebaik-baiknya sebagai sumber inspirasi.

Surabaya, 26 Juli 2023

Penulis,



Andika Surya Rahmad

## UCAPAN TERIMA KASIH

Penulis ingin mengucapkan terima kasih dan penghargaan yang sebesar-besarnya kepada seluruh pihak yang telah membantu Penulis dalam proses pengerjaan dan pelaksanaan Tugas Akhir ini. Penulis ingin mengucapkan terimakasih kepada :

1. Tuhan Yang Maha Esa atas pertolongan-Nya sehingga Penulis dapat menyelesaikan Tugas Akhir ini.
2. Orang tua yang telah memberikan doa dan dukungan sehingga Penulis dapat menyelesaikan Tugas Akhir ini dengan baik.
3. Ibu Tesa Eranti Putri, S.Kom., M.Kom. selaku Dosen Pembimbing, yang telah memberikan banyak inspirasi, motivasi, berbagi pengalaman, pengetahuan, serta saran dan kritik yang sangat membangun sehingga Penulis dapat menyelesaikan Tugas Akhir ini.
4. Pak Wahyu Prasetyo selaku mentor magang PT.Otak Kanan yang telah memandu berjalannya proyek E-Course ini sampai akhir.
5. Seluruh pihak yang turut mengambil peran dalam penyelesaian Tugas Akhir ini, yang tidak dapat disebutkan satu persatu.

Surabaya, 26 Juli 2023

Penulis,



Andika Surya Rahmad

Andika Surya Rahmad, 2023, Pengembangan Back-End E-Course Berbasis Laravel dan Rect Dengan Payment Gateway XENDIT. Tugas Akhir ini dibawah bimbingan Tesa Eranti Putri, S.Kom., M.Kom Program Studi Diploma Tiga Sistem Informasi, Fakultas Vokasi, Universitas Airlangga.

---

## ABSTRAK

Tujuan dari Tugas Akhir ini adalah membuat backend learning management system E-Course mulai dari proses pendaftaran, pelaksanaan pelatihan, hingga proses pengelolaan data pelatihan yang dapat menjadi sarana belajar mengajar efektif bagi peserta dan juga pengajar.

Dalam membuat *backend learning management system* E-Course dibutuhkan lima tahap. Tahap pertama mengidentifikasi kebutuhan data dengan meeting dengan mentor magang PT. Otak Kanan sebagai klien dalam proyek ini. Tahap kedua menganalisis permasalahan sistem saat ini dengan diagram fishbone dan *Document Flow Diagram* (Docflow). Tahap ketiga mendesain sistem dengan *Hierarchy Input Process Output* (HIPO) Diagram dan Data Flow Diagram (DFD) serta mendesain database dengan *Conceptual Data Model* (CDM) yang menghasilkan 19 entitas dan *Physical Data Model* (PDM) yang menghasilkan 19 tabel. Kemudian mendesain form input dan output. Tahap keempat menggambarkan sistem dengan Bagan Alir, serta mengimplementasikan kedalam program berbasis web. Tahap terakhir melakukan uji coba sistem.

Berdasarkan hasil implementasi dan uji coba backend learning management system E-Course yang dibuat telah sesuai dengan kebutuhan fungsional yaitu dapat menangani kendala dari beberapa proses bisnis yang telah ada dan dapat menampilkan data pelatihan yang diinginkan secara benar dan terdigitalisasi.

Kata Kunci: React.js, Laravel, Payment Gateway Xendit, *backend*, Pelatihan, *learning management system*, E-Course.

Andika Surya Rahmad, 2023, *Laravel and Rect Based E-Course Back-End Development With XENDIT Payment Gateway. This Final Project had been guiding by Tesa Eranti Putri, S.Kom., M.Kom Diploma Three of Information System Study Program, Faculty of Vocational, Airlangga University.*

---

## ABSTRACT

*The purpose of this Final Project is to create a backend learning management system for E-Course starting from the registration process, training implementation, to the training data management process which can be an effective teaching and learning tool for participants and also teachers.*

*In making the backend learning management system E-Course, five stages are needed. The first stage is identifying data needs by meeting with the mentor apprentice PT. Otak Kanan as a client in this project. The second stage analyzes current system problems with fishbone diagrams and Document Flow Diagrams (Docflow). The third stage is designing a system with Hierarchy Input Process Output (HIPO) Diagrams and Data Flow Diagrams (DFD) and designing a database using a Conceptual Data Model (CDM) which produces 19 entities and a Physical Data Model (PDM) which produces 19 tables. Then design the input and output forms. The fourth stage describes the system with a flowchart, and implements it into a web-based program. The final stage is testing the system.*

*Based on the results of the implementation and testing of the E-Course backend learning management system that has been made in accordance with functional requirements, namely being able to handle the constraints of several existing business processes and being able to display the desired training data correctly and digitally.*

Kata Kunci: React.js, Laravel, *Payment Gateway Xendit, backend, Training, learning management system, E-Course.*

## DAFTAR ISI

|                                                  |      |
|--------------------------------------------------|------|
| HALAMAN JUDUL.....                               | i    |
| HALAMAN PERNYATAAN TIDAK MELAKUKAN PLAGIAT ..... | ii   |
| LEMBAR PERSETUJUAN TUGAS AKHIR.....              | iii  |
| LEMBAR PENGESAHAN NASKAH TUGAS AKHIR.....        | iv   |
| LEMBAR PENGESAHAN PANITIA PENGUJI .....          | v    |
| PEDOMAN PENGGUNAAN TUGAS AKHIR .....             | vi   |
| KATA PENGANTAR .....                             | vii  |
| UCAPAN TERIMA KASIH.....                         | viii |
| ABSTRAK .....                                    | ix   |
| ABSTRACT.....                                    | x    |
| DAFTAR ISI.....                                  | xi   |
| DAFTAR GAMBAR .....                              | xiii |
| DAFTAR TABEL.....                                | xvi  |
| DAFTAR LAMPIRAN .....                            | xvii |
| BAB I PENDAHULUAN .....                          | 1    |
| 1.1    Latar Belakang .....                      | 1    |
| 1.2    Rumusan Masalah .....                     | 3    |
| 1.3    Tujuan.....                               | 3    |
| 1.4    Manfaat.....                              | 4    |
| 1.5    Batasan Masalah.....                      | 4    |
| BAB II ANALISIS SISTEM .....                     | 5    |
| 2.1    Profil Organisasi.....                    | 5    |
| 2.2    Sistem Kerja Saat Ini .....               | 8    |
| 2.3    Analisis Permasalahan System .....        | 17   |
| 2.4    kebutuhan Fungsionalitas Sistem .....     | 20   |
| BAB III DESAIN SISTEM.....                       | 22   |
| 3.1    Sistem Kerja Baru .....                   | 22   |
| 3.2    Desain Proses .....                       | 29   |

|                                        |                                    |     |
|----------------------------------------|------------------------------------|-----|
| 3.3                                    | Desain Database .....              | 34  |
| 3.4                                    | Desain Antarmuka .....             | 48  |
| BAB IV IMPLEMENTASI DAN UJI COBA ..... |                                    | 66  |
| 4.1                                    | Pengembangan <i>REST-API</i> ..... | 66  |
| 4.2                                    | Implementasi Sistem .....          | 73  |
| 4.3                                    | Uji Coba Sistem .....              | 92  |
| 4.3                                    | Analisis Hasil Pengujian .....     | 105 |
| BAB V KESIMPULAN DAN SARAN .....       |                                    | 106 |
| 5.1                                    | Kesimpulan .....                   | 106 |
| 5.2                                    | Saran .....                        | 107 |
| DAFTAR PUSTAKA .....                   |                                    | 108 |
| LAMPIRAN .....                         |                                    | 109 |

## DAFTAR GAMBAR

|                                                                                 |    |
|---------------------------------------------------------------------------------|----|
| Gambar 2.1 Logo PT. Otak Kanan (Sumber : otakkanan.co.id, 2023).....            | 5  |
| Gambar 2.2 Struktur Organisasi PT. Otak Kanan.....                              | 6  |
| Gambar 2.3 <i>Document Flow Diagram</i> Pendaftaran Pelatihan Konvensional..... | 13 |
| Gambar 2.4 <i>Document Flow Diagram</i> Pelaksanaan Pelatihan .....             | 14 |
| Gambar 2.5 <i>Document Flow Diagram</i> Pengerjaan Ujian .....                  | 15 |
| Gambar 2.6 <i>Document Flow Diagram</i> Penilaian.....                          | 16 |
| Gambar 2.7 <i>Fishbone Diagram</i> .....                                        | 19 |
| Gambar 3.1 Hierarchy Input Process Output (HIPO) Diagram .....                  | 30 |
| Gambar 3.2 Context Diagram .....                                                | 32 |
| Gambar 3.3 Data Flow Diagram Level 0 .....                                      | 33 |
| Gambar 3.4 Data Flow Diagram Level 1 Proses Pengelolaan Data Pelatihan.....     | 34 |
| Gambar 3.5 Conceptual Data Model (CDM).....                                     | 36 |
| Gambar 3.6 Physical Data Model (PDM) .....                                      | 42 |
| Gambar 3.7 Desain Form Login .....                                              | 51 |
| Gambar 3.8 Desain Halaman Form Registrasi.....                                  | 52 |
| Gambar 3.9 Desain Form Pencarian Pelatihan .....                                | 53 |
| Gambar 3.10 Desain halaman Tambah Pembayaran .....                              | 53 |
| Gambar 3.11 Desain Form Tambah Data Profile.....                                | 54 |
| Gambar 3.12 Desain Form Tambah Data Pelatihan.....                              | 55 |
| Gambar 3.13 Desain Form Tambah Data Kurikulum dan Materi Pelatihan.....         | 56 |
| Gambar 3.14 Desain Halaman Landing Page .....                                   | 57 |
| Gambar 3.15 Desain Halaman Hasil Pencarian Kursus.....                          | 58 |
| Gambar 3.16 Desain Halaman Detail pelatihan .....                               | 59 |
| Gambar 3.17 Desain Halaman Dashboard Siswa/Peserta.....                         | 60 |
| Gambar 3.18 Desain Halaman Kursus Saya .....                                    | 61 |
| Gambar 3.19 Desain Halaman Transaksi.....                                       | 62 |
| Gambar 3.20 Desain Halaman Pembelajaran.....                                    | 63 |
| Gambar 3.21 Desain Halaman Kursus .....                                         | 64 |
| Gambar 3.22 Desain Halaman Detail Kursus .....                                  | 65 |

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Gambar 4.1 Tahap Pengembangan REST-API.....                          | 66 |
| Gambar 4.2 Implementasi endpoint API /course/popular.....            | 72 |
| Gambar 4.3 eksekusi url /course/popular pada postman.....            | 72 |
| Gambar 4.1 Bagan Alir Login.....                                     | 74 |
| Gambar 4.2 Sequence Diagram Alir Login.....                          | 75 |
| Gambar 4.3 Bagan Alir Registrasi .....                               | 76 |
| Gambar 4.4 Sequence Diagram Registrasi.....                          | 77 |
| Gambar 4. 5 Bagan Alir Menu Siswa/Pelajar .....                      | 78 |
| Gambar 4. 6 Bagan Alir Menu Admin.....                               | 78 |
| Gambar 4. 7 Bagan Alir Menu Pengajar.....                            | 79 |
| Gambar 4.8 Bagan Alir Pendaftaran Pelatihan .....                    | 80 |
| Gambar 4.9 Sequence Diagram Pendaftaran Pelatihan.....               | 81 |
| Gambar 4.10 Bagan Alir Dashboard Peserta .....                       | 81 |
| Gambar 4.11 Bagan Alir Pelatihan Peserta.....                        | 82 |
| Gambar 4.12 Bagan Alir Data Transaksi Peserta.....                   | 82 |
| Gambar 4.13 Bagan Alir Dashboard Admin.....                          | 83 |
| Gambar 4.14 Bagan Alir Semua Pelatihan Admin .....                   | 83 |
| Gambar 4.15 Bagan Alir Menyetujui Pelatihan Admin.....               | 84 |
| Gambar 4.16 Bagan Alir Membuat Kategori Admin.....                   | 85 |
| Gambar 4.17 Bagan Alir Memperbarui Data Kategori Admin.....          | 87 |
| Gambar 4.18 Bagan Alir Menghapus Data Kategori Admin.....            | 87 |
| Gambar 4.19 Bagan Alir Menampilkan Semua Pelatihan Pengajar .....    | 88 |
| Gambar 4.20 Bagan Alir Menambah Data Pelatihan Pengajar.....         | 89 |
| Gambar 4.21 Bagan Alir Menambah Data Materi Pelatihan Pengajar ..... | 90 |
| Gambar 4.22 Bagan Alir Mengajukan Pelatihan Pengajar .....           | 91 |
| Gambar 4.23 Fitur Login.....                                         | 92 |
| Gambar 4.24 Skema Pengujian Berhasil Login Menggunakan Postman ..... | 93 |
| Gambar 4.25 Skema Pengujian Gagal Login 1 Menggunakan Postman .....  | 93 |
| Gambar 4.26 Skema Pengujian Gagal Login 2 Menggunakan Postman .....  | 94 |
| Gambar 4.27 Fitur Registrasi .....                                   | 95 |

|                                                                          |     |
|--------------------------------------------------------------------------|-----|
| Gambar 4.28 Skema Pengujian Berhasil Menggunakan Postman .....           | 95  |
| Gambar 4.29 Skema Pengujian Gagal Menggunakan Postman.....               | 96  |
| Gambar 4.30 Fitur List Pelatihan .....                                   | 97  |
| Gambar 4.31 Pengujian API Pelatihan Menggunakan API .....                | 97  |
| Gambar 4.32 Fitur Pembelian Pelatihan .....                              | 99  |
| Gambar 4.33 Skema Pengujian Pembelian Berhasil Menggunakan Postman ..... | 99  |
| Gambar 4.34 Skema Pengujian Pembelian Gagal 1 Menggunakan Postman .....  | 100 |
| Gambar 4.35 Skema Pengujian Pembelian Gagal 2 Menggunakan Postman .....  | 100 |
| Gambar 4.36 Data Pembayaran Berstatus Pending Sebelum Pengujian.....     | 102 |
| Gambar 4.37 Pengujian Callback.....                                      | 102 |
| Gambar 4.38 Fitur Histori Pembayaran .....                               | 103 |
| Gambar 4.39 Pengujian Ketika Terdapat Data yang Valid.....               | 104 |
| Gambar 4.40 Pengujian Ketika Tidak Terdapat Data yang Valid .....        | 104 |

## DAFTAR TABEL

|                                                                       |     |
|-----------------------------------------------------------------------|-----|
| Tabel 3.1 Tabel Objek pada Physical Data Model (PDM) .....            | 41  |
| Tabel 3.2 Struktur Tabel Role.....                                    | 43  |
| Tabel 3. 3 Struktur Tabel User .....                                  | 43  |
| Tabel 3.4 Struktur Tabel Profile.....                                 | 44  |
| Tabel 3.5 Struktur Tabel category.....                                | 44  |
| Tabel 3.6 Struktur Tabel Course .....                                 | 45  |
| Tabel 3.7 Struktur Tabel Materials .....                              | 45  |
| Tabel 3.8 Struktur Tabel Item .....                                   | 46  |
| Tabel 3.9 Struktur Tabel video_access .....                           | 46  |
| Tabel 3.10 Struktur Tabel Course_active.....                          | 47  |
| Tabel 3.11 Struktur Tabel Orders.....                                 | 47  |
| Tabel 3.12 Struktur Tabel Invoice .....                               | 48  |
| Tabel 3.13 Struktur Tabel Payments.....                               | 48  |
| Tabel 3.14 Tabel Komponen Input Output .....                          | 49  |
| Tabel 4.1 Tabel Kontrak API Error Validation.....                     | 67  |
| Tabel 4.2 Tabel Example Response API Error Validation .....           | 67  |
| Tabel 4.3 Tabel Kontrak API Response Error .....                      | 68  |
| Tabel 4.4 Tabel Example Response API Response Error.....              | 68  |
| Tabel 4.5 Tabel Kontrak API Response Success .....                    | 69  |
| Tabel 4.6 Tabel Example Response API Response Success .....           | 69  |
| Tabel 4.7 Tabel Kontrak API Response Success with data .....          | 70  |
| Tabel 4.8 Tabel Example Response API Response Success With Data ..... | 70  |
| Tabel 4.1 Hasil Pengujian Fitur Login.....                            | 94  |
| Tabel 4.2 Hasil Pengujian Proses Registrasi.....                      | 96  |
| Tabel 4.3 Hasil Pengujian API Pelatihan.....                          | 98  |
| Tabel 4.4 Hasil Pengujian Proses Pembelian Pelatihan .....            | 101 |

**DAFTAR LAMPIRAN**

|                                           |     |
|-------------------------------------------|-----|
| LAMPIRAN I DOKUMEN API DOCUMENTATION..... | 109 |
| LAMPIRAN II DOKUMEN HASIL PENGUJIAN ..... | 129 |

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Perkembangan teknologi informasi yang sangat cepat di era modern dan globalisasi memungkinkan berbagai aktifitas kegiatan dilakukan secara cepat dan efisien, tak terkecuali dalam hal pendidikan. Hal ini menyebabkan setiap instansi pendidikan dalam berbagai tingkatan harus bisa untuk melakukan proses digitalisasi dalam kegiatan belajar mengajar. Perkembangan teknologi informasi ini membawa paradigma baru dalam proses pembelajaran yang menyebabkan terjadinya kemajuan era dalam proses belajar mengajar yang menggunakan teknologi informasi sebagai wadah pembelajaran dan dapat dilakukan dari jarak jauh yang disebut *e-learning*.

*E-learning* adalah suatu alat pembelajaran yang menggunakan media elektronik untuk menyampaikan isi materi pembelajaran. *e-learning* merupakan salah satu strategi untuk memajukan pendidikan sehingga perlu diperluas dan dikembangkan. Pada dasarnya pengertian *e-learning* mempunyai makna yang sangat luas dan masih dipersepsikan secara berbeda-beda. Menurut Nuke L. Chusna (2019) *e-learning* merupakan kependekan dari *electronic learning*, yang mempunyai arti belajar dengan menggunakan elektronik, adapun pengertian dari elektronik adalah komputer atau internet. Konsep *e-learning* sendiri merupakan pengembangan dari sistem belajar *distance learning* atau pembelajaran jarak jauh. Jika *distance learning* tidak menitikberatkan pada pembelajaran berbasis teknologi informasi, maka *e-learning* lebih terfokus pada pembelajaran berbasis teknologi informasi.

Saat ini sudah banyak instansi maupun perusahaan yang memberikan pelatihan ataupun menjadikan *e-learning* sebagai bisnis komersial yang diperjual belikan untuk kalangan umum karena *e-learning* memiliki banyak manfaat. Salah satu manfaat *e-learning* menurut Sigit Haryanto (2018) adalah penggunaan *e-learning* memberi sejumlah kelebihan seperti: dapat meningkatkan prestasi belajar, mendapat respon positif, meningkatkan aktivitas belajar, dan dapat meningkatkan kemandirian belajar. Namun disisi lain *e-learning* memiliki beberapa kekurangan. Salah satunya adalah masih sangat terbatasnya *e-learning* yang dapat digunakan secara gratis oleh kalangan umum yang berperan sebagai pengajar atau pemberi materi.

PT Otak Kanan mencoba untuk menghadirkan sebuah platform *learning management system* bernama E-Course yang tujuan utamanya adalah dapat menghadirkan sebuah platform belajar digital yang dapat diakses oleh semua kalangan, baik sebagai peserta/siswa ataupun sebagai pengajar. Sehingga dengan melalui E-Course ini dapat mengatasi salah satu kekurangan yang dimiliki *system e-learning* saat ini yaitu terbatasnya akses untuk menjadi pengajar yang dapat membagikan materi secara gratis kepada masyarakat umum. Dan Kedepannya E-Course ini juga diharapkan dapat menjadi platform yang menjadi wadah dalam pelaksanaan studi independen yang diadakan oleh kementerian pendidikan dan kebudayaan pada batch yang akan datang.

Oleh karena itu pada tugas akhir ini akan *dikembangkan backend learning management system* yang meliputi proses pendaftaran, pembayaran, pelaksanaan pelatihan hingga proses pengelolaan data pelatihan oleh admin dan juga pengajar.

Dalam proses pengembangan *backend learning management system* E-Course akan dilakukan pemisahan untuk bagian *front-end* dan juga *back-end*, bagian *front-end* akan dikembangkan menggunakan react.js karena dapat menghasilkan website dengan tampilan yang bersifat *single page application* atau halaman website yang dapat diperbarui secara dinamis tanpa melakukan reload halaman secara keseluruhan, Sedangkan bagian *back-end* akan dikembangkan menggunakan Laravel 10. Kedua bagian ini akan disambungkan menggunakan REST-API yang memungkinkan komunikasi secara langsung antara *front-end* dan juga *back-end*.

Kemudian untuk dapat menghasilkan *backend learning management system* E-Course dengan proses pembayaran yang optimal pada tugas akhir ini juga mengimplementasikan *payment gateway* xendit, karena xendit menawarkan beberapa kelebihan diantaranya kemudahan integrasi dan juga dukungan untuk metode pembayaran lokal.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan diatas, maka rumusan masalah yang akan dibahas dari tugas akhir ini adalah bagaimana merancang dan membangun *backend learning management system* E-Course mulai dari proses pendaftaran, pelaksanaan pelatihan, hingga proses pengelolaan data pelatihan yang dapat menjadi sarana belajar mengajar efektif bagi peserta dan juga pengajar ?

## 1.3 Tujuan

Berdasarkan rumusan masalah yang telah diuraikan, maka tujuan dari proyek Tugas Akhir ini adalah merancang dan membangun *backend learning management System* E-Course mulai dari proses pendaftaran, pelaksanaan pelatihan, hingga

proses pengelolaan data pelatihan yang dapat menjadi sarana belajar mengajar yang efektif bagi peserta dan juga pengajar kemudian juga memberikan solusi terhadap permasalahan yang dibutuhkan *client*.

#### 1.4 Manfaat

Manfaat yang diharapkan dari pembuatan pengembangan *backend learning management System E-Course*:

1. Memberi kontribusi untuk membuat suatu sistem *Learning Management System E-Course* berbasis web bagi instansi ataupun individu yang membutuhkan.
2. Membantu instansi maupun individu untuk dapat melaksanakan proses belajar mengajar yang efektif melalui platform digital.
3. Membantu proses peningkatan mutu pendidikan melalui sebuah platform digital yang dapat digunakan oleh semua individu.

#### 1.5 Batasan Masalah

Adapun Batasan masalah dalam proses pembuatan *backend learning management System E-Course* adalah sebagai berikut:

1. Proses registrasi
2. Proses pendaftaran pelatihan melalui payment gateway xendit.
3. Proses pembelajaran
4. Proses pengelolaan data pelatihan
5. Proses pembuatan dashboard

## **BAB II**

### **ANALISIS SISTEM**

Analisis kebutuhan adalah metode untuk mengetahui perbedaan antara kondisi yang diinginkan/seharusnya (sould be / ought to be) atau diharapkan dengan kondisi yang ada saat ini (what is). Kondisi yang diinginkan sering kali disebut dengan kondisi ideal, sedangkan kondisi yang ada saat ini sering disebut dengan kondisi nyata. Berikut merupakan analisis kebutuhan yang meliputi bagian-bagian yang terlibat dalam sistem kerja dan dokumen-dokumen yang digunakan dalam sistem

#### **2.1 Profil Organisasi**

PT. Otak Kanan merupakan sebuah perusahaan yang bergerak di bidang industri kreatif yang berlokasi di Surabaya, didirikan pada tahun 2009, sesuai dengan akte pendirian perusahaan No. 01 tanggal 21 Desember 2009 oleh notaris Dicky Dwiarnanto, SH, HAM RI Nomor AHU-02110.AH.01.01.Tahun 2010 Tanggal 14 Januari 2010, Daftar Perseroan Nomor AHU-0003117.AH.01.09.Tahun 2010 Tanggal 14 Januari 2010 dengan nama PT. Otak Kanan.



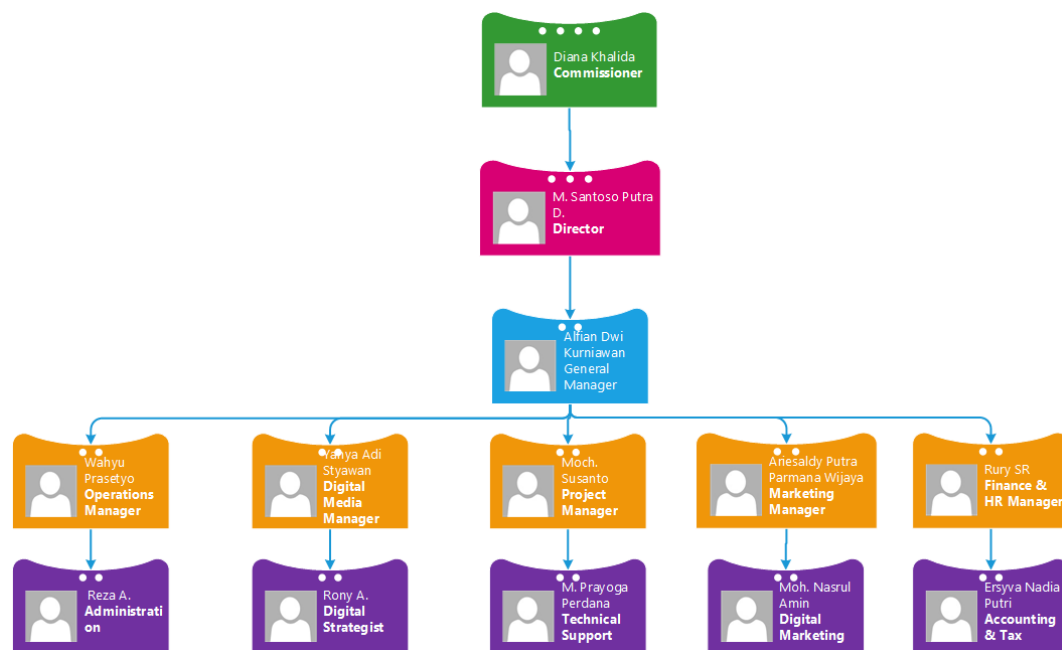
Gambar 2.1 Logo PT. Otak Kanan (Sumber : otakkanan.co.id, 2023)

Gambar 2.1 merupakan logo dari PT.Otak Kanan. PT Otak Kanan menciptakan, membangun dan menjalankan sebuah perusahaan yang terus-menerus dapat mendorong orang untuk menjadi kreatif dan inovatif dalam pemikiran

mereka. PT Otak Kanan mempunyai banyak ide yang telah teruji dan telah melayani lebih dari 100 perusahaan dari berbagai segmen pasar. PT Otak Kanan memahami bahwa setiap perusahaan itu unik, dengan kekuatan, kualitas dan pengalaman yang berbeda-beda. Sehingga PT Otak Kanan membahas setiap kesempatan dengan pendekatan kreatif. PT.Otak Kanan menyediakan beberapa jasa layanan digital marketing seperti Jasa Pembuatan Website, Jasa Pembuatan

### 2.1.1 Struktur Organisasi

Struktur organisasi dari PT. Otak Kanan dapat dilihat pada Gambar 2.2 :



Gambar 2.2 Struktur Organisasi PT. Otak Kanan

Deskripsi kerja pada struktur organisasi PT. Otak Kanan adalah sebagai berikut:

1. Commisioner merupakan orang yang memegang posisi tertinggi atas perusahaan yang bertanggung jawab atas semua keputusan atas perusahaan, fungsi korporasi, dan profitabilitas perusahaan. Dalam lingkup sistem ini,

commisioner tidak berinteraksi langsung dengan sistem namun hanya menjadi penanggung jawab dikembangkannya sistem E-Course ini.

2. Direktur bertugas untuk mengelola bisnis perusahaan termasuk pengembangan serta melakukan evaluasi terhadap kinerja karyawan di perusahaan yang dipimpin. Dalam lingkup sistem ini, direktur tidak berinteraksi langsung dengan sistem namun hanya menjadi penanggung jawab dikembangkannya sistem E-Course ini.
3. General Manager bertanggung jawab untuk merencanakan, mengimplementasikan, mengoordinasikan, memantau dan menganalisis semua kegiatan komersial perusahaan. Dalam lingkup sistem ini, general manager tidak berinteraksi langsung dengan sistem namun hanya menjadi penanggung jawab dikembangkannya sistem E-Course ini.
4. Operations Manager bertugas untuk mengawasi kegiatan operasional bisnis dan memastikan bahwa tenaga kerja produktif termasuk bekerja dengan staf senior lainnya dalam perencanaan strategis, memastikan bahwa semua staf mengetahui praktik terbaik operasional dan menerapkan teknologi baru. Dalam lingkup sistem ini, operation manager akan menjadi admin utama yang memegang peran untuk memverifikasi data pelatihan.
5. Digital Media Manager bertugas untuk dapat mengembangkan brand awarednes dan reputasi perusahaan dalam dunia digital.
6. Project Manager bertanggung jawab untuk merencanakan dan mengawasi proyek untuk memastikan untuk selesai tepat waktu dan sesuai anggaran. Manajer proyek merencanakan dan menetapkan sumber daya proyek,

menyiapkan anggaran, memantau kemajuan, dan terus memberi informasi kepada pemangku kepentingan.

7. Marketing Manager memiliki tugas untuk untuk mengembangkan, menerapkan, dan melaksanakan rencana pemasaran strategis untuk seluruh organisasi (atau lini bisnis dan merek dalam suatu organisasi) untuk menarik pelanggan potensial dan mempertahankan pelanggan yang sudah ada.
8. Finance & HR Manager memiliki tanggung jawab untuk penyusunan anggaran, mengelola catatan dan penerimaan serta mengembangkan dan melaksanakan strategi sumber daya manusia untuk mendukung rencana bisnis secara keseluruhan dan arah strategis organisasi, khususnya di manajemen talenta.

## **2.2 Sistem Kerja Saat Ini**

Pada saat ini PT. Otak Kanan menginginkan untuk dapat memiliki sebuah learning management sistem dengan sistem kerja yang merujuk pada kebanyakan program pelatihan konvensional sebagai berikut:

### **2.2.1 Bagian Yang Terlibat**

Bagian yang terlibat dalam proses pendaftaran pelatihan konvensional adalah sebagai berikut:

1. Staf Admin sebagai orang yang bertugas untuk memproses pendaftaran yang diajukan oleh calon peserta dan juga merencanakan pelaksanaan pelatihan sesuai dengan materi yang telah dibuat oleh pengajar.
2. Pengajar berperan dalam pembuatan materi pelatihan mulai dari kurikulum hingga materi yang akan dibagikan kepada peserta. Selain itu pengajar juga

bertugas untuk membuat soal ujian dan juga kunci jawaban yang nantinya akan dikerjakan oleh peserta.

3. Peserta, dalam hal ini calon peserta berperan untuk menghubungi staff admin untuk melakukan pendaftaran pelatihan yang dipilih. Kemudian peserta mengikuti pelatihan yang telah dibuat oleh admin hingga dapat menyelesaikan segala pelatihan dan juga ujian penilaian.

### **2.2.2 Dokumen Yang Terlibat**

Dokumen yang terlibat dalam pendaftaran program pelatihan konvensional adalah sebagai berikut:

1. Dokumen Permintaan Pendaftaran

Dokumen Permintaan Pendaftaran merupakan dokumen yang diserahkan oleh peserta kepada admin saat proses pendaftaran pelatihan.

2. Invoice Pembayaran

Invoice Pembayaran merupakan dokumen yang berisi detail mengenai nomor virtual akun dan juga besaran nominal yang harus dibayarkan dalam mendaftar sebuah pelatihan.

3. Bukti Pembayaran

Bukti pembayaran dalam hal ini adalah dokumen yang dapat menunjukkan bahwa calon peserta telah melakukan pembayaran yang valid melalui merchant yang sah. Dokumen ini dapat berbentuk bukti pembayaran langsung dari bank atau *screen shot* pembayaran melalui *mobile banking*.

#### 4. Dokumen Materi Pelatihan

Dokumen materi pelatihan merupakan dokumen berisi mengenai kurikulum dan juga materi – materi pelatihan yang dibuat oleh pemateri yang selanjutnya dibagikan kepada peserta pelatihan sebagai bahan pembelajaran.

#### 5. Lembar Soal

Lembar soal merupakan document yang dibuat oleh pengajar yang kemudian diserahkan kepada admin untuk kemudian dibagikan kepada peserta pelatihan pada saat pelaksanaan ujian.

#### 6. Lembar Jawaban

Lembar jawaban merupakan dokumen yang berisi jawaban peserta setelah peserta mengerjakan ujian. Setelah peserta mengerjakan ujian dokumen ini dikembalikan ke admin atau pemateri agar dilakukan proses koreksi jawaban untuk menentukan penilaian terhadap peserta penilaian.

#### 7. Sertifikat Pelatihan / Dokumen Penilaian

Sertifikat pelatihan merupakan dokumen akhir yang menjadi penanda bahwa peserta telah mengikuti dan lulus dalam sebuah pelatihan. Dokumen ini dibuat oleh admin dan kemudian diserahkan kepada peserta pada saat akhir pelatihan.

### 2.2.2 Proses Kerja Saat Ini

Proses kerja saat ini terdiri atas proses pendaftaran pelatihan, proses pelaksanaan pelatihan dan proses pengerjaan tes dalam pelatihan. Penjelasan dari masing-masing prosedur adalah sebagai berikut:

### 1. Proses Pendaftaran Pelatihan

Calon peserta yang ingin untuk mendaftar pelatihan menghubungi staff admin untuk menyatakan keinginannya dalam mendaftar pelatihan yang dipilih, kemudian staf admin akan membuatkan nomor virtual akun dan juga dokumen invoice yang akan digunakan oleh calon peserta untuk melakukan pembayaran. Setelah calon peserta mendapatkan dokumen invoice, calon peserta melakukan pembayaran melalui nomor virtual akun yang tertera pada dokumen invoice. Setelah calon peserta melakukan pembayaran, bukti bayar yang telah didapatkan akan dikirimkan kembali kepada staff admin sebagai bukti bahwa peserta telah melakukan proses pendaftaran yang valid, Setelah itu staff admin akan memberikan informasi lebih lanjut mengenai pelaksanaan pelatihan nantinya melalui pesan khusus.

Proses pendaftaran program pelatihan konvensional digambarkan dalam Document Flow Diagram seperti pada **Gambar 2.3**.

### 2. Proses Pelaksanaan Pelatihan

Pada kebanyakan pelatihan konvensional saat ini proses pelaksanaan pelatihan diadakan dengan jadwal dan waktu tertentu yang telah ditentukan oleh penyelenggara atau admin, sehingga proses ini hanya bisa diikuti oleh peserta tertentu. Dalam proses ini melibatkan dokumen materi pelatihan yang dibuat oleh pengajar yang dicetak dalam bentuk hard file ataupun soft file yang kemudian diberikan kepada admin atau penyelenggara dan dibagikan kepada seluruh peserta saat dilaksanakannya pelatihan.

Proses pelaksanaan program pelatihan konvensional digambarkan dalam Document Flow Diagram seperti pada **Gambar 2.4**

### 3. Proses Pengerjaan Tes

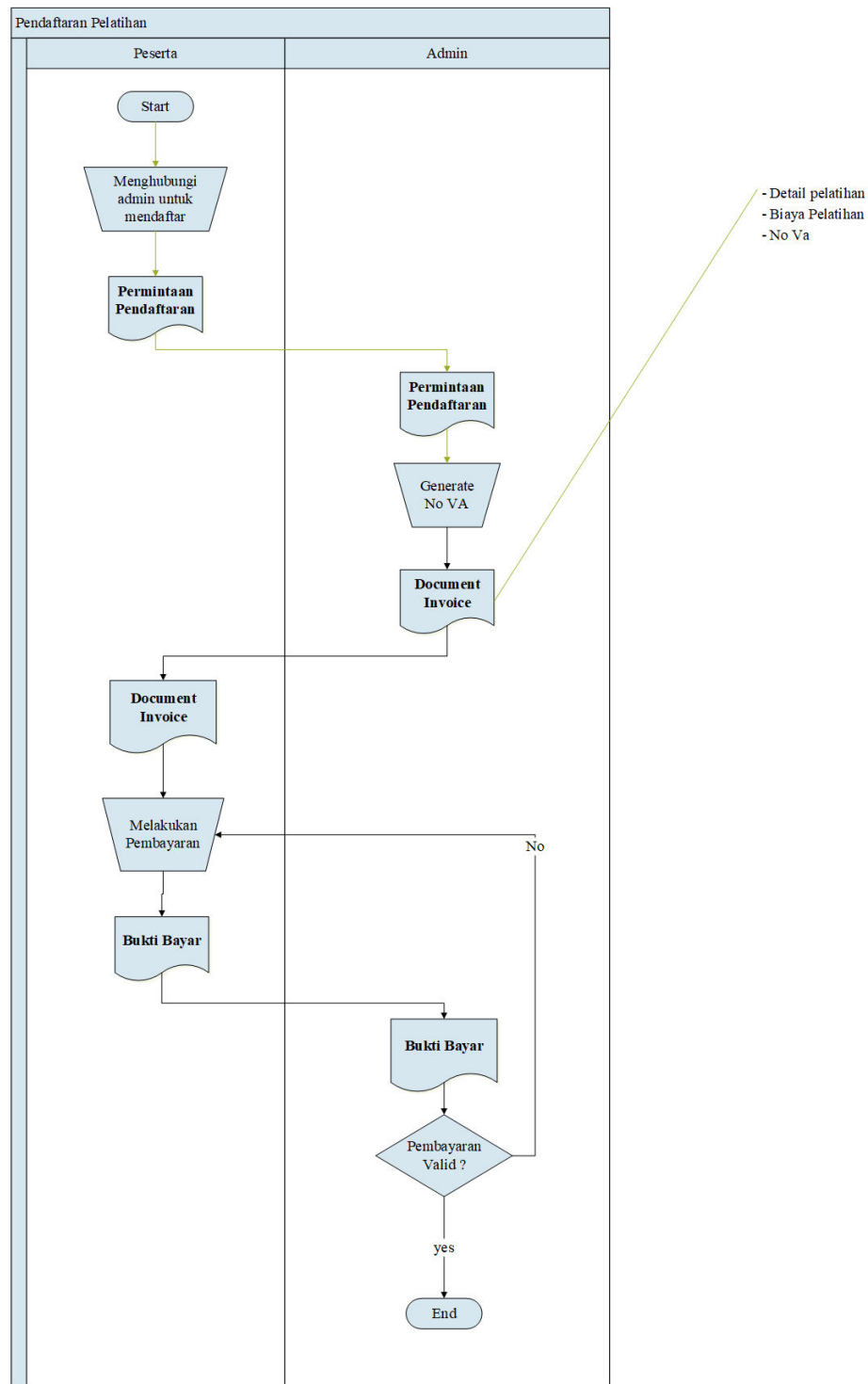
Proses pengerjaan tes ini tidak wajib dilakukan karena tidak semua pelatihan menjadikan nilai tes sebagai tolak ukur lulus atau tidaknya seseorang dalam mengikuti pelatihan. Proses pengerjaan tes diawali dengan pengajar membuat soal yang akan diujikan dan kemudian diberikan kepada admin untuk selanjutnya dapat dikerjakan oleh peserta.

Prosedur pengerjaan tes pelatihan konvensional digambarkan dalam Document Flow Diagram seperti pada **Gambar 2.5**.

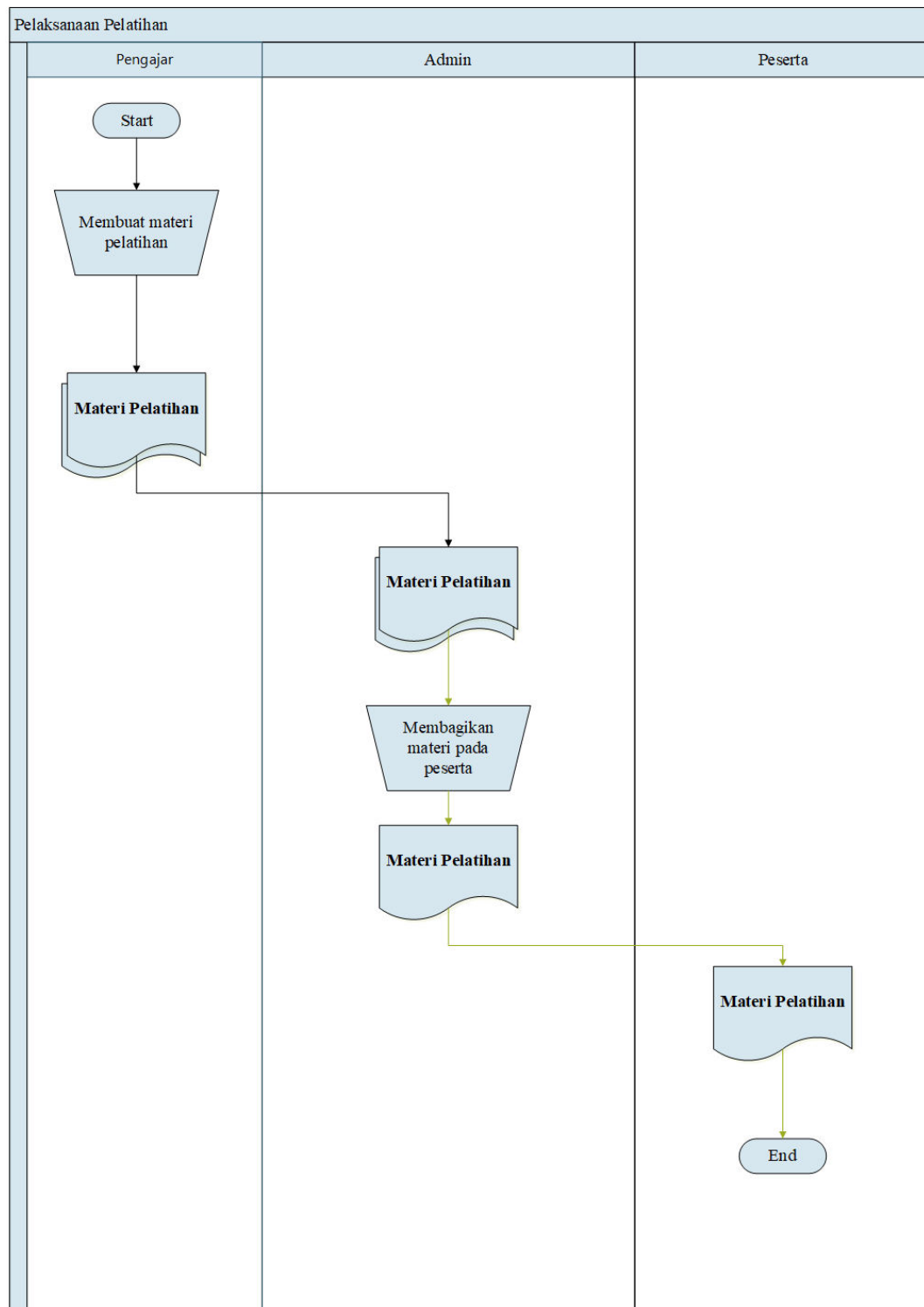
### 4. Proses Penilaian

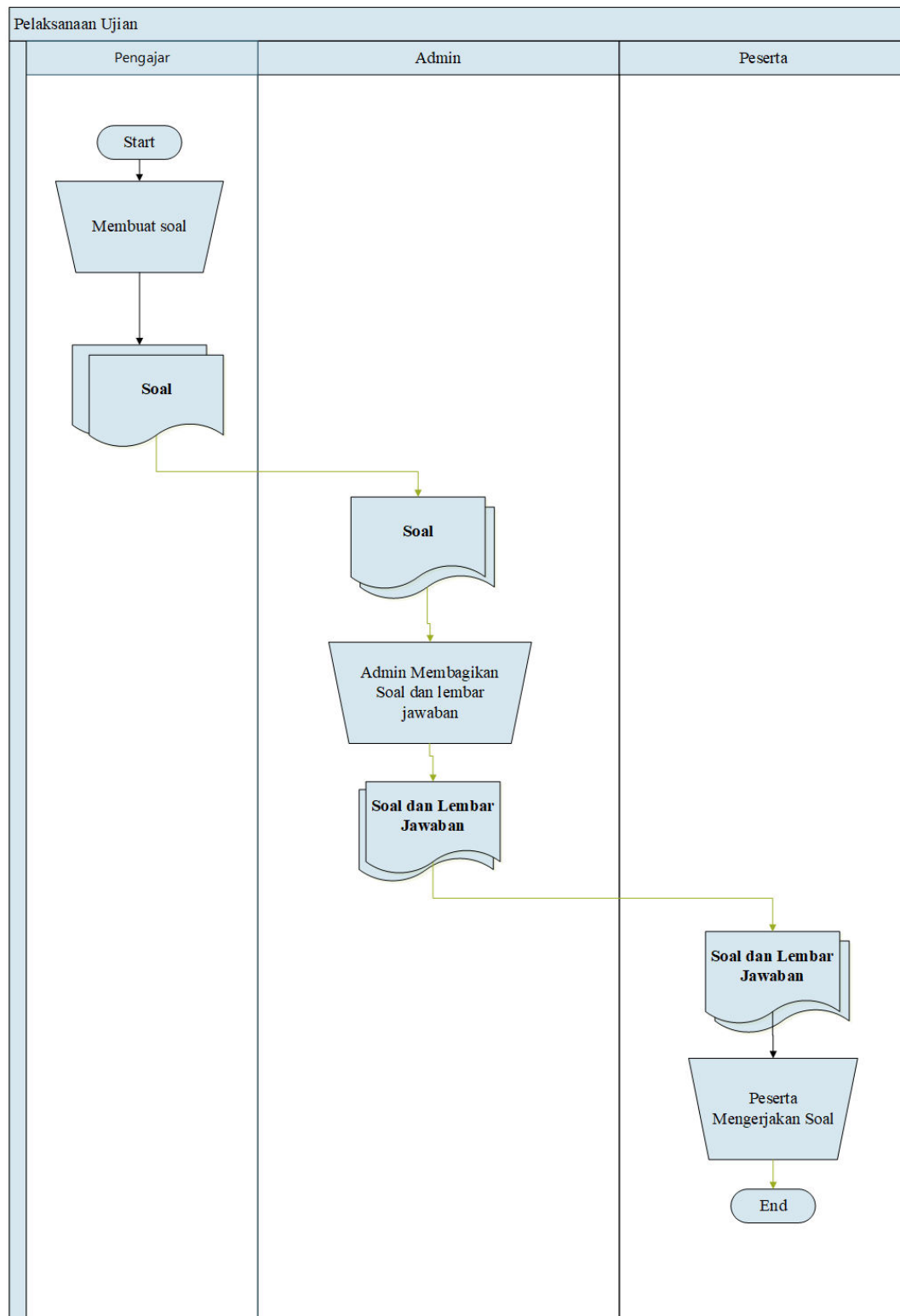
Proses penilaian tes dilakukan oleh peserta sebagai tahapan akhir dalam melakukan pelatihan jika dalam pelatihan tersebut menggunakan proses ujian sebagai tolak ukur kelulusan. Proses penilaian dilakukan oleh admin dengan mencocokkan jawaban yang dikerjakan oleh peserta dan juga kunci jawaban yang telah dibuat oleh pengajar. Setelah proses penilaian selesai admin akan membagikan dokumen penilaian atau sertifikat jika peserta mampu mendapatkan nilai diatas batas minimal yang telah ditentukan. Untuk pelatihan yang tidak menggunakan ujian sebagai standar kelulusan, standar kelulusan ditentukan dalam presentase kehadiran siswa dalam mengikuti pelatihan

Proses penilaian tes pelatihan konvensional digambarkan dalam Document Flow Diagram seperti pada **Gambar 2.6**.

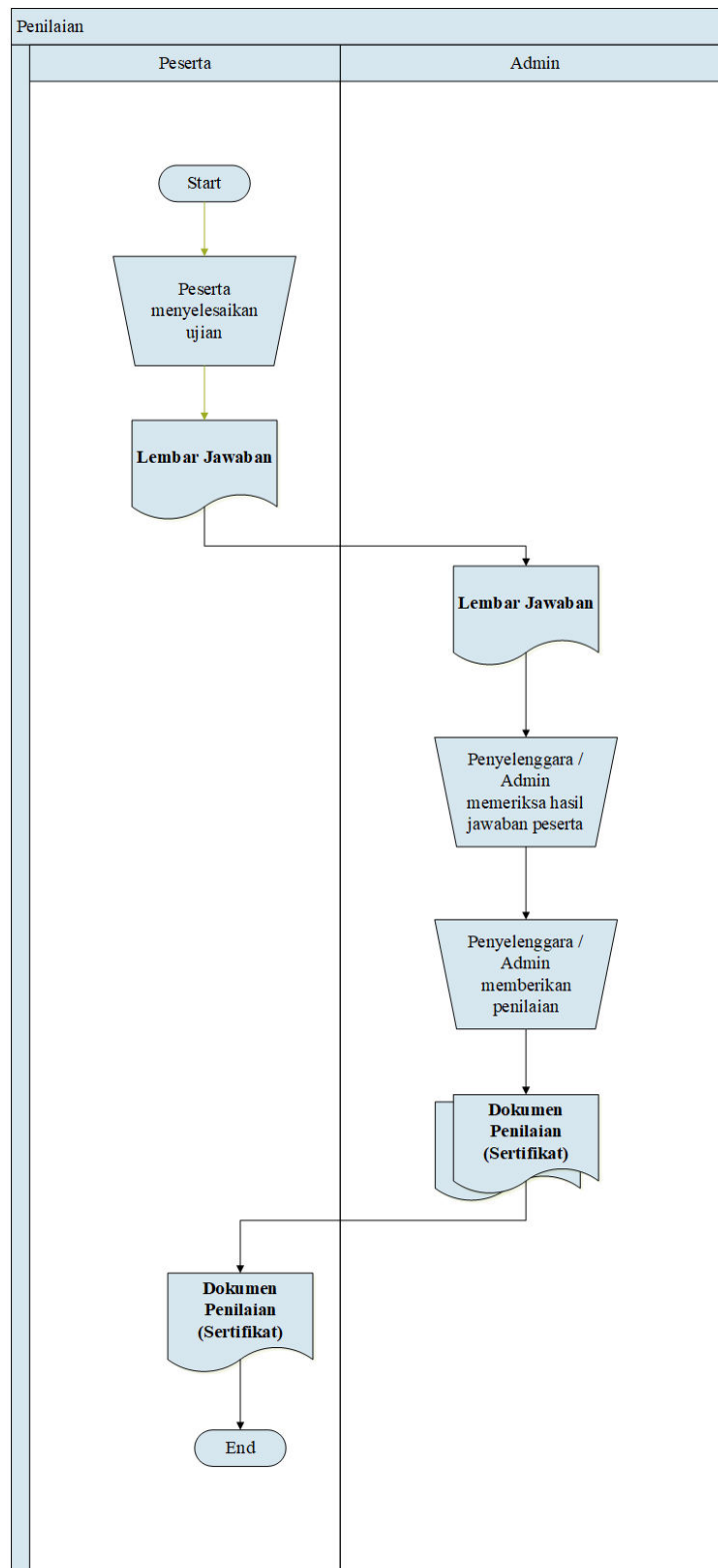


Gambar 2.3 *Document Flow Diagram* Pendaftaran Pelatihan  
Konvensional

Gambar 2.4 *Document Flow Diagram* Pelaksanaan Pelatihan



Gambar 2.5 Document Flow Diagram Pengerjaan Ujian



Gambar 2.6 Document Flow Diagram Penilaian

### 2.3 Analisis Permasalahan System

Berdasarkan analisis yang telah dilakukan, penulis dapat menyimpulkan bahwa pelaksanaan program pelatihan konvensional berjalan dengan kurang efektif dan kurang efisien. Hal ini disebabkan beberapa masalah yang telah digambarkan pada *fishbone* diagram pada **Gambar 2.7** yang meliputi:

#### 1. Proses

Permasalahan yang ada pada proses adalah sebagai berikut:

- Metode Pembayaran tidak efektif karena admin perlu untuk membuat nomor virtual akun pada *merchant* tertentu secara manual untuk diberikan kepada peserta pada saat mendaftar.
- Materi pelajaran terkadang tidak tuntas untuk diajarkan pada rentang waktu tertentu karena waktu yang terbatas.
- Metode pembelajaran yang dilaksanakan pada pelaksanaan pelatihan konvensional kurang menarik karena media yang digunakan terbatas dan hanya bisa dilaksanakan pada rentang waktu tertentu.
- Hanya dapat dilaksanakan pada tempat dan waktu yang tertentu karena hanya dapat dilakukan secara luring.

#### 2. SDM

Permasalahan yang ada pada SDM adalah sebagai berikut:

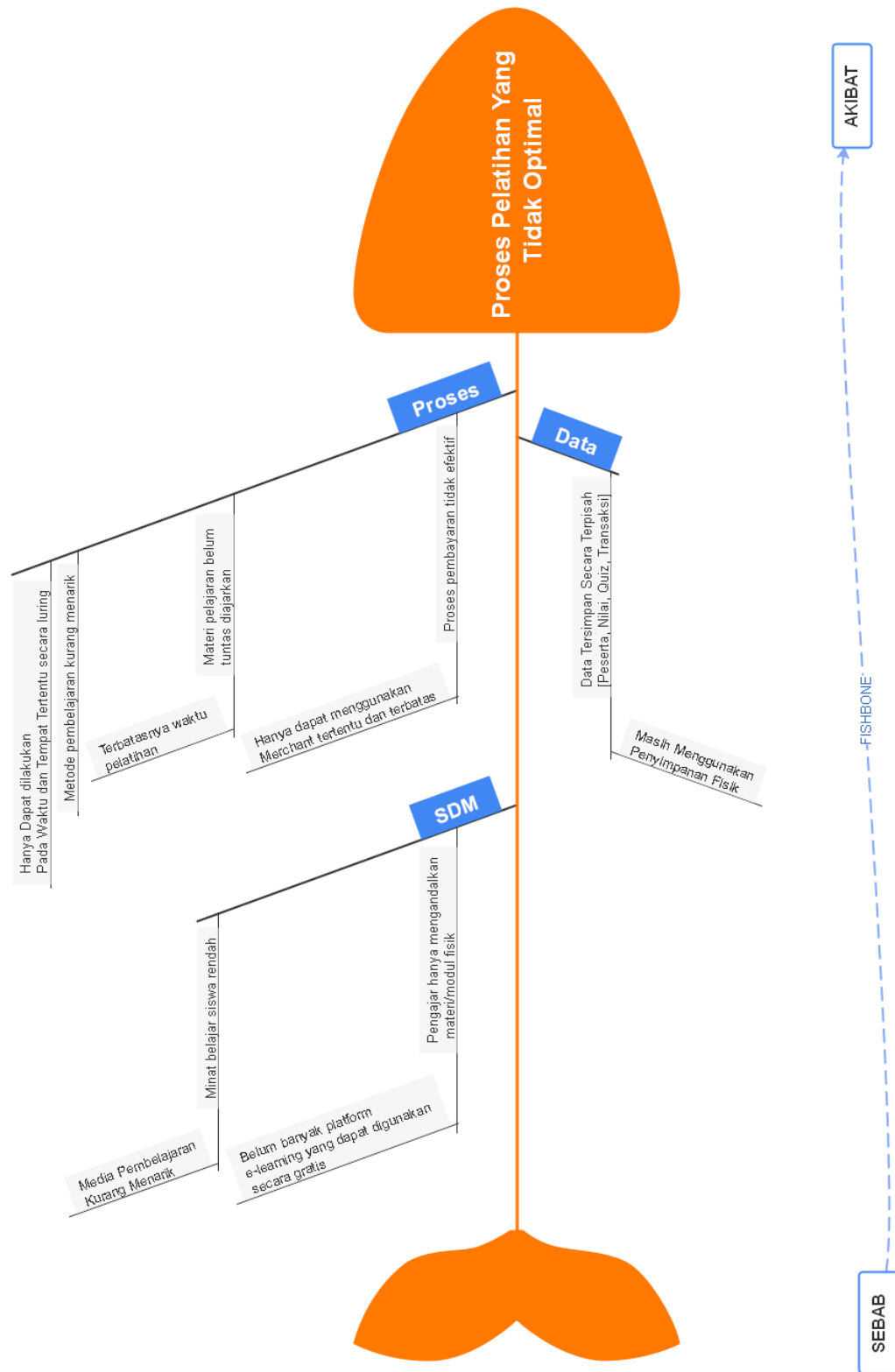
- Pada pelatihan konvensional pengajar hanya menggunakan modul fisik sebagai materi, sehingga materi yang digunakan terbatas dan tidak efisien.

- Peserta pelatihan memiliki minat untuk belajar yang rendah karena media pembelajaran yang dihadirkan oleh pengajar/mentor kurang menarik.

### 3. Data

Permasalahan yang ada pada data adalah sebagai berikut:

- Data pelatihan masih disimpan secara terbisa karena masih menggunakan penyimpanan fisik, seperti nilai quiz dan juga bukti pembayaran.



Gambar 2.7 Fishbone Diagram

## 2.4 kebutuhan Fungsionalitas Sistem

Kebutuhan fungsional yang dibutuhkan pada *Learning Management System E-Course* adalah sebagai berikut:

### 1. Fungsi registrasi

Fungsi registrasi adalah proses untuk membuat akun peserta dan juga pengajar agar dapat mengikuti proses belajar mengajar di e-course. Kemudian akan dilanjutkan dengan proses *autentifikasi* dan *authorisasi* yang berguna untuk membatasi hak akses pengguna pada sistem agar tidak terjadi pelanggaran akses terhadap fitur-fitur dan data tertentu pada sistem.

### 2. Fungsi pendaftaran pelatihan melalui *payment gateway* xendit.

Fungsi ini mencakup proses dimana peserta dapat membeli pelatihan yang diinginkan kemudian melakukan pembayaran pada merchant yang dipilih melalui *payment gateway* xendit.

### 3. Fungsi pembelajaran

Fungsi pembelajaran akan dilakukan oleh peserta ketika mengikuti suatu pelatihan. Dalam fungsi ini peserta dapat melihat materi dari pelatihan yang sudah dibeli dan mengetahui berapa presentase dari progress pelatihan yang diikuti.

### 4. Fungsi Pengelolaan Data Pelatihan

Fungsi pengelolaan data pelatihan ini dibuat untuk menampilkan data pelatihan beserta detail materi dimana setiap role memiliki akses yang berbeda beda dengan detail :

- Admin dapat melakukan verifikasi data pelatihan yang sudah dibuat oleh pengajar.
- Pengajar dapat membuat, mengedit dan menghapus data pelatihan.
- Pengajar dapat menambah, mengedit dan menghapus data materi pelatihan
- Peserta dapat melihat data pelatihan dan materi pelatihan yang telah dibuat pengajar dan telah diverifikasi admin.
- Peserta dapat menjalani pelatihan hingga selesai.

#### 5. Fungsi Dashboard

Fungsi dashboard adalah fungsi yang dapat diakses oleh admin dan juga pengajar. Fungsi utama dari dashboard ini adalah untuk dapat memantau berbagai data master dan juga transaksi seperti data pembayaran oleh peserta dan data jumlah peserta yang membeli pelatihan.

### **BAB III**

#### **DESAIN SISTEM**

Desain sistem merupakan tahap lanjutan dalam pengembangan sistem setelah proses analisis sistem. Desain adalah langkah yang penting dalam pembuatan sistem, karena langkah ini menentukan fondasi sistem informasi. Kesalahan dalam desain dapat menimbulkan hambatan bahkan kegagalan proyek.

#### **3.1 Sistem Kerja Baru**

Sistem kerja baru merupakan sistem yang akan dikembangkan untuk menggantikan sistem manual yang saat ini sedang berjalan. Sehingga pada sistem yang akan dibuat atau dirancang saat ini diharapkan dapat membantu menyelesaikan masalah yang telah diuraikan pada Bab I, sehingga proses belajar dan mengajar dapat lebih efektif dan efisien serta mengurangi resiko permasalahan data saat masih menggunakan cara manual.

##### **3.1.1 Proses Kerja Baru**

Proses kerja baru pada *Learning Management System* E-Course sebagai berikut:

##### **3.1.1.1 Proses Registrasi**

Proses registrasi merupakan proses yang hanya dilakukan oleh pengajar dan juga peserta. Proses ini dilakukan untuk membuat data user yang kemudian nantinya akan mendapatkan hak akses sesuai *role* yang dimiliki.

#### **3.1.1.2 Proses Pendaftaran Pelatihan Melalui *Payment Gateway Xendit*.**

Proses pendaftaran pelatihan melalui *payment gateway xendit*. yaitu proses yang menangani pendaftaran hingga pembayaran oleh pelajar yang ingin membeli pelatihan secara langsung melalui *payment gateway xendit*. Proses pendaftaran pelatihan melalui *payment gateway xendit* ini dibagi menjadi 3 sub-proses yaitu :

- Sub-proses yang pertama adalah peserta melakukan pembayaran pelatihan melalui metode pembayaran yang tersedia pada *payment gateway xendit*.

#### **3.1.1.3 Proses Pembelajaran**

Proses pembelajaran yaitu proses dilakukan oleh peserta ketika mengikuti suatu pelatihan. Dalam proses ini peserta dapat melihat materi dari pelatihan yang sudah dibeli dan mengetahui berapa presentase dari progress pelatihan yang diikuti.

#### **3.1.1.4 Proses Pengelolaan Data Pelatihan**

Proses pengelolaan data pelatihan merupakan proses yang dapat dilakukan oleh ketiga role baik admin, pengajar dan siswa. Proses pengelolaan data pelatihan ini dibagi menjadi 5 sub-proses yaitu :

- Sub-proses yang pertama adalah admin dapat melakukan verifikasi data pelatihan yang sudah dibuat oleh pengajar.
- Sub-proses yang ketiga adalah admin dapat membuat, mengedit, dan menghapus data kategori pelatihan.
- Sub-proses yang keempat adalah pengajar dapat membuat, mengedit dan menghapus data pelatihan yang dimiliki.
- Sub-proses yang kelima adalah pengajar dapat membuat, mengedit dan menghapus data kurikulum dan materi pelatihan yang dimiliki.

### 3.1.1.5 Proses Pemantauan Dashboard

Proses pemantauan penjualan pelatihan adalah proses yang dapat dilakukan oleh admin dan juga pengajar melalui halaman dashboard. Halaman dashboard berisikan ringkasan jumlah penjualan dan juga jumlah peserta dari pelatihan yang dimiliki.

### 3.1.2 Prosedur Kerja Baru

Prosedur kerja baru pada Learning Management System E-Course adalah sebagai berikut:

#### 3.1.2.1 Prosedur *Login* dan Registrasi

Prosedur baru pada proses *login* dijabarkan sebagai berikut:

- 1) Pengajar maupun siswa meng-akses halaman E-Course.
- 2) Sistem menampilkan halaman *landing page*.
- 3) Admin, pengajar maupun siswa meng-klik tombol *login* yang terdapat pada header.
- 4) Sistem menampilkan halaman *form login*.
- 5) Admin, Pengajar maupun siswa mengisi *username/email* dan *password*.
- 6) Jika data yang dimasukkan salah, sistem akan memunculkan *warning* bahwa data yang dimasukkan salah.
- 7) Jika data yang dimasukkan benar, sistem menampilkan halaman *dashboard* sesuai *role* yang dimiliki.

Prosedur Registrasi yang dilakukan oleh peserta atau pengajar sebagai berikut :

- 1) Pengajar maupun siswa meng-akses halaman E-Course.

- 2) Sistem menampilkan halaman landing page.
- 3) Pengajar maupun siswa meng-klik tombol registrasi yang terdapat pada header.
- 4) Sistem menampilkan halaman form registrasi.
- 5) Pengajar maupun siswa mengisi nama, username/email dan password.
- 6) Jika data yang dimasukkan tidak valid, sistem akan memunculkan warning bahwa data yang dimasukkan salah.
- 7) Jika data yang dimasukkan benar, sistem menampilkan halaman login.

#### **3.1.2.2 Prosedur Pendaftaran Pelatihan Melalui Payment Gateway Xendit.**

Prosedur pendaftaran yang dilakukan oleh peserta sebagai berikut :

- 1) Siswa meng-akses halaman E-Course.
- 2) Sistem menampilkan halaman landing page.
- 3) Siswa memilih pelatihan mana yang akan dibeli.
- 4) Sistem menampilkan detail pelatihan.
- 5) Siswa mengklik tombol beli sekarang
- 6) Jika siswa belum *login*, maka akan diarahkan ke halaman *login*
- 7) Jika siswa sudah *login*, sistem menampilkan halaman *checkout*.
- 8) Siswa mengklik tombol *checkout*
- 9) Sistem memproses *payment gateway xendit* untuk *generate link* pembayaran.
- 10) Sistem menampilkan halaman *dashboard* transaksi.
- 11) Siswa klik detail pembayaran
- 12) Sistem menampilkan tata cara pembayaran
- 13) Siswa memilih metode pembayaran.
- 14) Sistem akan menampilkan detail metode pembayaran yang dipilih

15) Siswa melakukan pembayaran.

16) Sistem akan secara otomatis melakukan update jika terdapat data pembayaran yang masuk, baik gagal ataupun berhasil.

Prosedur pemantauan data pendaftaran yang dilakukan oleh admin dan juga pengajar sebagai berikut :

- 1) Admin atau pengajar login ke halaman dashboard E-course
- 2) Admin atau pengajar mengakses halaman pembayaran untuk melihat data pembayaran yang dilakukan peserta.
- 3) Pengajar dapat menghapus data pembayaran yang tidak valid atau telah melewati batas waktu pembayaran

### **3.1.2.3 Prosedur Pembelajaran**

Prosedur pembelajaran oleh peserta antara lain sebagai berikut :

- 1) Siswa login untuk dapat mengakses halaman dashboard peserta.
- 2) Sistem menampilkan halaman dashboard.
- 3) Peserta mengakses halaman kursus saya.
- 4) Sistem akan menampilkan semua pelatihan yang dimiliki peserta
- 5) Siswa memilih pelatihan yang akan dipelajari.
- 6) Sistem akan menampilkan detail pelatihan hingga kurikulum dan materi pelatihan yang dipilih oleh peserta.

#### 3.1.2.4 Prosedur Pengelolaan Data Pelatihan

Prosedur verifikasi data pelatihan yang dilakukan oleh admin antara lain sebagai berikut :

- 1) Admin melakukan login ke halaman dashboard.
- 2) Admin memilih menu pengajuan kursus.
- 3) Admin memilih kursus yang akan di verifikasi.
- 4) Admin memeriksa detail materi kursus apakah layak untuk di verifikasi.
- 5) Admin menekan tombol verifikasi jika kursus dinilai layak.

Prosedur menolak pengajuan data pelatihan yang dilakukan oleh admin antara lain sebagai berikut :

- 1) Admin melakukan login ke halaman dashboard.
- 2) Admin memilih menu pengajuan kursus.
- 3) Admin memilih kursus yang akan di verifikasi.
- 4) Admin memeriksa detail materi kursus apakah layak untuk di verifikasi.
- 5) Admin menekan tombol *reject* jika kursus dinilai tidak layak.

Prosedur pembuatan data kategori yang dilakukan oleh admin antara lain sebagai berikut :

- 1) Admin melakukan login ke halaman dashboard.
- 2) Admin memilih menu kategori.
- 3) Admin menekan tombol tambah kategori.
- 4) Admin mengisi formulir data kategori
- 5) Admin menekan tombol simpan.

Prosedur pembuatan data pelatihan yang dilakukan oleh pengajar antara lain sebagai berikut :

- 1) Pengajar melakukan login ke halaman dashboard.
- 2) Pengajar memilih menu upload kursus.
- 3) Pengajar menekan tombol tambah kursus.
- 4) Pengajar mengisi formulir data kursus
- 5) Pengajar menekan tombol simpan.

Prosedur pembuatan data kurikulum serta materi pelatihan yang dilakukan oleh pengajar antara lain sebagai berikut :

- 1) Pengajar melakukan login ke halaman dashboard.
- 2) Pengajar memilih menu kursus saya.
- 3) Pengajar memilih kursus yang akan ditambah materinya.
- 4) Pengajar menekan tombol tambah kurikulum
- 5) Pengajar mengisi formulir data kurikulum dan juga materi
- 6) Pengajar menekan tombol simpan.

### **3.1.2.5 Prosedur Pemantauan Dashboard**

Prosedur pemantauan data penjualan oleh pengajar atau admin antara lain sebagai berikut :

- 1) Pengajar atau admin melakukan login ke halaman dashboard.
- 2) Sistem akan menampilkan ringkasan data penjualan pelatihan dan juga jumlah peserta.

### 3.2 Desain Proses

Proses adalah kumpulan aktivitas, tindakan, dan tugas yang dilakukan ketika akan membuat beberapa produk kerja. (Roger S. Pressman, Ph.D., 2010).

Sedangkan desain Proses merupakan tahap dalam merancang secara fisik proses-proses yang berkaitan dengan kebutuhan, dimana hasil dari desain proses ini adalah diagram-diagram yang saling berkaitan seperti Diagram Jenjang dan Data Flow Diagram (DFD).

#### 3.2.1 *Hierarchy Input Process Output (HIPO) Diagram*

Diagram jenjang pada E-Course ini memiliki 5 proses utama yaitu proses registrasi, proses pendaftaran pelatihan melalui *payment gateway* xendit, proses pembelajaran, proses pengelolaan data pelatihan dan proses pembuatan dashboard. Proses pengelolaan data pelatihan memiliki 5 sub proses yaitu proses input data kategori pelatihan oleh admin, proses input data pelatihan oleh pengajar, proses input data kurikulum oleh pengajar, proses input data materi oleh pengajar dan proses verifikasi pelatihan oleh admin.

Diagram jenjang untuk sistem tersebut dapat dilihat pada gambar 3.1



Gambar 3.1 *Hierarchy Input Process Output (HIPO) Diagram*

### 3.2.2 Data Flow Diagram (DFD)

Hubungan proses dan data dalam E-Course ini digambarkan dalam *Data Flow Diagram (DFD)*. *Data Flow Diagram (DFD)* adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kerja atau proses dilakukan dalam sistem tersebut. Istilah dalam bahasa Indonesianya adalah diagram aliran data (Indrajani, 2011).

#### 3.2.2.1 Context Diagram

*Context Diagram* merupakan tahap awal penggambaran DFD. Pada Level Context Diagram E-Course terdapat empat entitas yang terlibat pada sistem yang mempunyai otoritas yang berbeda yaitu:

1. Admin

Admin adalah orang yang mempunyai hak akses untuk dapat melihat semua data yang masuk, mulai data pengajar, data siswa, data pembayaran, data pelatihan hingga data materi. Admin juga dapat membuat kategori yang digunakan untuk mengelompokkan pelatihan yang dibuat oleh pengajar, dan yang terakhir admin dapat melakukan verifikasi data pelatihan yang telah di inputkan oleh pengajar.

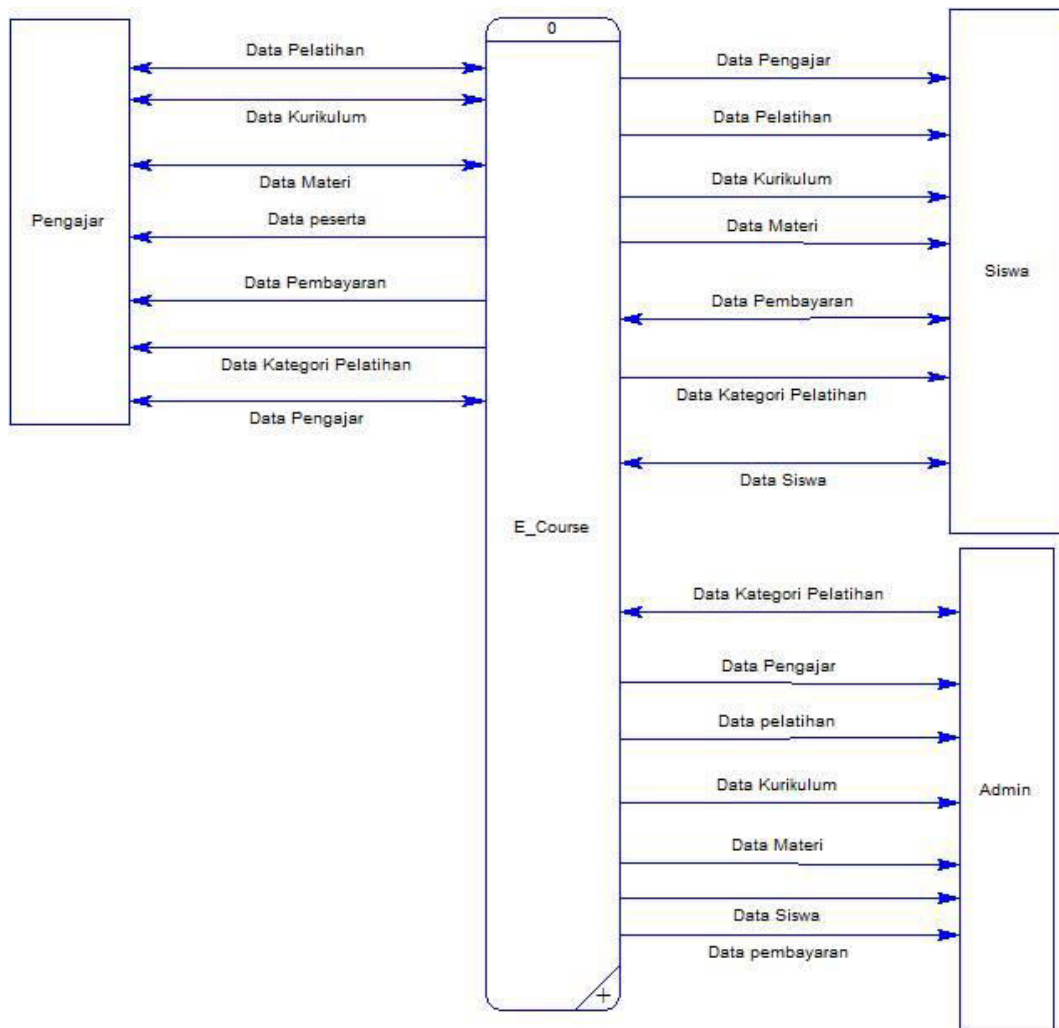
2. Pengajar

Pengajar adalah orang yang mempunyai hak akses membuat pelatihan, kurikulum, materi yang nantinya akan dapat dibeli oleh peserta. Selain itu pengajar juga dapat melihat data peserta yang membeli pelatihan hingga data pembayaran.

3. Peserta

Peserta adalah orang yang mempunyai hak akses untuk melihat pelatihan serta detailnya, membeli pelatihan, melaksanakan pelatihan hingga melihat progress pelatihan yang telah peserta ikuti.

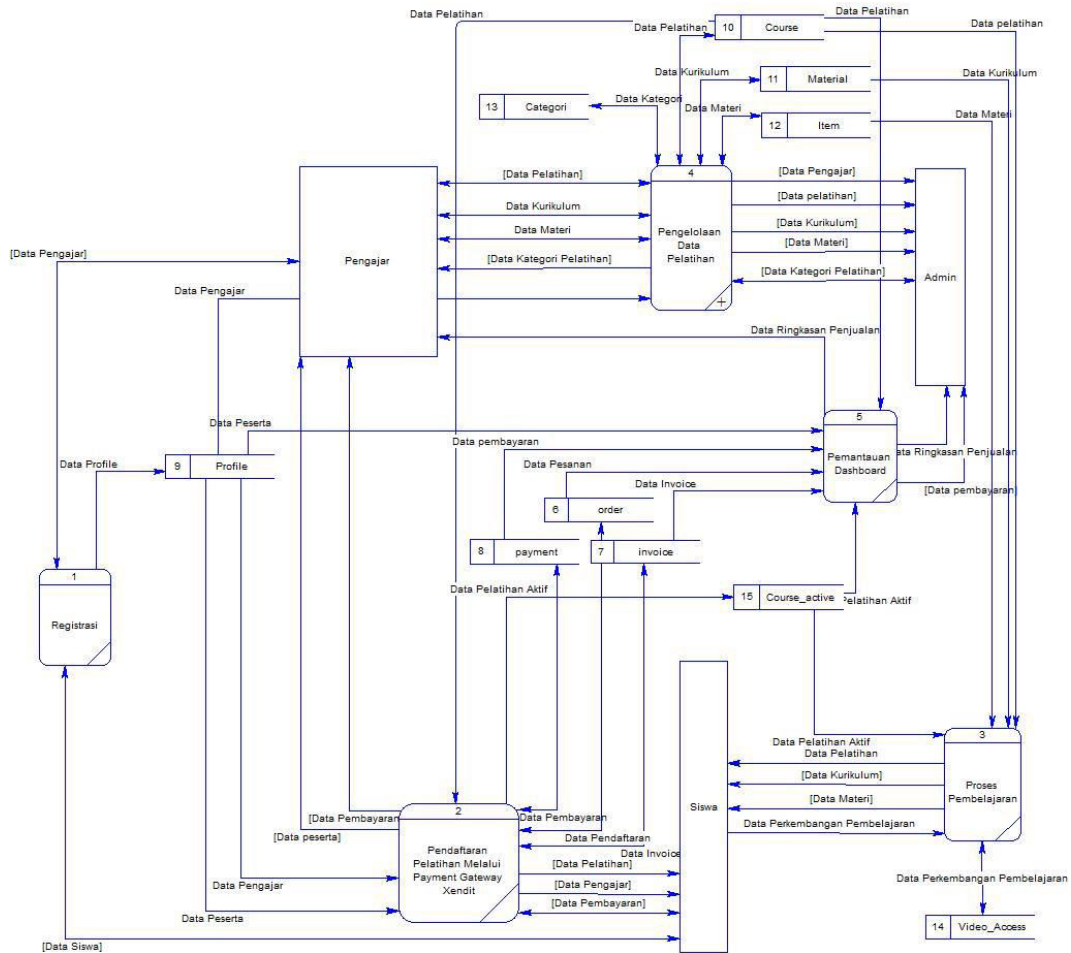
*Context Diagram* E-Course dapat dilihat pada gambar 3.2.



Gambar 3.2 Context Diagram

### 3.2.2.2 DFD Level 0

DFD *Level 0* adalah pengembangan dari context diagram terdapat lima proses pada DFD *Level 0* proses registrasi, proses pendaftaran pelatihan melalui *payment gateway* xendit, proses pembelajaran, proses pengelolaan data pelatihan, proses pemantauan dashboard. DFD *Level 0* E-Course dapat dilihat pada gambar 3.3.



### 3.2.2.3 DFD *Level 1* Proses Pengelolaan Data Pelatihan

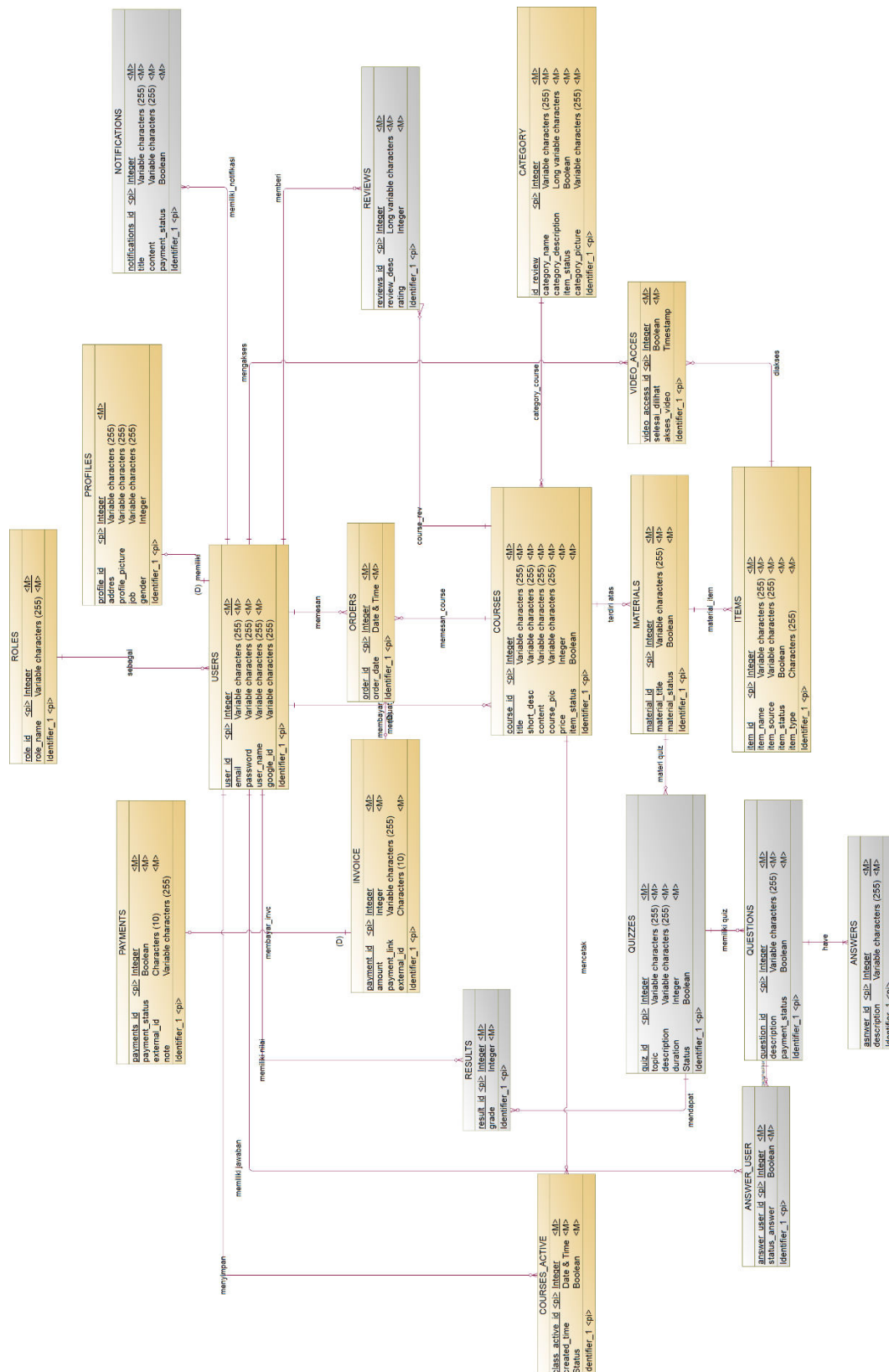
DFD *Level 1* dari proses pengelolaan data pelatihan memiliki 5 sub-proses yaitu proses pembuatan data kategori, pembuatan data pelatihan, pembuatan data kurikulum, pembuatan data materi, dan proses verifikasi pelatihan. DFD *Level 1* proses pengelolaan data pelatihan dapat dilihat pada gambar 3.4.



### 3.3.1 *Conceptual Data Model (CDM)*

Desain *database* dari E-Course ini disajikan dalam bentuk model logika yang digambarkan dengan menggunakan *Conceptual Data Model (CDM)*, yang digunakan untuk melakukan identifikasi entitas, atribut dan relasi antar entitas.

Untuk memberikan gambaran yang lebih jelas mengenai beberapa entitas pada E-Course dapat dilihat pada Gambar 3.5.



*Conceptual Data Model* (CDM) pada E-Course ini memiliki 19 entitas namun pada ruang lingkup pengerjaan tugas akhir ini hanya 12 entitas yang terlibat yaitu entitas role, entitas user, entitas profile, entitas orders, entitas invoice, entitas payment, entitas category, entitas course, entitas material, entitas items, entitas course\_active dan entitas video access..

Untuk keterangan lebih rinci dapat dilihat pada penjabaran berikut :

1. Entitas Role

Entitas role bereleasi dengan entitas user. Relasi role dengan user adalah one to many dengan mandatori terletak pada entitas role. Sehingga role dapat memiliki banyak user, dan user hanya wajib memiliki 1 role.

2. Entitas user

Entitas user memiliki relasi dengan beberapa entitas antara lain many to one dengan role yang berarti user hanya dapat memiliki satu role, one to one dengan profile dan mandatory berada di user yang berarti satu user hanya dapat memiliki satu profile dimana user belum tentu memiliki profile namun profile pasti memiliki user.

3. Entitas Profile

Entitas profile berelasi one to one dengan user dengan mandatori berada di user yang berarti satu user hanya dapat memiliki satu profile dimana user belum tentu memiliki profile namun profile pasti memiliki user.

4. Entitas *Category*

Entitas *category* berelasi one to many dengan *course* dimana mandatory berada pada sisi *category* yang berarti kategori dapat memiliki banyak *course* dan *course* wajib memiliki satu *category*.

5. Entitas *Course*

Entitas *course* memiliki beberapa relasi yaitu one to many dengan *category* dimana mandatory berada pada sisi *category* yang berarti kategori dapat memiliki banyak *course* dan *course* wajib memiliki satu *category*. One to many dengan *order* dan mandatory berada dalam *course* yang berarti *order* wajib memiliki satu *course* dan *course* dapat memiliki banyak *order*. One to many dengan *materials* dengan mandatory berada pada *course* yang berarti satu *course* dapat memiliki banyak *materials* dan *materials* wajib memiliki satu *course*. One to many dengan *course\_active* dan mandatory berada pada *course* yang berarti satu *course* dapat memiliki banyak *course\_active* dan *course\_active* wajib memiliki satu *course*.

6. Entitas *Materials*

Entitas *materials* berelasi one to many dengan *course* dan mandatory terletak pada *course* yang berarti satu *course* dapat memiliki banyak *materials* dan *materials* wajib memiliki satu *course*. Memiliki relasi one to many dengan *item* dan mandatory terletak pada *materials* yang berarti satu *materials* dapat memiliki banyak *item* dan *item* wajib memiliki satu *materials*.

7. Entitas *Item*

Entitas item memiliki relasi one to many dengan materials dimana mandatory terletak pada materials yang berarti materials yang berarti satu materials dapat memiliki banyak item dan item wajib memiliki satu materials. Item juga memiliki relasi one to many dengan video\_access dimana mandatory terletak pada item yang berarti satu item dapat memiliki banyak video\_access dan satu video\_access wajib memiliki 1 item.

8. Entitas video\_access

Entitas video\_access memiliki relasi one to many dengan item dimana posisi mandatory terletak pada item yang berarti satu item dapat memiliki banyak video\_access dan satu video\_access wajib memiliki 1 item. Video\_access juga memiliki relasi one to many dengan user dimana mandatory terletak pada user yang berarti user dapat memiliki banyak video\_access dan video\_access wajib memiliki satu user.

9. Entitas course\_active

Entitas course\_active memiliki relasi one to many dengan course dimana mandatory terletak pada course yang berarti course dapat memiliki banyak entitas course\_active dan course\_active wajib memiliki satu course. Entitas course\_active juga memiliki relasi one to many dengan user dimana mandatory terletak pada user yang berarti satu user dapat memiliki banyak course\_active dan course\_active wajib memiliki satu user.

10. Entitas Orders

Entitas orders memiliki relasi one to many dengan course dimana mandatory terletak pada course yang berarti satu course dapat memiliki banyak orders

dan satu orders wajib memiliki satu course. Entitas orders juga memiliki relasi one to one dengan invoice dimana mandatory terletak pada orders yang berarti satu invoice wajib memiliki satu orders.

11. Entitas Invoice

Entitas invoice memiliki relasi one to one dengan orders dimana mandatory dengan terletak pada orders yang berarti satu invoice wajib memiliki satu orders. Entitas invoice juga memiliki relasi one to one dengan payment dimana mandatory terletak pada invoice yang berarti satu payment wajib memiliki satu invoice.

12. Entitas Payments

Entitas payments memiliki relasi one to one dengan invoice dimana mandatory terletak pada invoice yang berarti satu payment wajib memiliki satu invoice.

### 3.3.2 *Physical Data Model (PDM)*

*Physical Data Model (PDM)* adalah bentuk representasi bentuk fisik dari desain data yang telah dibuat dengan mempertimbangkan perangkat lunak dan struktur *storage* yang akan digunakan untuk implementasi dari model. PDM merupakan hasil *generate* dari *Conceptual Data Model (CDM)* dengan atribut yang diperluas dan mendetail pada tiap tabelnya. Hasil generate tersebut menghasilkan objek yaitu :

Tabel 3.1 Tabel Objek pada *Physical Data Model* (PDM)

| Objek              | Keterangan                                                                                                                                |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| Tabel              | Menggambarkan sekumpulan data yang diatur dalam bentuk baris dan kolom yang merupakan pemodelan dari tabel basis data.                    |
| <i>Primary Key</i> | Suatu atribut yang sudah dipilih untuk mengidentifikasi baris secara unik yang berfungsi untuk mempermudah pengaturan dan perbaikan data. |
| <i>Foreign Key</i> | Merupakan kunci ( <i>key</i> ) pada suatu tabel yang terhubung dengan primary key pada tabel yang lain.                                   |
| <i>Reference</i>   | Merupakan hubungan antara <i>primary key</i> dan <i>foreign key</i> dari tabel yang berbeda.                                              |

Gambaran *Physical Data Model* (PDM) untuk E-Course dapat dilihat pada gambar 3.6.



*Entity* (entitas) didefinisikan sebagai sebuah kelas dari orang, tempat, obyek, kejadian, dan sebagainya atau sebuah konsep yang mana kita perlukan untuk menangkap dan menyimpan data (Jeffery L. Whitten et al, 2004: 295). *Entity* biasa disebut dengan tabel.

## TUGAS AKHIR

## 1. Tabel Role

Fungsi : Menyimpan data role atau hak akses setiap user

*Primary Key* : role\_id (*not null*)

*Foreign Key* : -

Tabel 3.2 Struktur Tabel Role

| No | Field Name | Data Type | Field Size | Keterangan         |
|----|------------|-----------|------------|--------------------|
| 1  | Role_id    | integer   | 20         | <i>Primary Key</i> |
| 2  | name       | varchar   | 255        |                    |

## 2. Tabel User

Fungsi : Menyimpan data pengguna E-Course

*Primary Key* : user\_id (*not null*)

*Foreign Key* : role\_id dari entitas role (*not null*)

Tabel 3. 3 Struktur Tabel User

| No | Field Name | Data Type | Field Size | Keterangan         |
|----|------------|-----------|------------|--------------------|
| 1  | User_id    | integer   | 20         | <i>Primary Key</i> |
| 2  | Role_id    | integer   | 20         | <i>Foreign Key</i> |
| 3  | Name       | varchar   | 255        |                    |
| 4  | email      | varchar   | 255        |                    |
| 5  | password   | varchar   | 255        |                    |

## 3. Tabel Profile

Fungsi : Menyimpan data profile setiap user

*Primary Key* : profile\_id (*not null*)

*Foreign Key* : user\_id dari entitas user (*not null*)

Tabel 3.4 Struktur Tabel Profile

| No | Field Name      | Data Type | Field Size | Keterangan  |
|----|-----------------|-----------|------------|-------------|
| 1  | Profile_id      | Integer   | 20         | Primary Key |
| 2  | address         | Varchar   | 255        | Foreign Key |
| 3  | Profile_picture | Varchar   | 255        |             |
| 4  | Job             | varchar   | 255        |             |
| 5  | gender          | Integer   |            |             |

4. Tabel *Category*

Fungsi : Menyimpan data Kategori pelatihan

Primary Key : category\_id (not null)

Foreign Key : -

Tabel 3.5 Struktur Tabel category

| No | Field Name           | Data Type | Field Size | Keterangan  |
|----|----------------------|-----------|------------|-------------|
| 1  | Category_id          | integer   | 20         | Primary Key |
| 2  | Category_name        | varchar   | 255        |             |
| 3  | Category_description | Varchar   | 255        |             |
| 4  | status               | boolean   |            |             |
| 5  | Category_picture     | varchar   | 255        |             |

5. Tabel *Course*

Fungsi : Menyimpan data pelatihan

Primary Key : course\_id (not null)

Foreign Key : category\_id dari entitas category, user\_id dari entitas user

Tabel 3.6 Struktur Tabel Course

| No | <i>Field Name</i> | <i>Data Type</i> | <i>Field Size</i> | Keterangan                                                  |
|----|-------------------|------------------|-------------------|-------------------------------------------------------------|
| 1  | Course_id         | integer          | 20                | <i>Primary Key</i>                                          |
| 2  | Category_id       | integer          | 20                | <i>Foreign Key</i>                                          |
| 3  | User_id           | integer          | 20                | <i>Foreign Key</i>                                          |
| 4  | title             | varchar          | 255               |                                                             |
| 5  | Short_desc        | varchar          | 255               |                                                             |
| 6  | content           | varchar          | 255               |                                                             |
| 7  | Course_picture    | varchar          | 255               |                                                             |
| 8  | price             | integer          | 20                |                                                             |
| 9  | status            | enum             |                   | ('unprocessed',<br>'pending',<br>'accepted',<br>'rejected') |

## 6. Tabel Materials

Fungsi : Menyimpan data kurikulum pelatihan

*Primary Key* : material\_id (*not null*)

*Foreign Key* : course\_id dari entitas course

Tabel 3.7 Struktur Tabel Materials

| No | <i>Field Name</i> | <i>Data Type</i> | <i>Field Size</i> | Keterangan         |
|----|-------------------|------------------|-------------------|--------------------|
| 1  | Materials_id      | integer          | 20                | <i>Primary Key</i> |
| 2  | Course_id         | integer          | 20                | <i>Foreign Key</i> |
| 3  | title             | Varchar          | 255               |                    |

## 7. Tabel Item

Fungsi : Menyimpan data materi dari setiap pelatihan

*Primary Key* : item\_id (*not null*)

*Foreign Key* : material\_id dari entitas materials

Tabel 3.8 Struktur Tabel Item

| No | Field Name  | Data Type | Field Size | Keterangan  |
|----|-------------|-----------|------------|-------------|
| 1  | Item_id     | Integer   | 20         | Primary Key |
| 2  | Material_id | Integer   | 20         | Foreign Key |
| 3  | name        | varchar   | 255        |             |
| 4  | Source      | varchar   | 255        |             |

8. Tabel Video\_acces

Fungsi : Menyimpan data perkembangan pembelajaran siswa/peserta

*Primary Key* : video\_access\_id (not null)

*Foreign Key* : user\_id dari entitas user dan item\_id dari entitas item

Tabel 3.9 Struktur Tabel video\_access

| No | Field Name      | Data Type | Field Size | Keterangan  |
|----|-----------------|-----------|------------|-------------|
| 1  | Video_access_id | integer   | 20         | Primary Key |
| 2  | User_id         | integer   | 20         | Foreign Key |
| 3  | Item_id         | integer   | 20         | Foreign Key |
| 4  | Akses_video     | timestamp |            |             |
| 5  | Selesai_dilihat | boolean   |            |             |

9. Tabel Course\_active

Fungsi : Menyimpan data pelatihan yang sudah dibeli oleh siswa/peserta

*Primary Key* : course\_active\_id (not null)

*Foreign Key* : user\_id dari entitas user (*not null*) dan course\_id dari entitas course (*not null*)

Tabel 3.10 Struktur Tabel Course\_active

| No | Field Name       | Data Type | Field Size | Keterangan             |
|----|------------------|-----------|------------|------------------------|
| 1  | Course_active_id | integer   | 20         | Primary Key            |
| 2  | User_id          | integer   | 20         | Foreign Key            |
| 3  | Course_id        | integer   | 20         | Foreign Key            |
| 4  | status           | enum      |            | ('in progres', 'done') |

## 10. Tabel Orders

Fungsi : Menyimpan data history pembelian pelatihan oleh siswa

*Primary Key* : orders\_id (*not null*)

*Foreign Key* : user\_id dari entitas user id (*not null*) dan course\_id dari entitas course id (*not null*)

Tabel 3.11 Struktur Tabel Orders

| No | Field Name | Data Type | Field Size | Keterangan  |
|----|------------|-----------|------------|-------------|
| 1  | Order_id   | integer   | 20         | Primary Key |
| 2  | User_id    | integer   | 20         | Foreign Key |
| 3  | Course_id  | integer   | 2          | Foreign Key |
| 4  | Order_date | timestamp |            |             |

## 11. Tabel Invoice

Fungsi : Menyimpan data detail nominal pembelian pelatihan

*Primary Key* : invoice\_id (*not null*)

*Foreign Key* : order\_id (*not null*)

Tabel 3.12 Struktur Tabel Invoice

| No | Field Name   | Data Type | Field Size | Keterangan  |
|----|--------------|-----------|------------|-------------|
| 1  | Invoice_id   | integer   | 20         | Primary Key |
| 2  | Order_id     | integer   | 20         | Foreign Key |
| 3  | amount       | integer   | 20         |             |
| 4  | Payment_link | varchar   | 255        |             |
| 5  | External_id  | varchar   | 255        |             |

## 12. Tabel Payments

Fungsi : Menyimpan data pembayaran pelatihan

Primary Key : payment\_id (not null)

Foreign Key : invoice\_id (not null)

Tabel 3.13 Struktur Tabel Payments

| No | Field Name  | Data Type | Field Size | Keterangan  |
|----|-------------|-----------|------------|-------------|
| 1  | Payment_id  | integer   | 20         | Primary Key |
| 2  | Order_id    | Integer   | 20         | Foreign Key |
| 3  | status      | varchar   | 255        |             |
| 4  | External_id | varchar   | 255        |             |

### 3.4 Desain Antarmuka

Antarmuka dalam sebuah sistem informasi bertujuan untuk memudahkan penggunaan sistem sehingga nantinya proses yang berjalan menjadi lebih efektif. Ruang lingkup pada pengerjaan tugas akhir ini tidak mencakup desain antarmuka sistem e-course, Sehingga desain antarmuka yang tertera pada bagian ini bukan merupakan hasil dari penulis. Desain antarmuka pada sistem e-course dijelaskan sebagai berikut :

### 3.4.1 Desain *Input*

Desain *Input* adalah desain yang digunakan untuk menerima masukan data yang akan disimpan dalam database dan nantinya akan digunakan ke dalam proses transaksi maupun sebagai *Output* yang dilaporkan. (Jogiyanto, 2001)

Desain *Input* terdiri dari desain form utama dan form pendukung. Desain *Input* terdiri dari beberapa komponen yang dapat dilihat pada tabel 3.24.

Tabel 3.14 Tabel Komponen *Input Output*

| Komponen                                                                                         | Keterangan                                                                                                                                                                                        |
|--------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | <i>Textbox</i> , digunakan sebagai <i>Inputan</i> data.                                                                                                                                           |
|               | <i>Grid</i> , digunakan untuk menampilkan seluruh data.                                                                                                                                           |
|  Radio Button | <i>Radio Button</i> , digunakan untuk memberikan pilihan dan harus dipilih salah satu.                                                                                                            |
|               | <i>Button</i> , digunakan sebagai tombol yang berfungsi sebagai suatu proses, misalnya proses menyimpan, mengubah dan sejenisnya atau digunakan sebagai tombol untuk pindah ke halaman yang lain. |

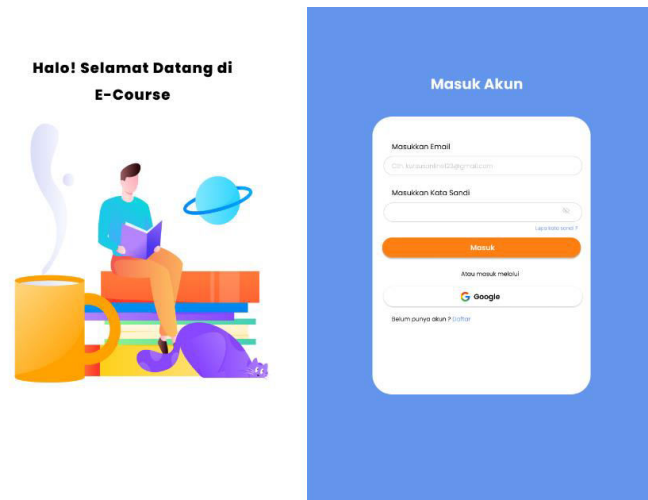
Berikut ini adalah desain *Input* dari Sistem E-Course.

#### 3.4.1.1 Form *Login*

Form *Login* merupakan form yang terdapat pada halaman awal Sistem E-Course, pada form ini user memasukkan username dan password untuk dapat masuk ke dalam sistem sesuai dengan otorisasi masing – masing user. Adapun beberapa hak akses adalah sebagai berikut :

- a. Admin bertanggung jawab atas pengelolaan data kategori pelatihan, verifikasi pelatihan dan memantau proses transaksi penjualan pelatihan melalui dashboard .
- b. Pengajar bertanggung jawab atas pengelolaan data pelatihan, pengelolaan data kurikulum dan pengelolaan data materi pelatihan. Pengajar juga memiliki hak untuk dapat memantau proses pembelian yang dilakukan peserta/siswa.
- c. Peserta/Siswa bertanggung jawab atas proses pembelajaran dan pembelian pelatihan.

Desain Form *Login* dapat dilihat pada gambar 3.7.



Gambar 3.7 Desain Form *Login*

#### 3.4.1.2 Form registrasi

Pada halaman registrasi berfungsi untuk memasukkan inputan email dan kata sandi untuk user. Desain Form *registrasi* dapat dilihat pada gambar 3.8.

**Ayo Bergabung Bersama  
E-Course**



**Daftar Akun**

Masukkan Email

Cth. kursusonline123@gmail.com

Masukkan Kata Sandi

Masukkan Ulang Kata Sandi

**Daftar**

Atau daftar melalui

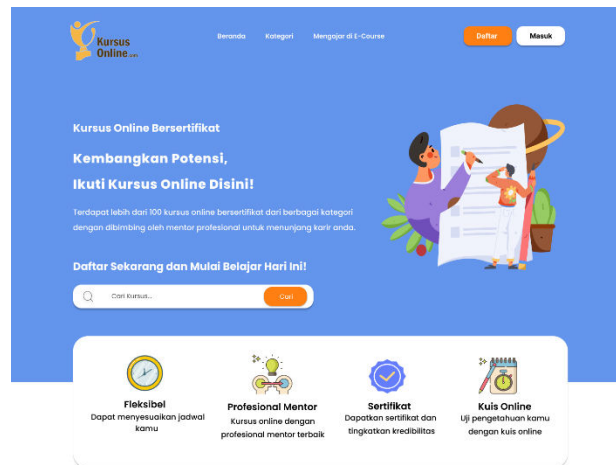
 **Google**

Sudah punya akun? [Masuk](#)

Gambar 3.8 Desain Halaman Form Registrasi.

### 3.4.1.3 Halaman Input Untuk Mencari Pelatihan

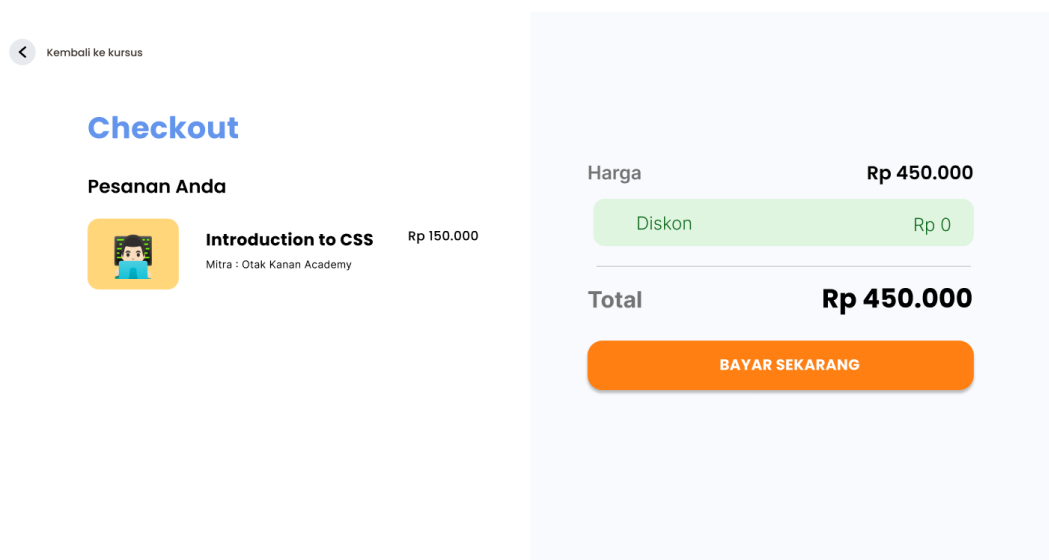
Pada halaman ini, peserta/siswa dapat memasukkan kata kunci yang digunakan mencari pelatihan yang sesuai. Desain form input pencarian pelatihan dapat dilihat pada gambar 3.9.



Gambar 3.9 Desain Form Pencarian Pelatihan

#### 3.4.1.4 Form Tambah Data Pembayaran

Pada halaman ini, Peserta/siswa dapat memasukkan data pembayaran secara otomatis Ketika menekan tombol checkout yang kemudian akan diproses oleh *payment gateway xendit*. Desain form tambah tambah data pembayaran dilihat pada gambar 3.10.



Gambar 3.10 Desain halaman Tambah Pembayaran

### 3.4.1.5 Form Tambah Data Profile

Pada halaman ini, Peserta/siswa dapat menambahkan atau mengedit data profile yang dimiliki. Desain form tambah data profile dapat dilihat pada gambar 3.11.

The screenshot displays the 'Form Tambah Data Profile' in the E-Course application. The top navigation bar is blue with the 'E Course.' logo and links for 'Beranda', 'Kategori', and 'Mengajar di E-Course'. A user profile icon and name 'Putra Wirawan' are visible in the top right. Below the navigation bar, a white box contains a welcome message: 'Selamat Datang, Putra Wirawan!' with the email 'putrawirawan@gmail.com'. The main content area is divided into two sections. On the left, a 'Navigasi' sidebar lists 'Dashboard', 'Kursus saya', 'Sertifikat', 'Transaksi', and 'Pengaturan' (highlighted). On the right, the 'Profile Saya' form is shown, featuring a profile picture placeholder and input fields for 'Nama Lengkap' (Putra Wirawan), 'Email' (putrawirawan@gmail.com), 'Alamat' (Jl. Ahmad Yani No. 18, Surabaya, Jawa Timur), 'Jenis Kelamin' (radio buttons for 'Laki-laki' and 'Perempuan'), and 'Pekerjaan' (Mahasiswa). At the bottom right of the form are 'Batal' and 'Simpan' buttons.

Gambar 3.11 Desain Form Tambah Data Profile.

### 3.4.1.6 Form Tambah Data Pelatihan

Pada halaman ini, Pengajar dapat menambahkan data pelatihan yang dimiliki. Desain form tambah data pelatihan dapat dilihat pada gambar 3.12.

The screenshot shows a web interface for uploading a course. On the left is a sidebar with a logo 'E Course.' and a menu: Dashboard, Kursus saya, Upload kursus (highlighted), Buat Quiz, Invoice, and a settings icon. The main area is titled 'Upload Kursus' and shows the user 'Mark Zuckerberg'. Below this is the 'Informasi Dasar' section with the instruction 'Tambahkan lebih detail tentang kursus ini'. The form contains the following fields:

- Judul Kursus:** A text input field with the placeholder 'Ketikkan judul kursus disini'.
- Deskripsi Singkat:** A text input field with the placeholder 'Ketikkan intro kursus disini'.
- Deskripsi:** A larger text input field with the placeholder 'Ketikkan jdeskripsi kursus disini'.
- Harga Kursus:** A text input field with a 'Rp.' prefix and the placeholder 'Rp.'.
- Upload Cover:** A large rectangular area with a central upload icon and the text 'Klik untuk mengunggah atau seret dan lepas .JPG, .JPEG, .PNG'.

A blue 'Submit' button is located at the bottom right of the form.

Gambar 3.12 Desain Form Tambah Data Pelatihan

### 3.4.1.7 Form Tambah Data Kurikulum dan Materi Pelatihan

Pada halaman ini, Pengajar setelah melakukan upload kursus, pengajar akan menambahkan data kurikulum serta materi pelatihan. Desain form tambah data kurikulum dan materi pelatihan dapat dilihat pada gambar 3.13.

The screenshot shows a web interface for uploading course curriculum. The main heading is 'Kurikulum Kursus' with a subtext 'Tambahkan lebih detail tentang kursus ini'. The form consists of several input fields: 'Judul Kurikulum', 'Judul item', 'Import Video Materi dari URL', and 'File lampiran PDF (opsional)'. A 'Browse' button is located next to the PDF field. At the bottom of the form is a large blue button labeled '+ Tambah Item Kurikulum'. A 'Submit' button is positioned at the bottom right. The left sidebar contains navigation links: 'Dashboard', 'Kursus saya', 'Upload kursus', 'Buat Quiz', and 'Invoice'. The top right corner shows a user profile for 'Mark Zuckerberg'.

Gambar 3.13 Desain Form Tambah Data Kurikulum dan Materi Pelatihan

### 3.4.2 Desain Output

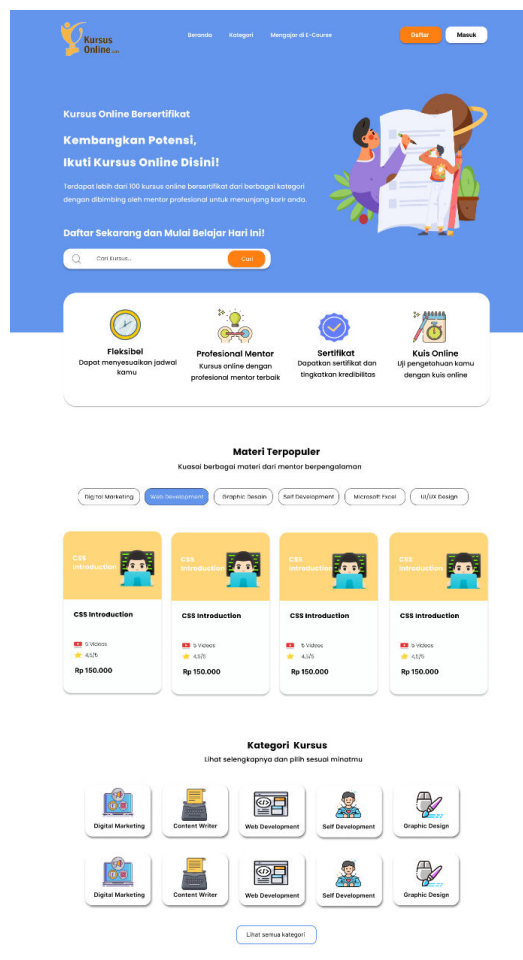
*Output* adalah tampilan / keluaran hasil dari sebuah masukan pada sistem.

Berikut ini adalah desain *Output* dari Sistem E-Course sebagai berikut.

#### 3.4.2.1 Halaman *Landing Page*

Pada halaman ini, sistem akan menampilkan daftar pelatihan yang telah diverifikasi oleh admin. Desain halaman *landing page* dapat dilihat pada gambar

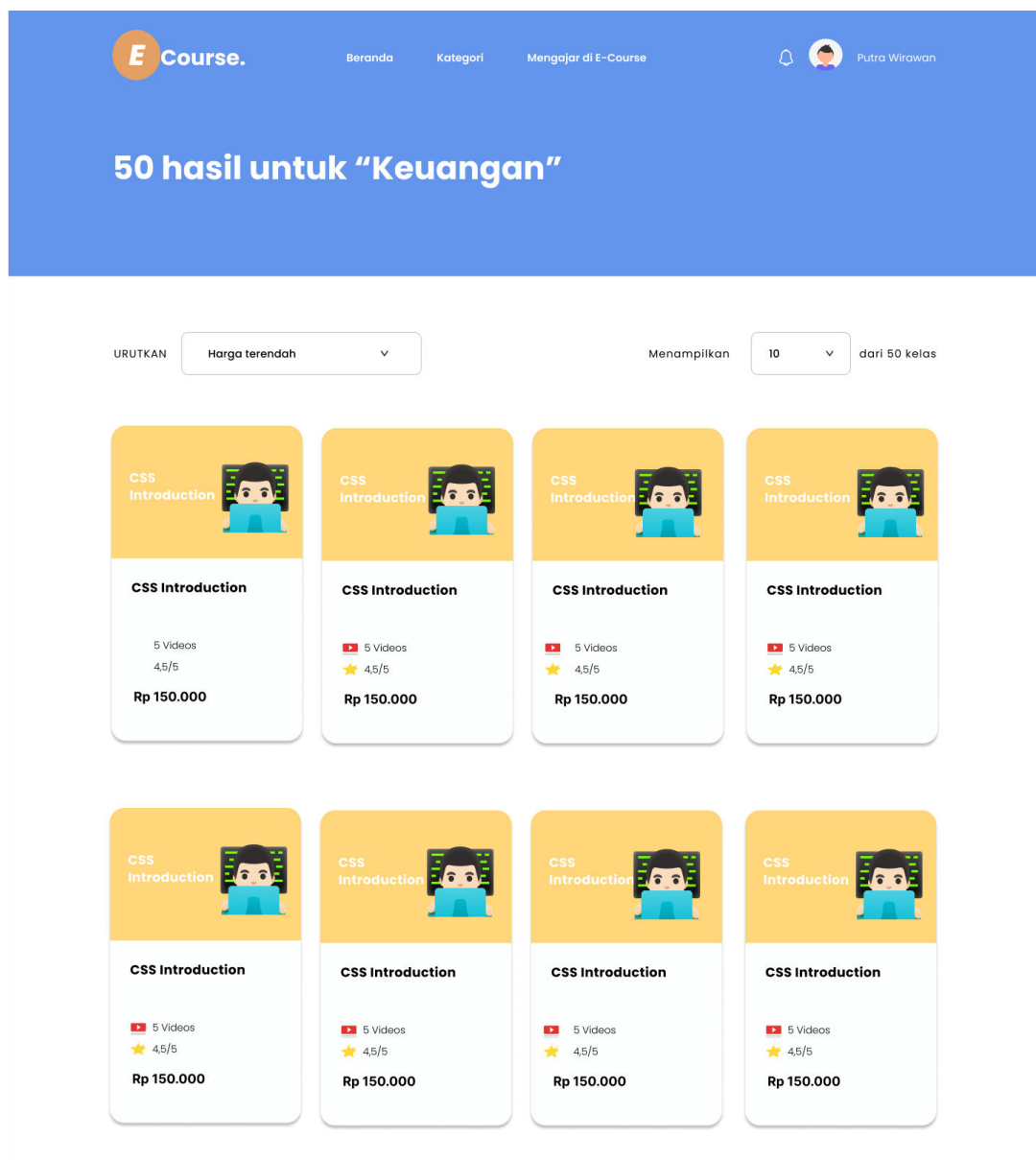
3.14.



Gambar 3.14 Desain Halaman *Landing Page*

### 3.4.2.2 Halaman Hasil Pencarian Pelatihan

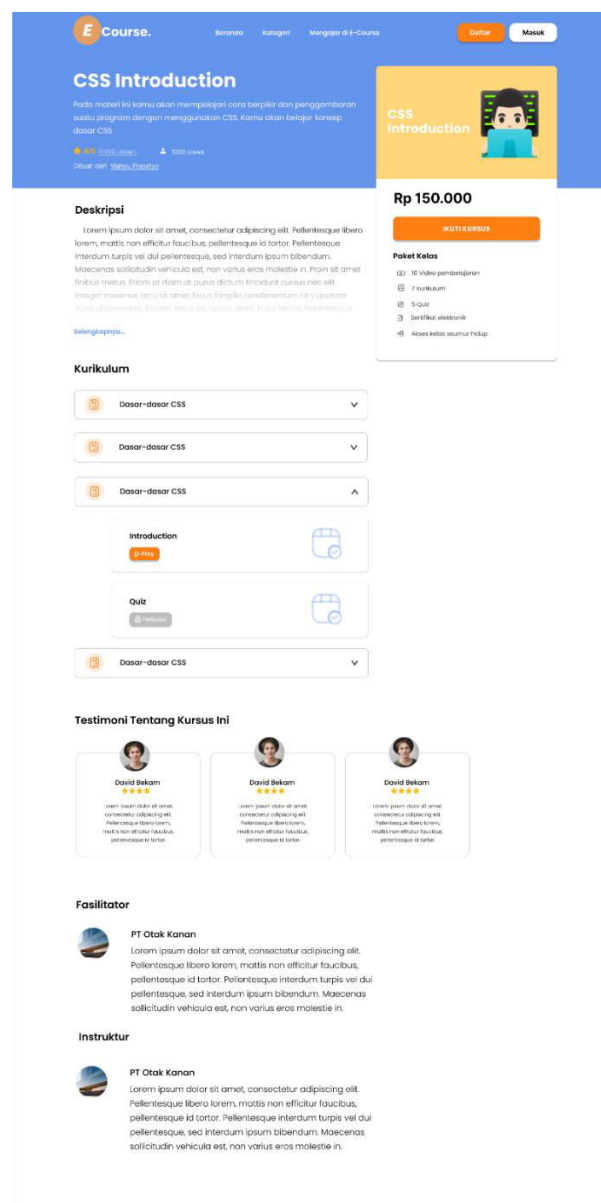
Pada halaman ini, sistem akan menampilkan daftar pelatihan berdasarkan kata kunci yang diinputkan oleh user. Desain halaman hasil pencarian pelatihan dapat dilihat pada gambar 3.15.



Gambar 3.15 Desain Halaman Hasil Pencarian Kursus

### 3.4.2.3 Halaman Detail Pelatihan

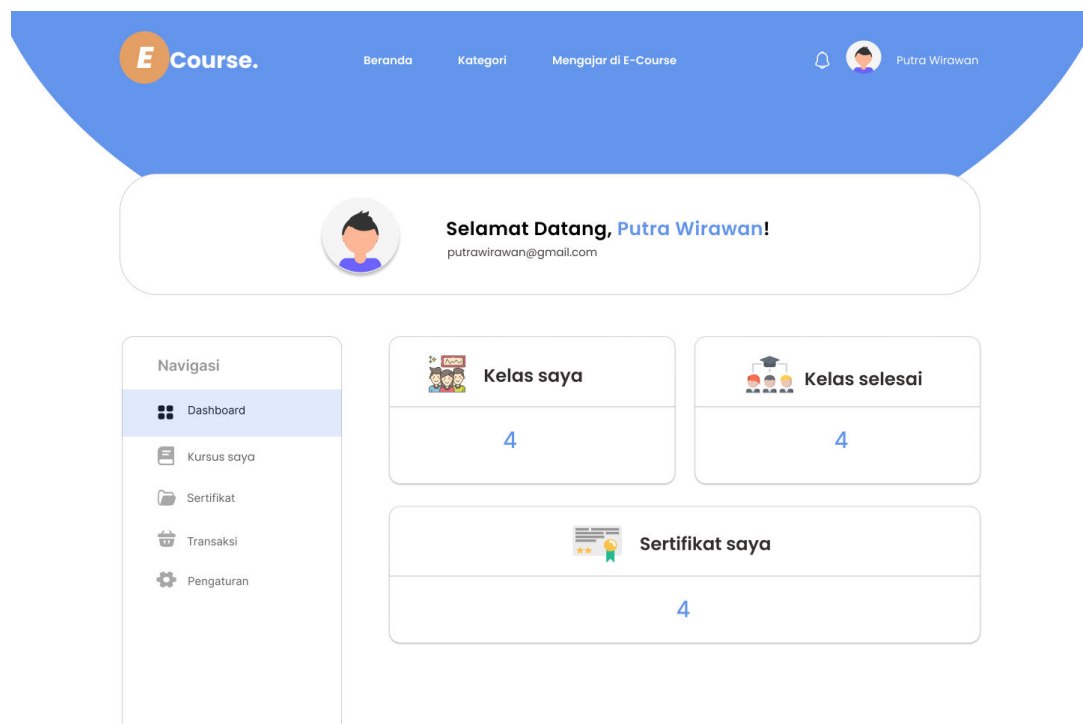
Pada halaman ini, sistem akan menampilkan detail dari salah satu pelatihan yang dipilih oleh peserta atau siswa. Desain halaman detail pelatihan dapat dilihat pada gambar 3.16.



Gambar 3.16 Desain Halaman Detail pelatihan

#### 3.4.2.4 Halaman Dashboard Siswa/Peserta

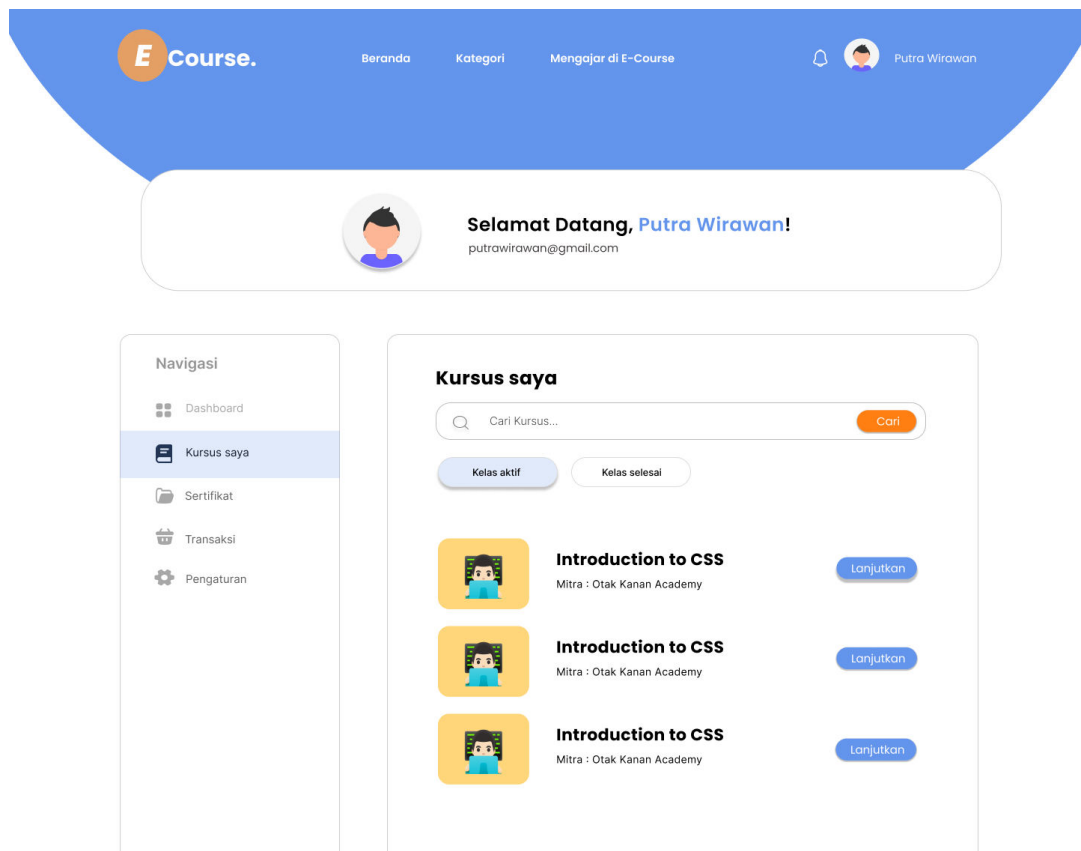
Pada halaman ini, sistem akan menampilkan daftar ringkasan pelatihan yang dimiliki siswa/peserta. Desain halaman dashboard siswa dapat dilihat pada gambar 3.17.



Gambar 3.17 Desain Halaman Dashboard Siswa/Peserta

#### 3.4.2.5 Halaman Kursus Saya

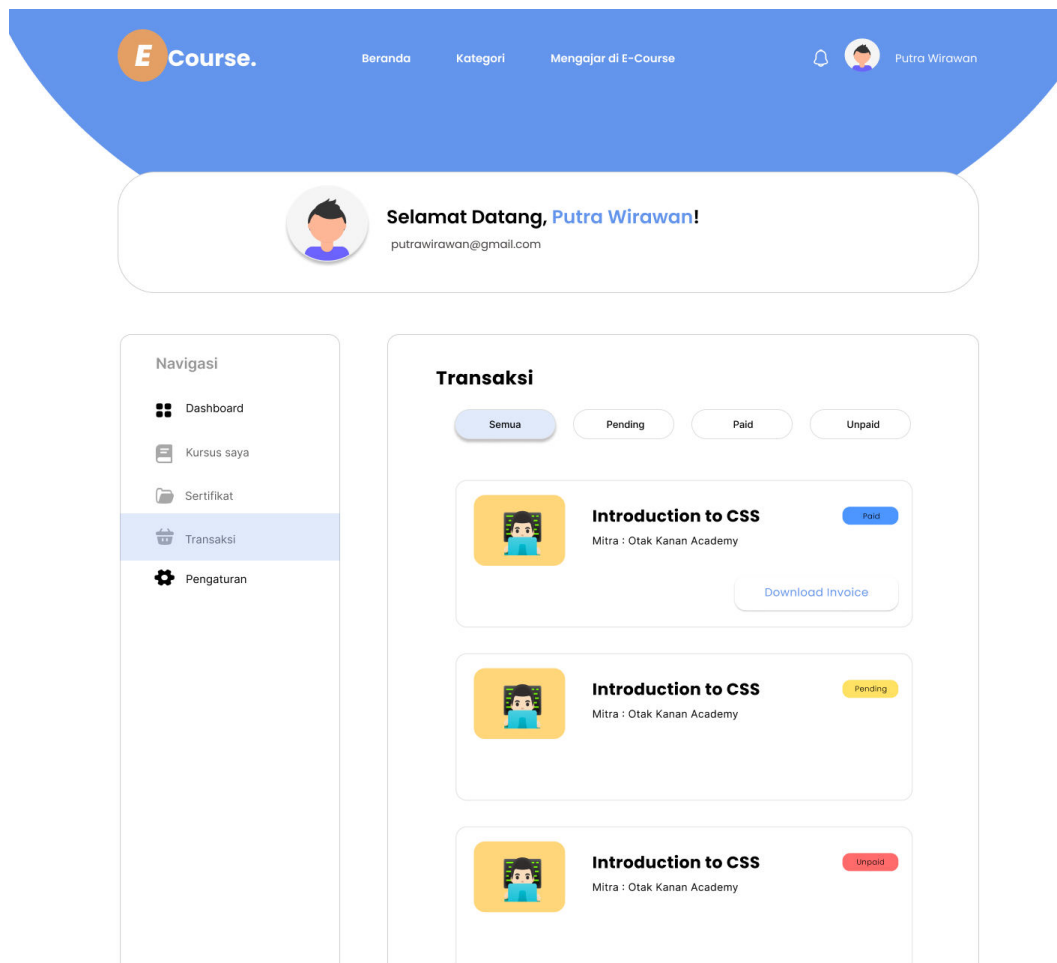
Pada halaman ini, sistem akan menampilkan daftar pelatihan yang dimiliki oleh siswa /peserta. Desain halaman kursus saya dapat dilihat pada gambar 3.18.



Gambar 3.18 Desain Halaman Kursus Saya

### 3.4.2.6 Halaman Transaksi

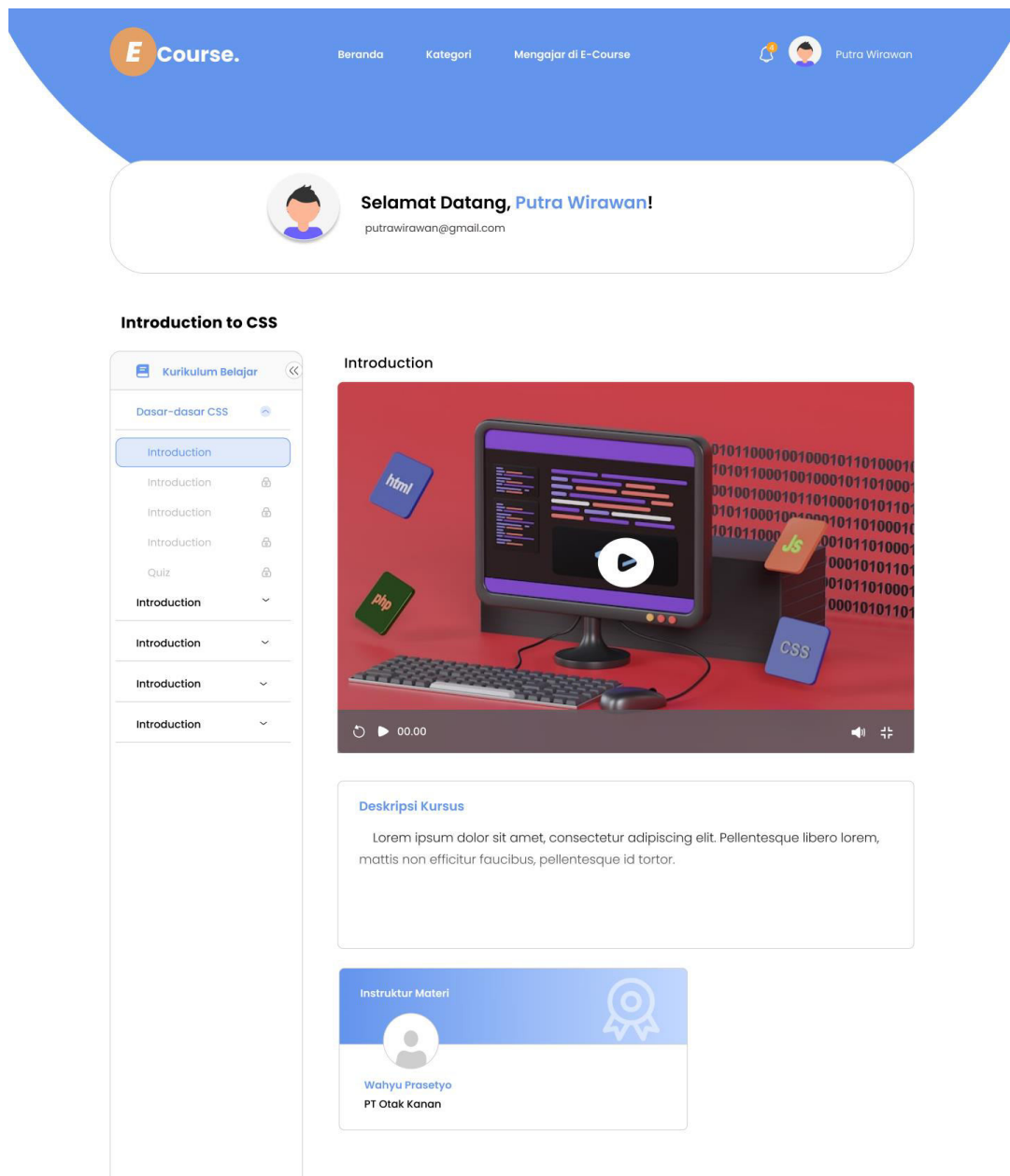
Pada halaman ini, sistem akan menampilkan daftar transaksi yang dimiliki siswa/peserta. Desain halaman transaksi dapat dilihat pada gambar 3.19.



Gambar 3.19 Desain Halaman Transaksi

### 3.4.2.7 Halaman Pembelajaran

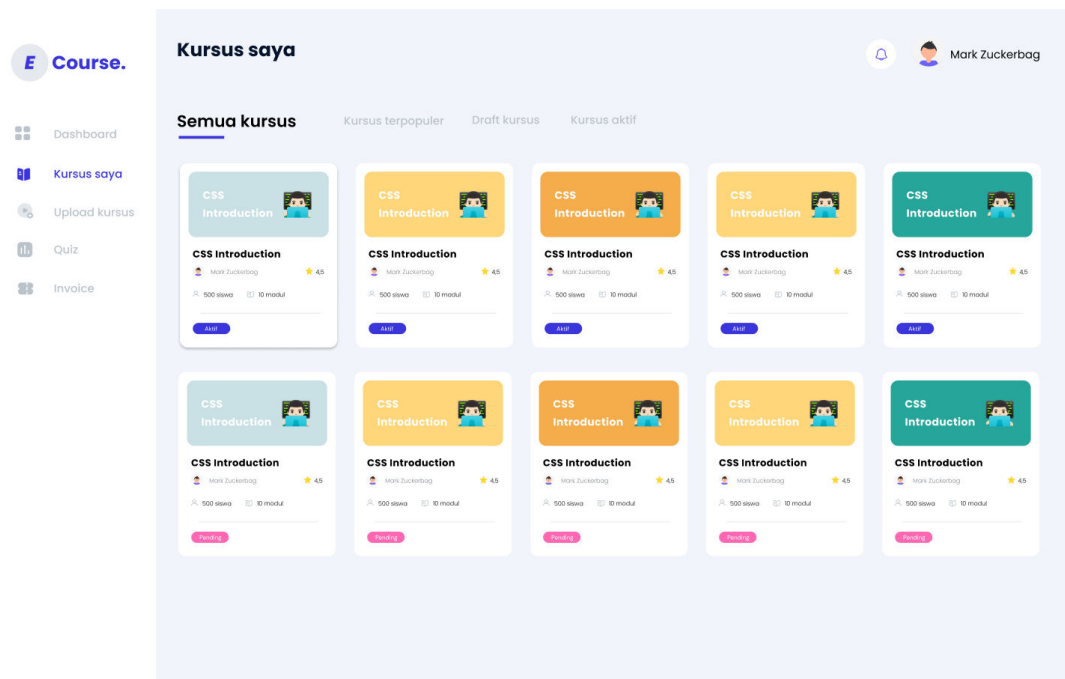
Pada halaman ini, sistem akan menampilkan materi dari pelatihan yang ingin dipelajari oleh peserta/siswa. Desain halaman pembelajaran dapat dilihat pada gambar 3.20.



Gambar 3.20 Desain Halaman Pembelajaran

### 3.4.2.8 Halaman Kursus Pada Pengajar/Admin

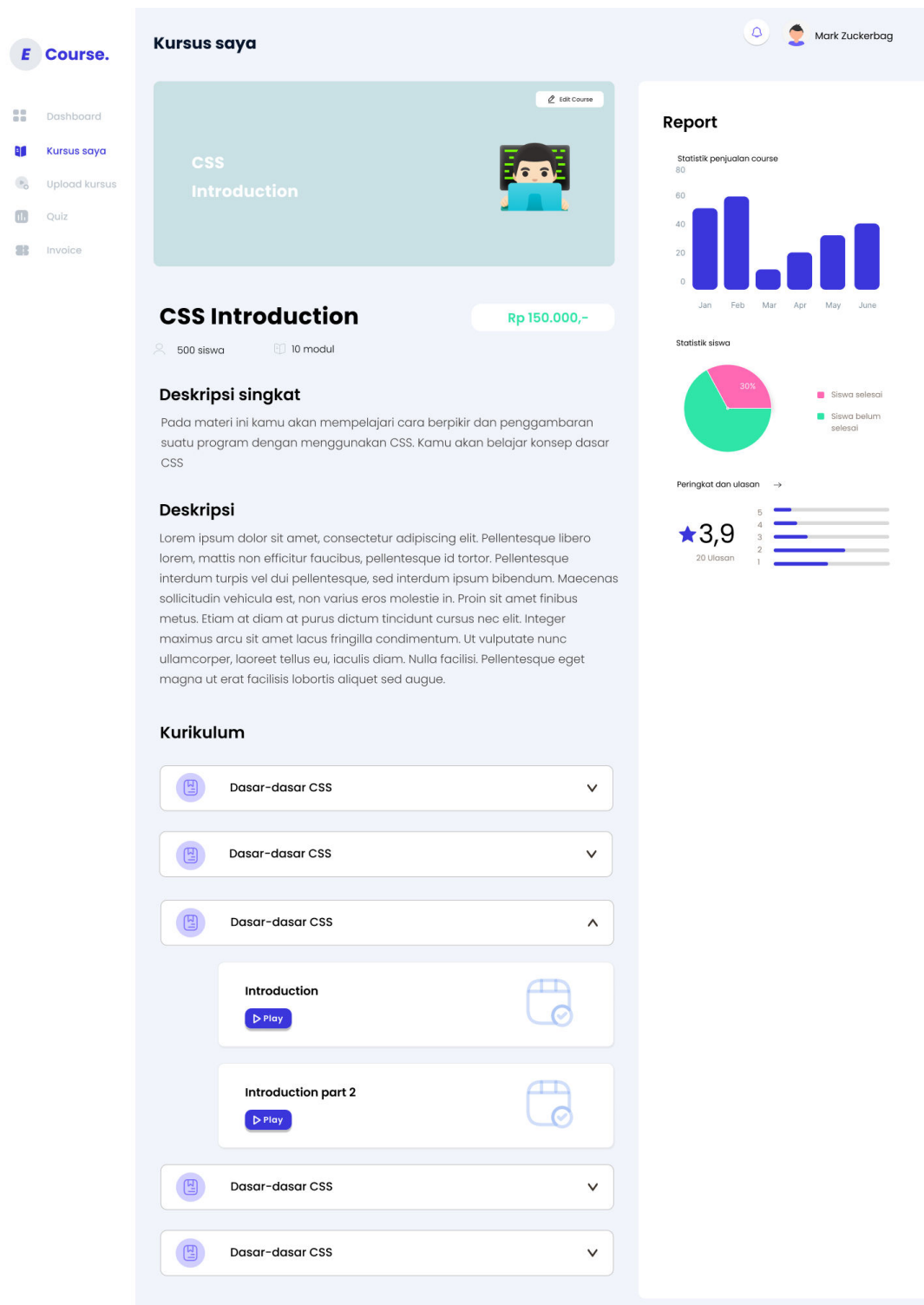
Pada halaman ini, sistem akan menampilkan semua pelatihan sesuai dengan statusnya. Desain halaman kursus dapat dilihat pada gambar 3.21.



Gambar 3.21 Desain Halaman Kursus

### 3.4.2.9 Halaman Detail Kursus Pada Pengajar/Admin

Pada halaman ini, sistem akan menampilkan detail dari pelatihan yang dipilih oleh pengajar atau admin. Desain halaman detail kursus kursus dapat dilihat pada gambar 3.22.



Gambar 3.22 Desain Halaman Detail Kursus

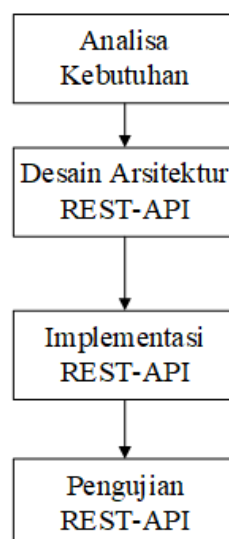
## BAB IV

### IMPLEMENTASI DAN UJI COBA

Implementasi merupakan tahap penerapan sistem yang telah dirancang sesuai dengan kebutuhan agar dapat dioperasikan dengan baik, sedangkan tahap uji coba sistem merupakan tahap pengujian terhadap sistem yang telah diimplementasikan, untuk mengetahui apakah sistem yang dirancang telah memenuhi kebutuhan dan berjalan sesuai dengan apa yang direncanakan atau tidak. Berikut ini merupakan penjabaran implementasi sistem dan uji coba sistem E-Course:

#### 4.1 Pengembangan *REST-API*

Pengembangan *REST API* pada Sistem E-Course ini bertujuan untuk memisahkan dan menyederhanakan interaksi antara client dan server. Terdapat beberapa tahap pengembangan *REST API* yang dapat dilihat pada Gambar 4.1.



Gambar 4.1 Tahap Pengembangan REST-API

#### 4.1.1 Desain Arsitektur *REST-API*

Pada pengembangan *REST API* E-Course ini memiliki beberapa kerangka yang akan dijelaskan pada tabel kontrak *API* berikut.:

##### 4.1.1.1 *API Error Validation*

*API error validation* digunakan untuk memberikan respon jika terdapat inputan yang tidak sesuai dengan yang di inginkan oleh *system*. Kontrak *API error validation* dapat dilihat dalam table 4.1 dan *Example Response API error validation* dapat dilihat dalam table 4.2.

Tabel 4.1 Tabel Kontrak *API Error Validation*

| Attribute | Tipe Data | Deskripsi                            |
|-----------|-----------|--------------------------------------|
| success   | boolean   | status Respon API (true atau false ) |
| message   | string    | Pesan dari respon API                |

Tabel 4.2 Tabel *Example Response API Error Validation*

```
{
  "success": false,
  "message": {
    "email": [
      "The email field must be a valid email address."
    ]
  }
}
```

Dalam contoh tabel diatas terdapat kesalahan inputan dari klien Ketika memasukkan email sehingga system akan memberikan pesan validasi error jika email yang dimasukkan tidak valid.

#### 4.1.1.2 API Response Error

*API response error* digunakan untuk memberikan respon jika terdapat proses yang tidak dapat diselesaikan oleh system. Kontrak *API response error* dapat dilihat dalam table 4.3 dan *Example Response API response error* dapat dilihat dalam table 4.4.

Tabel 4.3 Tabel Kontrak API Response Error

| Attribute | Tipe Data | Deskripsi                            |
|-----------|-----------|--------------------------------------|
| success   | boolean   | status Respon API (true atau false ) |
| message   | string    | Pesan dari respon API                |
| error     | string    | Pesan error respon API               |

Tabel 4.4 Tabel Example Response API Response Error

```
{ "success": false,
  "message": "Error creating order",
  "error": [
    "HY000",
    1364,
    "Field 'invoice_id' doesn't have a default value"
  ]
}
```

Dalam contoh response diatas, column invoice\_id tidak boleh kosong, sehingga Ketika membuat data order dalam keadaan invoice\_id kosong maka system tidak dapat membuat data order baru.

#### 4.1.1.3 *API Response Success*

*API response success* digunakan untuk memberikan respon jika terdapat proses yang berhasil diselesaikan oleh system tanpa mengembalikan data. Kontrak *API response success* dapat dilihat dalam table 4.5 dan *Example Response API response success* dapat dilihat dalam table 4.5.

Tabel 4.5 Tabel Kontrak *API Response Success*

| Attribute | Tipe Data | Deskripsi                            |
|-----------|-----------|--------------------------------------|
| success   | boolean   | status Respon API (true atau false ) |
| message   | string    | Pesan dari respon API                |

Tabel 4.6 Tabel *Example Response API Response Success*

```
{
  "success": true,
  "message": "Course deleted"
}
```

#### 4.1.1.4 *API Response Success With Data*

*API response success with data* digunakan untuk memberikan respon jika terdapat proses yang berhasil diselesaikan oleh system dengan mengembalikan

data. Kontrak *API response success with data* dapat dilihat dalam table 4.7 dan

*Example Response API response success with data* dapat dilihat dalam table 4.8.

Tabel 4.7 Tabel Kontrak API Response Success with data

| Attribute | Tipe Data | Deskripsi                            |
|-----------|-----------|--------------------------------------|
| success   | boolean   | status Respon API (true atau false ) |
| message   | string    | Pesan dari respon API                |
| data      | object    | Data hasil query                     |

Tabel 4.8 Tabel Example Response API Response Success With Data

```
{
  "success": true,
  "message": "Successfully registered.",
  "data": {
    "name": "rahmad",
    "email": "rahmad@gmail.com",
    "role_id": 3,
    "slug": "rahmad-NKx48",
    "updated_at": "2023-07-08T13:19:32.000000Z",
    "created_at": "2023-07-08T13:19:32.000000Z",
    "id": 8
  }
}
```

#### 4.1.2 Implementasi *REST API*

Pada arsitektur *REST API* terdapat beragam *method* yang digunakan pada protokol *HTTP*, namun pada proyek tugas akhir ini ini hanya memerlukan tiga method yaitu *GET* dan *POST*. Pada Tabel 4.9 merupakan beberapa daftar *endpoint REST API* pada proyek tugas akhir e-course.

Tabel 4.9 Tabel beberapa daftar *endpoint REST API*

| Endpoint                      | Method | Kegunaan                                                       |
|-------------------------------|--------|----------------------------------------------------------------|
| /auth/login                   | post   | Untuk proses login                                             |
| /course/popular               | get    | Untuk mengambil data pelatihan terpopuler                      |
| /course/popular/{category_id} | get    | Untuk mengambil data pelatihan terpopuler berdasarkan kategori |
| /category                     | get    | Untuk mengambil data kategori                                  |
| /order/create-payment         | post   | Untuk memproses data pembayaran peserta                        |

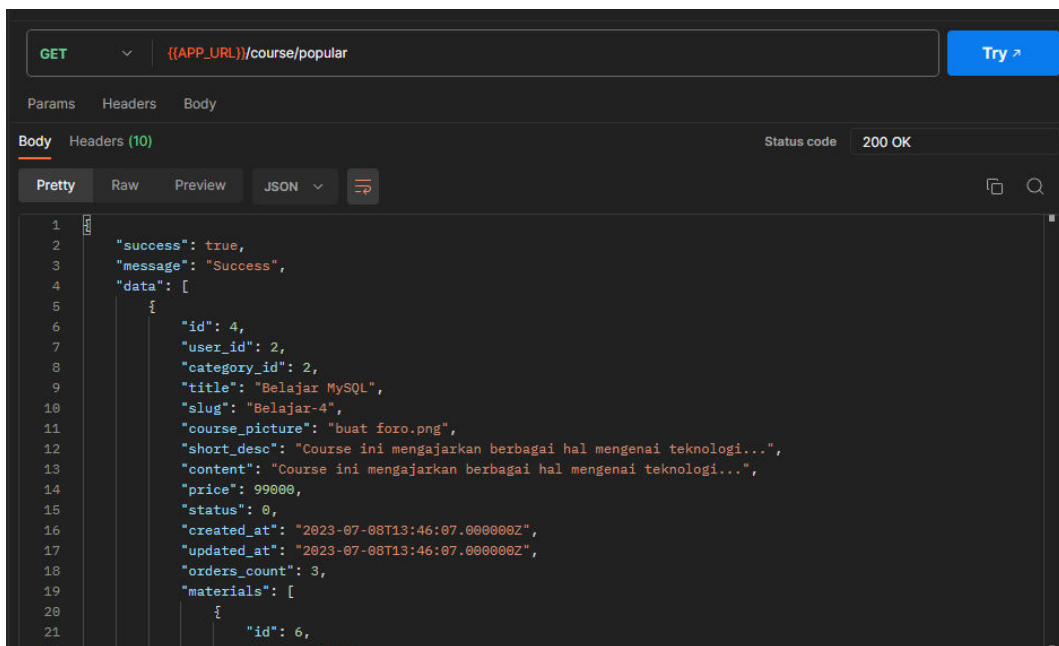
Selanjutnya, perlu membuat controller untuk menentukan proses pengiriman data, agar data dapat diakses oleh pengguna. Controller akan berisikan *function* yang akan memproses data atau query untuk dimasukkan ke dalam kerangka *REST API*. Gambar 4.2 adalah salah satu contoh implementasi dari endpoint *API /course/popular* pada controller menggunakan method GET.

Pada Gambar 4.3, merupakan hasil eksekusi url */course/popular* menggunakan Postman. *Output* yang didapatkan berupa format JSON.

```
public function popular()
{
    $courses = Course::with(['materials.items', 'user:id,name,email', 'reviews','orders'])
        ->withCount('orders')
        ->orderByDesc('orders_count')
        ->where('status','accepted')
        ->get();

    if(count($courses)){
        return $this->responseSuccessWithData('Success', $courses);
    }
    else{
        return $this->responseError('Course not found', 404);
    }
}
```

Gambar 4.2 Implementasi endpoint *API /course/popular*



```
{
  "success": true,
  "message": "Success",
  "data": [
    {
      "id": 4,
      "user_id": 2,
      "category_id": 2,
      "title": "Belajar MySQL",
      "slug": "Belajar-4",
      "course_picture": "buat foro.png",
      "short_desc": "Course ini mengajarkan berbagai hal mengenai teknologi...",
      "content": "Course ini mengajarkan berbagai hal mengenai teknologi...",
      "price": 99000,
      "status": 0,
      "created_at": "2023-07-08T13:46:07.000000Z",
      "updated_at": "2023-07-08T13:46:07.000000Z",
      "orders_count": 3,
      "materials": [
        {
          "id": 6,
          "course_id": 4,
          "material": "Belajar MySQL",
          "slug": "Belajar-4",
          "status": 0,
          "created_at": "2023-07-08T13:46:07.000000Z",
          "updated_at": "2023-07-08T13:46:07.000000Z"
        }
      ]
    }
  ]
}
```

Gambar 4.3 eksekusi url */course/popular* pada postman

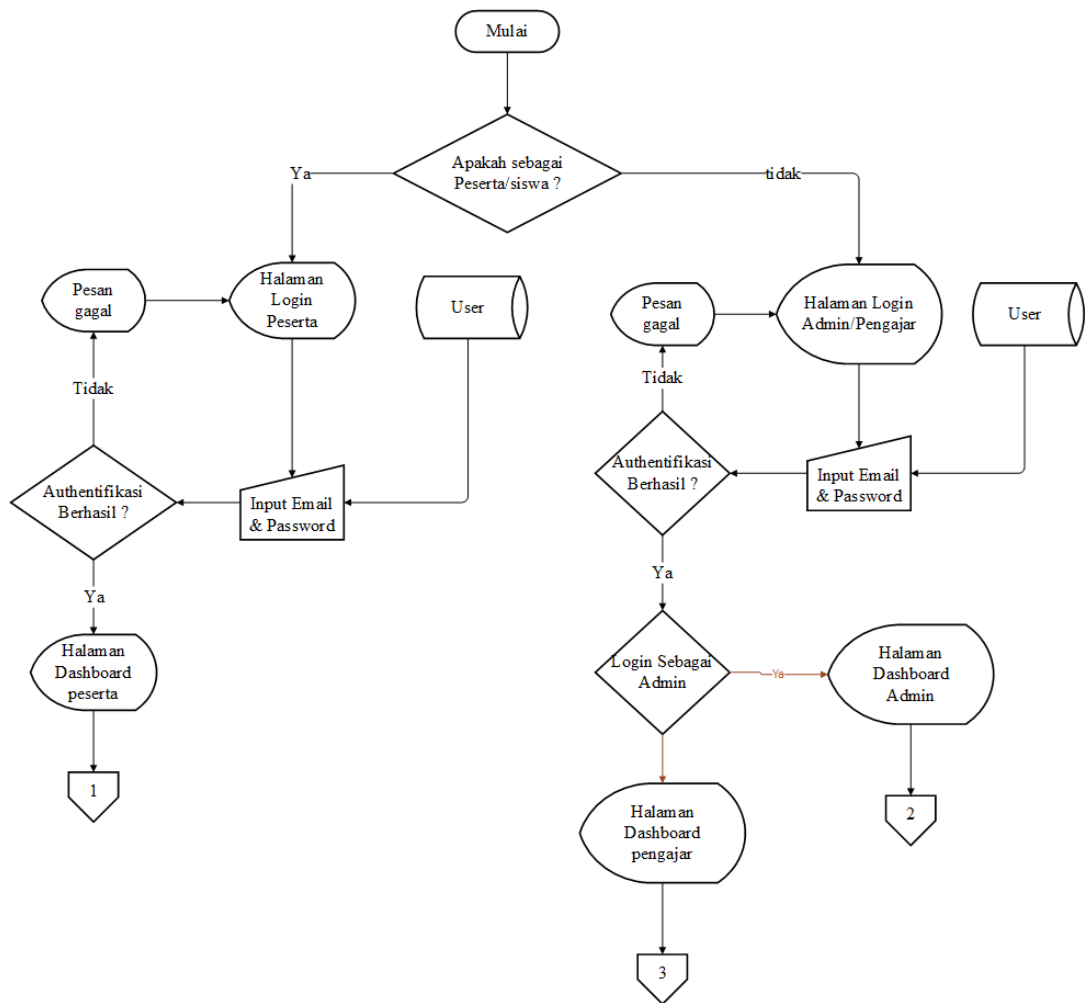
Penulisan output data dengan format JSON seperti pada Gambar 3.8, memudahkan sisi client untuk membaca data karena memiliki struktur kode yang sederhana dibandingkan dengan format XML.

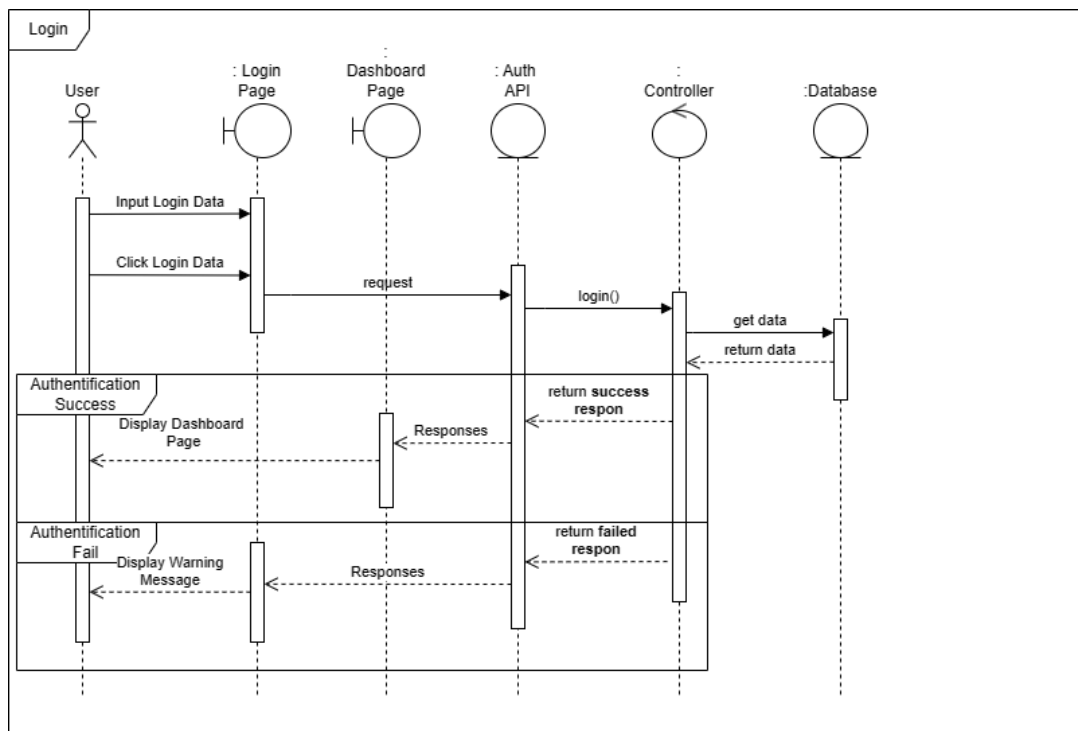
## 4.2 Implementasi Sistem

Implementasi sistem E-Course akan dijelaskan dengan bagan alir sistem yang akan merepresentasikan alur kerja sistem dan *sequence diagram* yang menggambarkan alur penggunaan REST-API, namun tidak semua alur kerja sistem memuat penjelasan mengenai *sequence diagram*, karena hanya beberapa alur kerja yang memanfaatkan REST-API . Berikut bagan alir sistem dan *sequence diagram* beserta penjelasannya.

### 4.2.1 Proses Alur *Login*

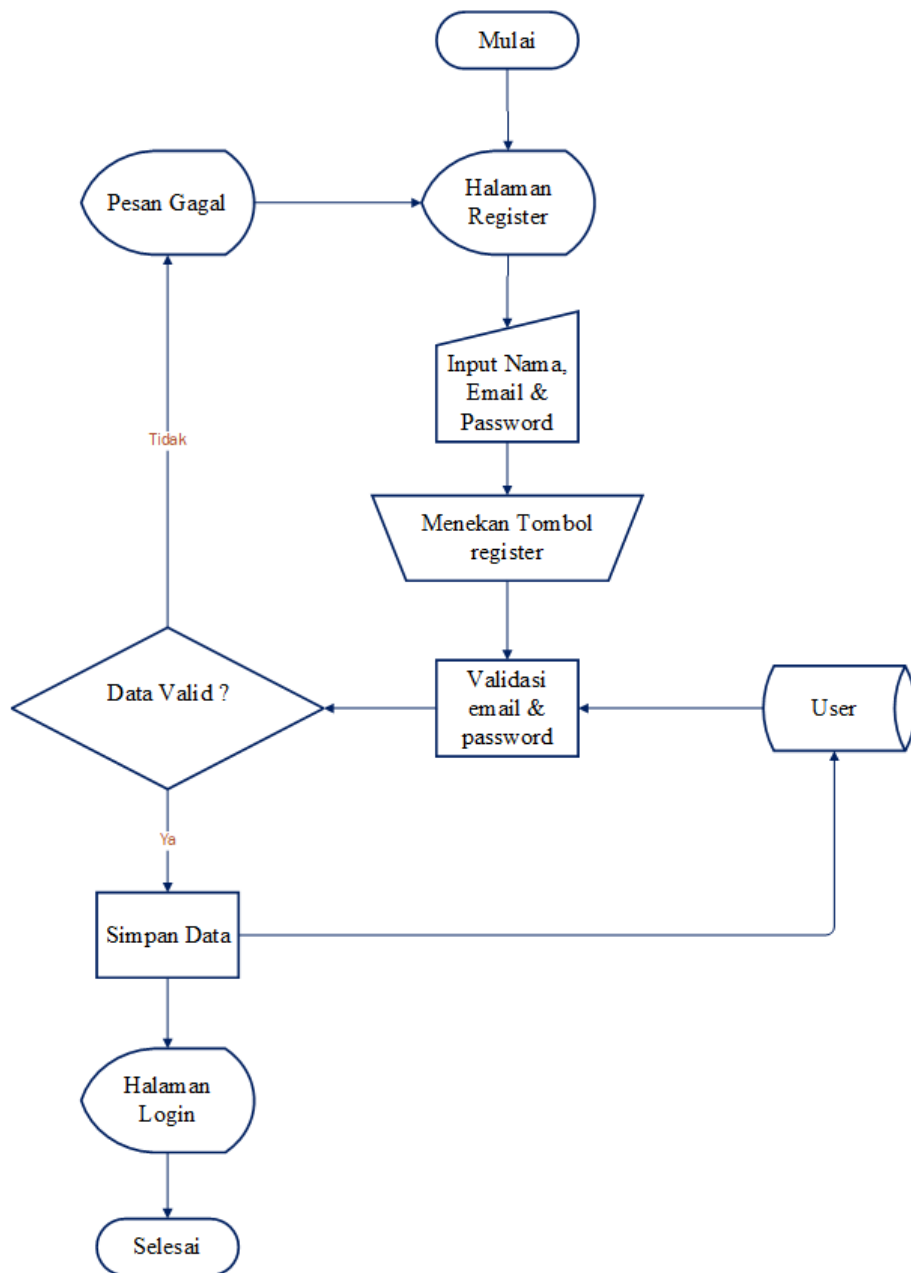
Proses *login* dilakukan semua pengguna E-Course untuk dapat mendapat hak akses sesuai role yang dimiliki. Bagan alir *login* menggambarkan proses *login* tiap pengguna sesuai dengan hak akses masing – masing. Sedangkan *sequence diagram login* menggambarkan penggunaan REST-API untuk proses *login* pelajar. Terdapat tiga pengguna yang memiliki hak akses kedalam sistem ini, yaitu siswa/pelajar, pengajar dan admin. Bagan alir login dapat dilihat pada gambar 4.1 dan *sequence diagram login* dapat dilihat pada gambar 4.2.

Gambar 4.4 Bagan Alir *Login*

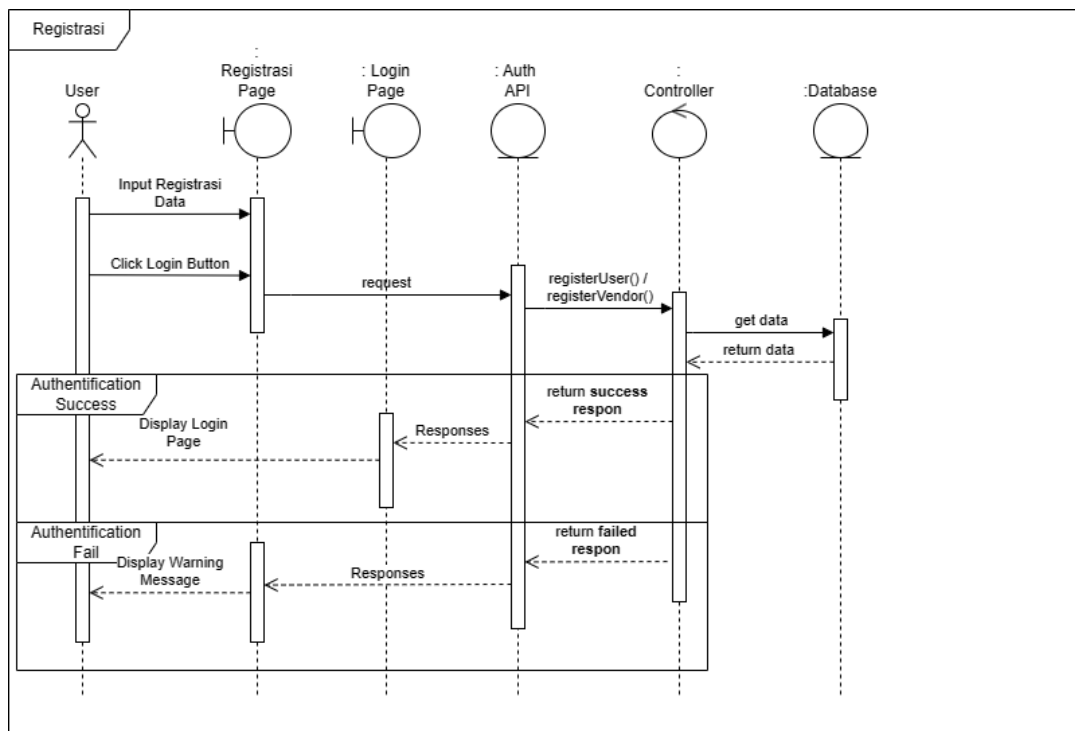
Gambar 4.5 Sequence Diagram Alir *Login*

#### 4.2.2 Proses Alur Registrasi

Proses Registrasi merupakan proses yang dilakukan oleh siswa/pengajar dan pengajar untuk menyimpan data user. Urutan alur registrasi dapat dilihat pada gambar 4.3 bagan alir registrasi dan penggunaan REST-API dapat dilihat pada gambar 4.4 *sequence diagram* registrasi.



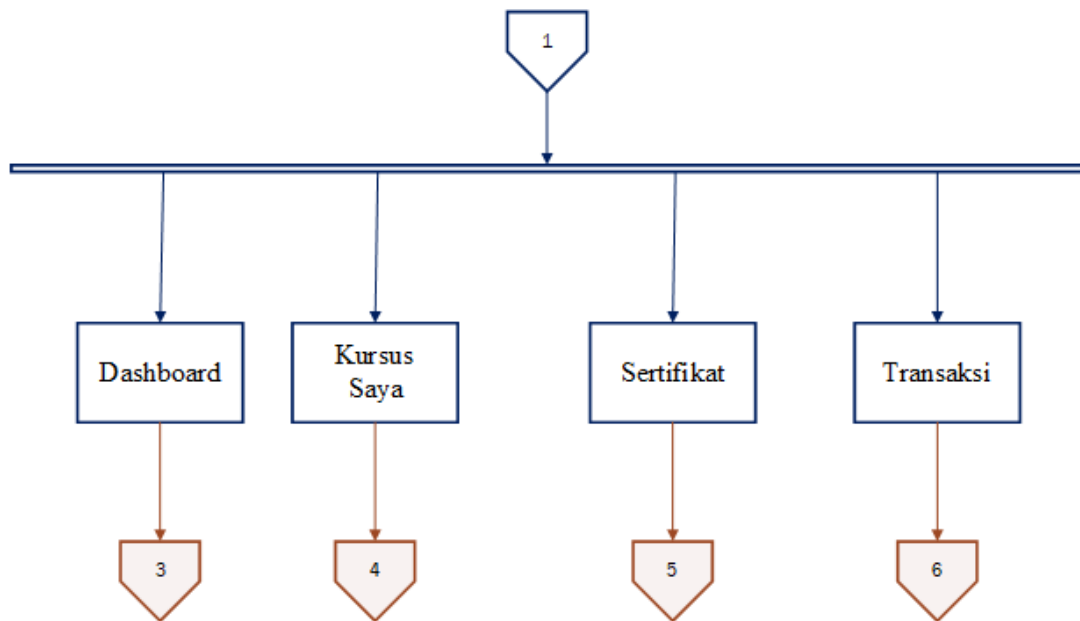
Gambar 4.6 Bagan Alir Registrasi



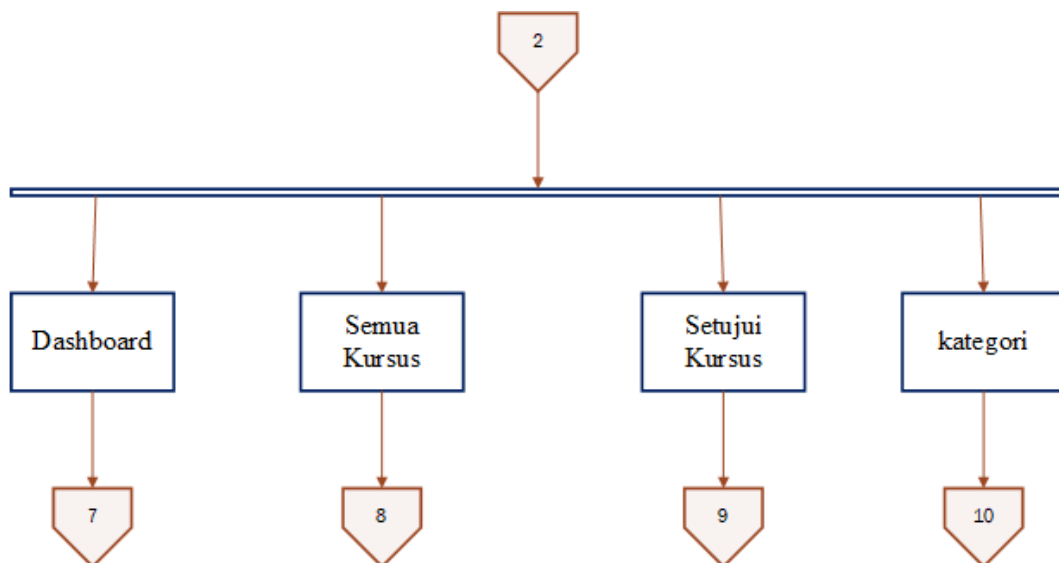
Gambar 4.7 Sequence Diagram Registrasi

### 4.2.3 Proses Alur Menu

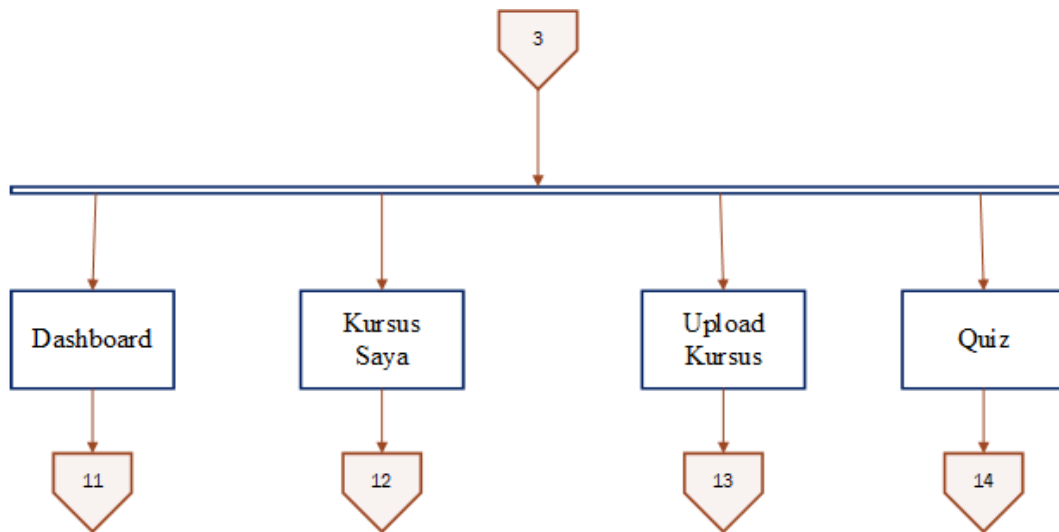
Proses Alur menu menggambarkan menu-menu apa saja yang dapat diakses oleh pengguna sistem sesuai dengan hak akses pengguna masing-masing. Dalam sistem ini terdapat tiga alur menu yang berbeda yaitu menu untuk siswa/pelajar dan menu untuk bagian admin dan pengajar. Bagan alir menu siswa/pelajar dapat dilihat pada gambar 4.5, Bagan alir menu bagian admin dapat dilihat pada gambar 4.6 dan pengajar dapat dilihat pada gambar 4.7.



Gambar 4. 8 Bagan Alir Menu Siswa/Pelajar



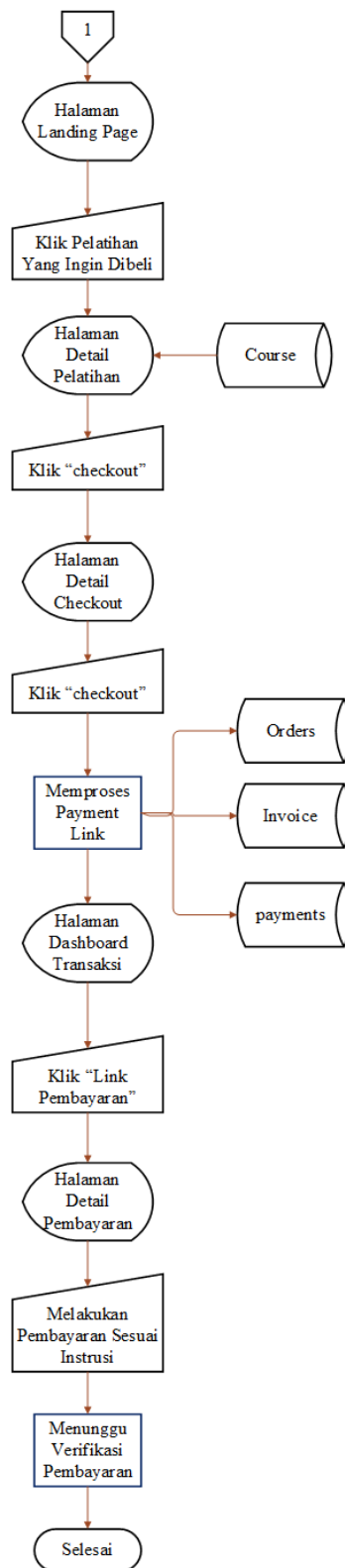
Gambar 4. 9 Bagan Alir Menu Admin



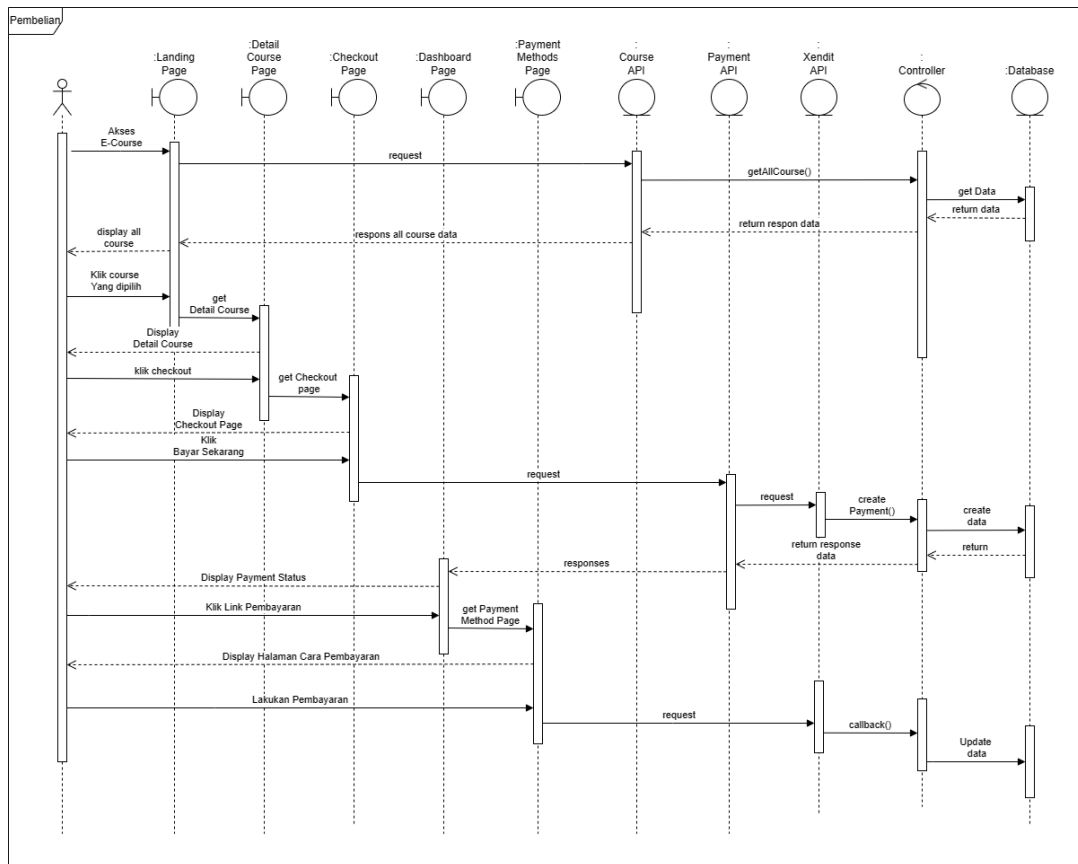
Gambar 4. 10 Bagan Alir Menu Pengajar

#### 4.2.4 Proses Alur Pendaftaran Pelatihan

Proses alur pendaftaran pelatihan menjelaskan urutan yang dilalui siswa/pelajar Ketika akan membeli pelatihan melalui E-Course. Urutan alur pendaftaran pelatihan dapat dilihat pada gambar 4.8 bagan alir alur pendaftaran pelatihan dan alur penggunaan REST-API dapat dilihat pada gambar 4.9 *sequence diagram* alur pendaftaran pelatihan.



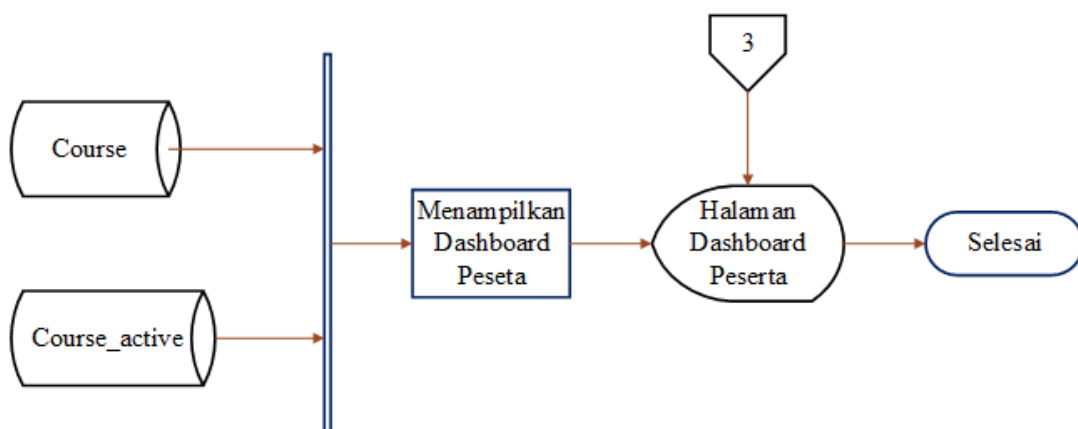
Gambar 4.11 Bagan Alir Pendaftaran Pelatihan



Gambar 4.12 Sequence Diagram Pendaftaran Pelatihan

#### 4.2.5 Proses Alur Menampilkan Dashboard Peserta

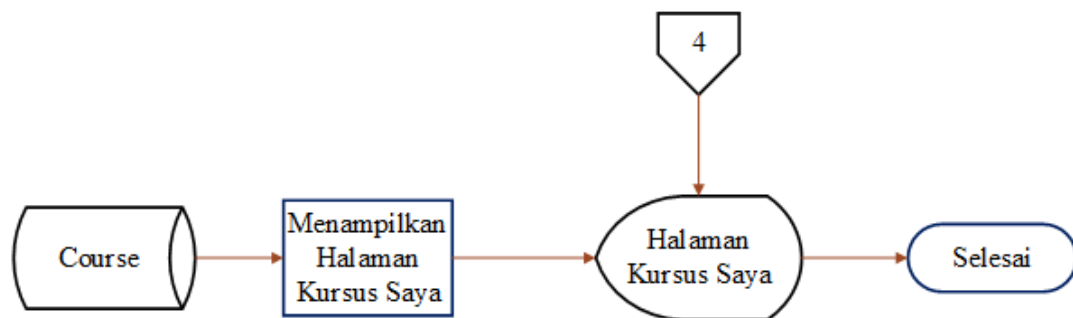
Proses Alur untuk menampilkan dashboard pada sisi peserta dapat dilihat pada gambar 4.10 Bagan Alir Dashboard Peserta.



Gambar 4.13 Bagan Alir Dashboard Peserta

#### 4.2.6 Proses Alur Menampilkan Data Pelatihan Peserta

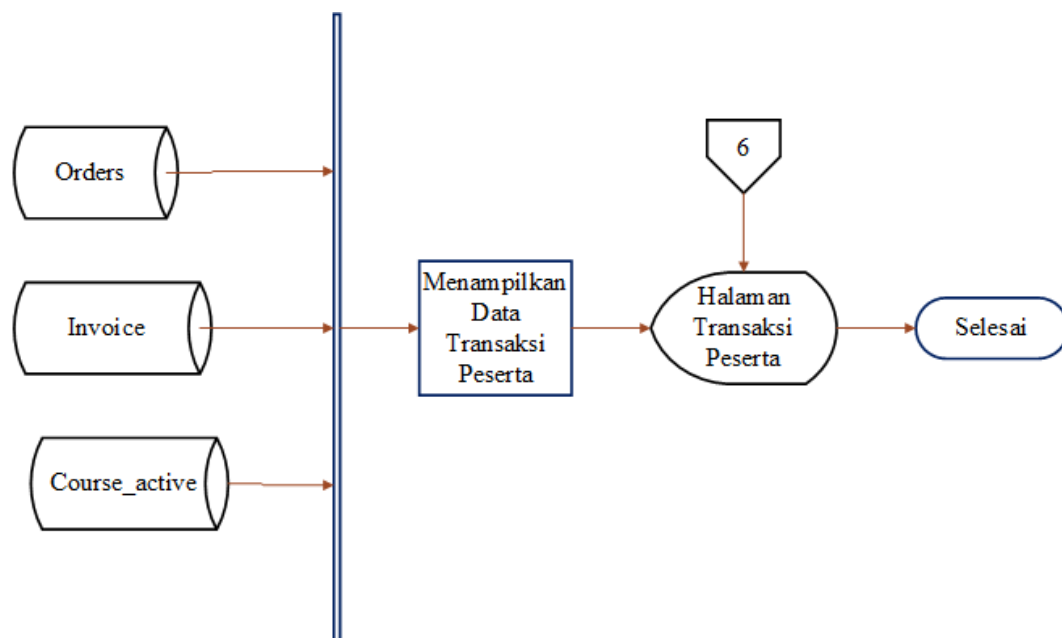
Proses Alur untuk menampilkan data pelatihan pada sisi peserta dapat dilihat pada gambar 4.11 Bagan Alir Pelatihan Peserta.



Gambar 4.14 Bagan Alir Pelatihan Peserta

#### 4.2.7 Proses Alur Menampilkan Data Transaksi Peserta

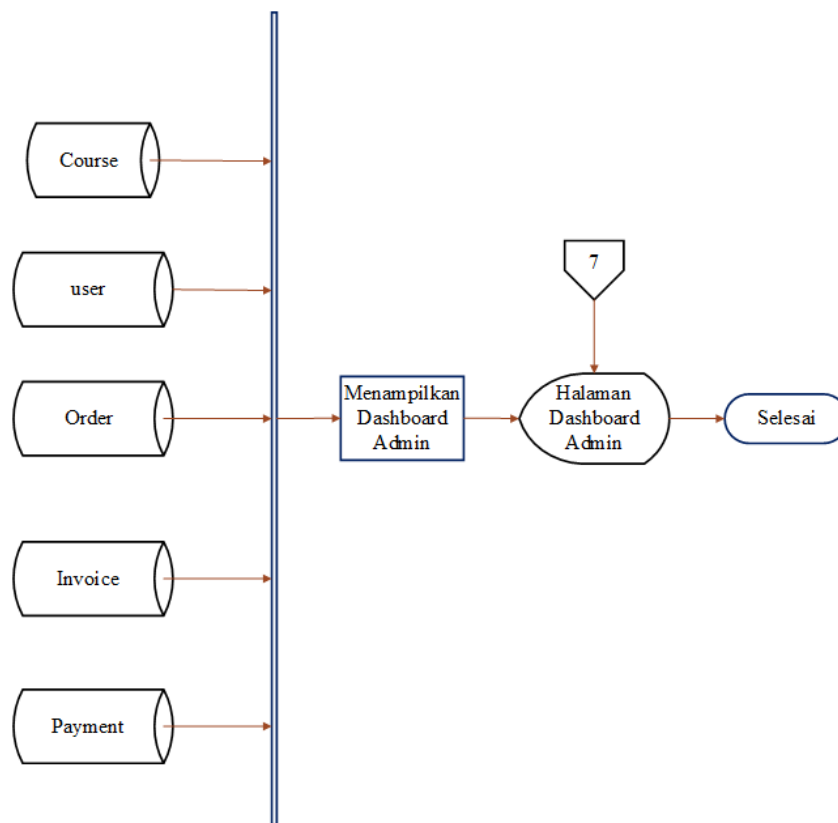
Proses Alur untuk menampilkan data transaksi pada sisi peserta dapat dilihat pada gambar 4.12 Bagan Alir Data Transaksi Peserta.



Gambar 4.15 Bagan Alir Data Transaksi Peserta

#### 4.2.8 Proses Alur Menampilkan Dashboard Admin

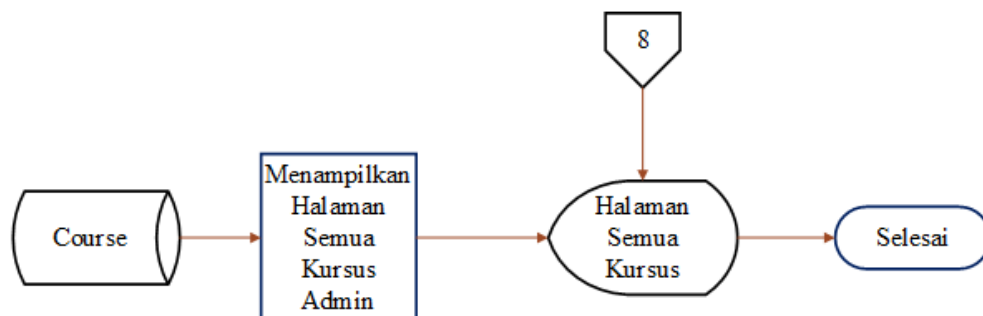
Proses Alur untuk menampilkan dashboard pada sisi admin dapat dilihat pada gambar 4.13 Bagan Alir Dashboard Admin.



Gambar 4.16 Bagan Alir Dashboard Admin

#### 4.2.9 Proses Alur Menampilkan Semua Pelatihan Admin

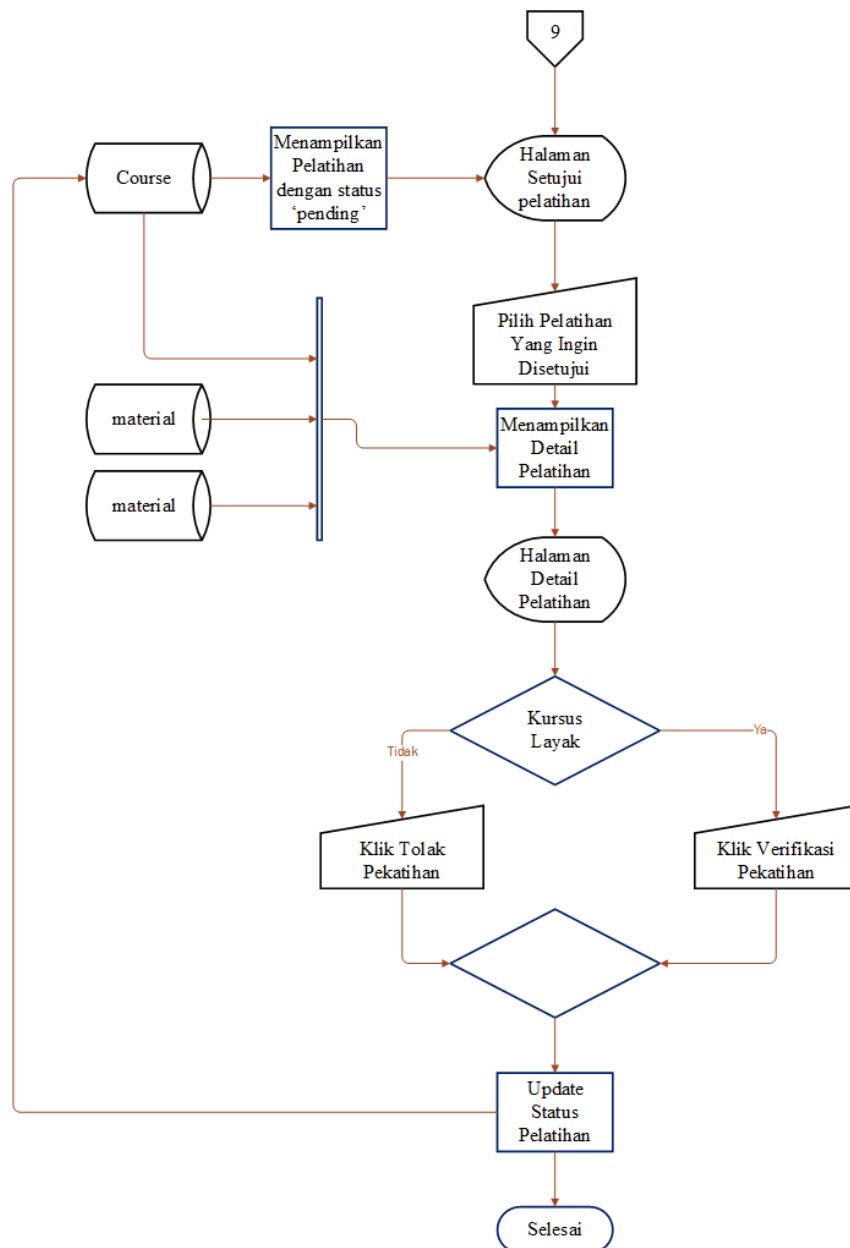
Proses Alur untuk menampilkan semua pelatihan pada sisi admin dapat dilihat pada gambar 4.14 Bagan Alir Semua Pelatihan Admin.



Gambar 4.17 Bagan Alir Semua Pelatihan Admin

#### 4.2.10 Proses Alur Menyetujui Pelatihan Admin

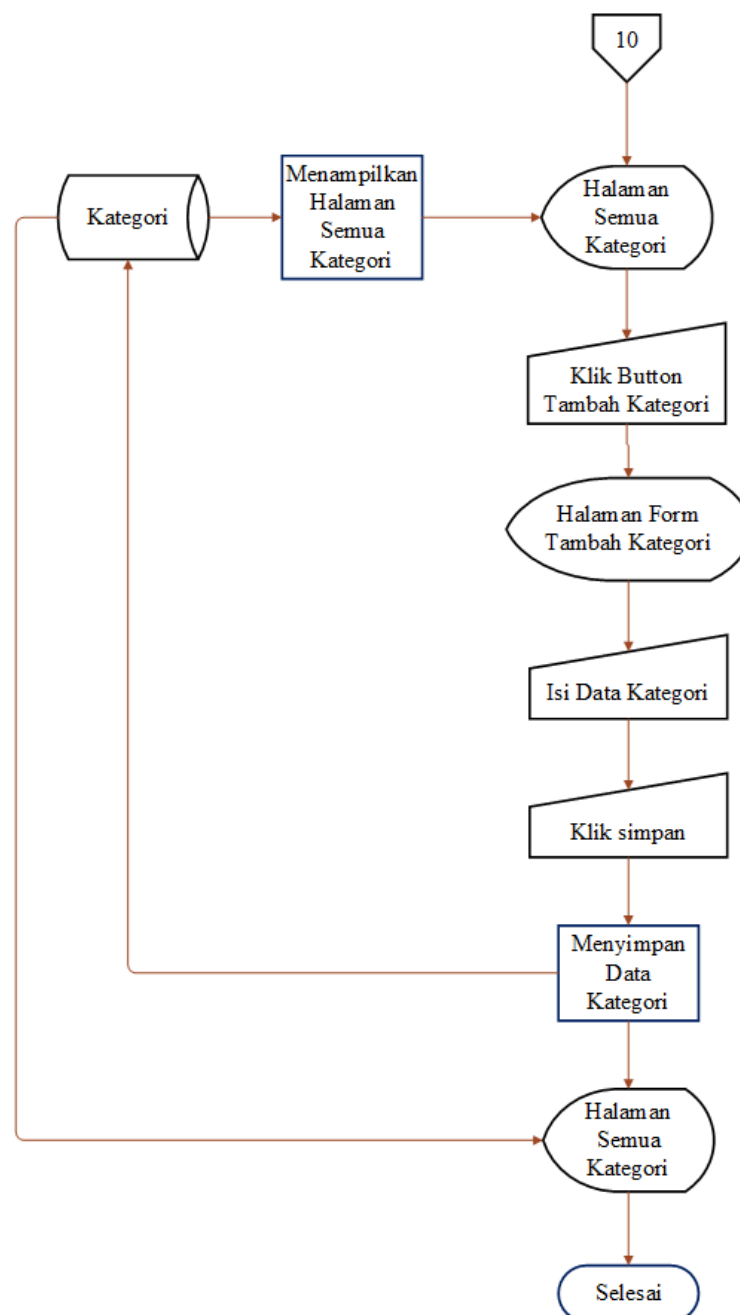
Proses Alur untuk menyetujui pelatihan pada sisi admin dapat dilihat pada gambar 4.15 Bagan Alir Menyetujui Pelatihan Admin.



Gambar 4.18 Bagan Alir Menyetujui Pelatihan Admin

#### 4.2.11 Proses Alur Membuat Kategori Admin

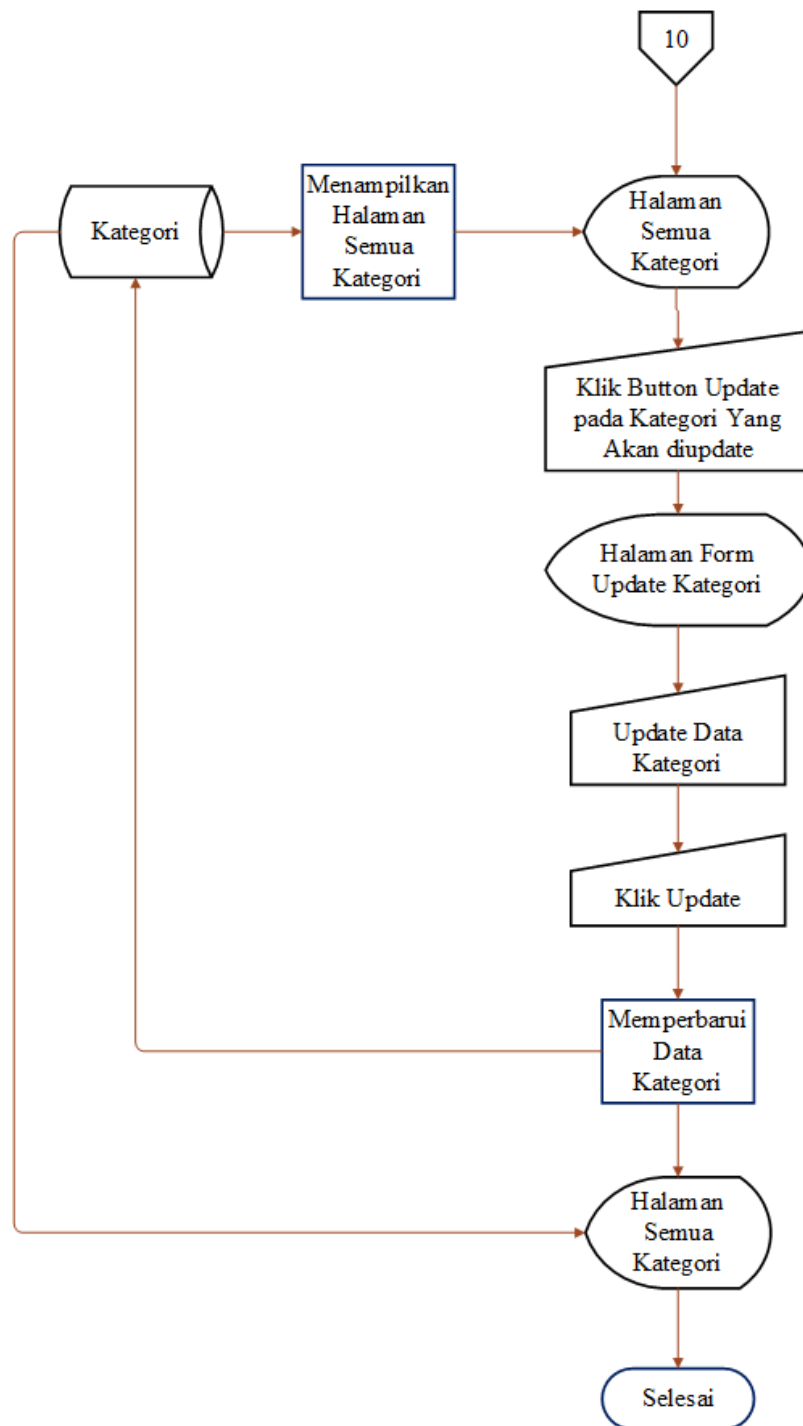
Proses Alur untuk membuat data kategori pada sisi admin dapat dilihat pada gambar 4.16 Bagan Alir Membuat Kategori Admin.



Gambar 4.19 Bagan Alir Membuat Kategori Admin

#### 4.2.11 Proses Alur Memperbarui Data Kategori Admin

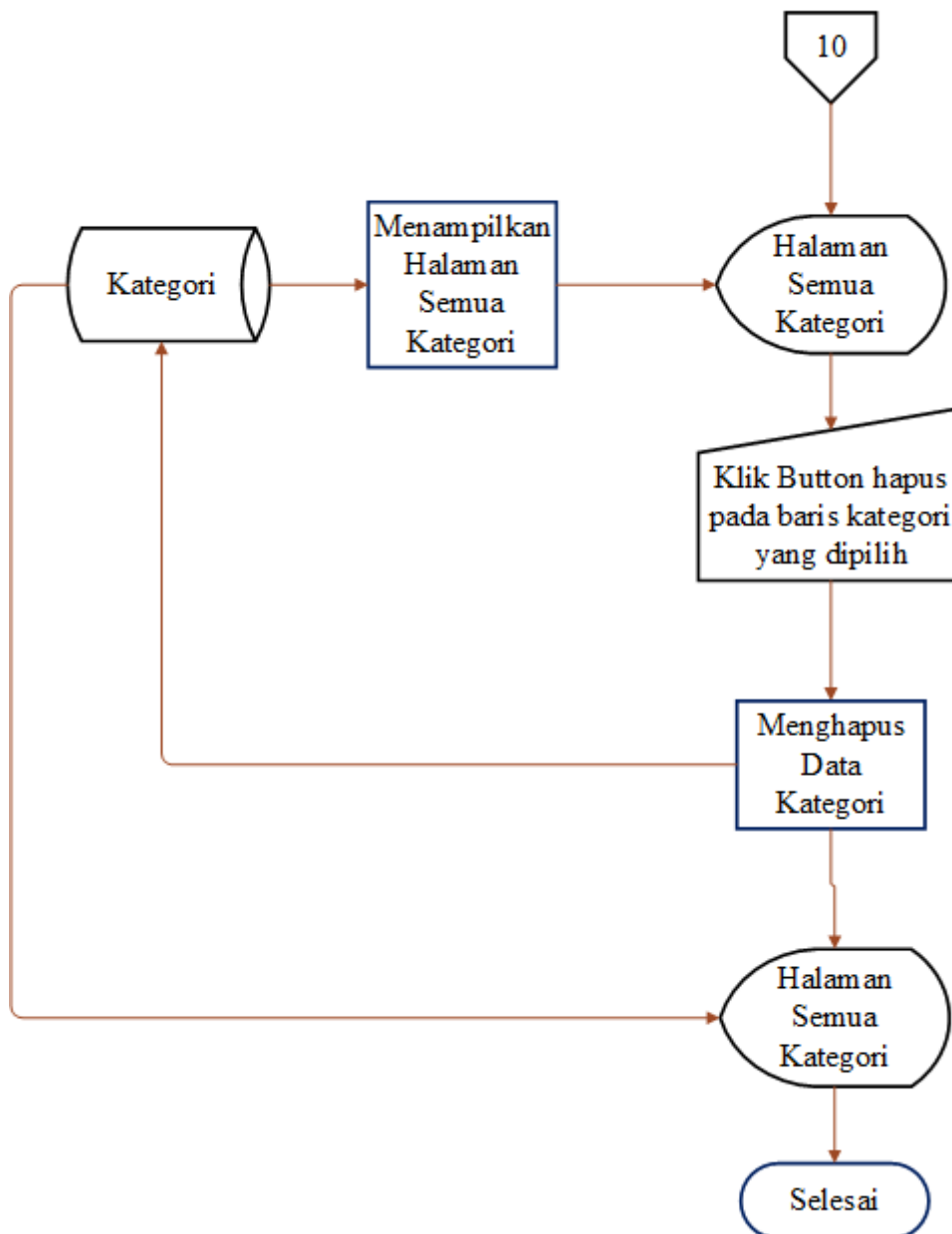
Proses Alur untuk memperbarui data kategori pada sisi admin dapat dilihat pada gambar 4.17 Bagan Alir Memperbarui Data Kategori Admin.



Gambar 4.20 Bagan Alir Memperbarui Data Kategori Admin

#### 4.2.12 Proses Alur Menghapus Data Kategori Admin

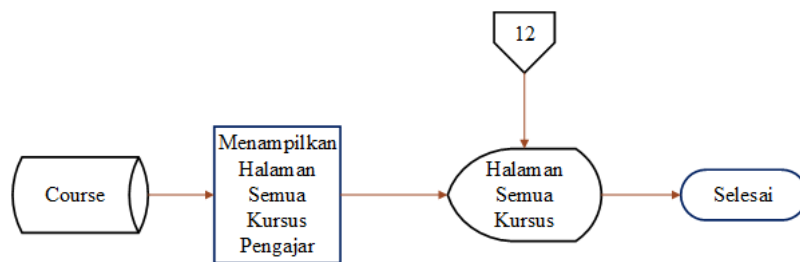
Proses Alur untuk menghapus data kategori pada sisi admin dapat dilihat pada gambar 4.18 Bagan Alir Menghapus Data Kategori Admin.



Gambar 4.21 Bagan Alir Menghapus Data Kategori Admin

#### 4.2.13 Proses Alur Menampilkan Semua Pelatihan Pengajar

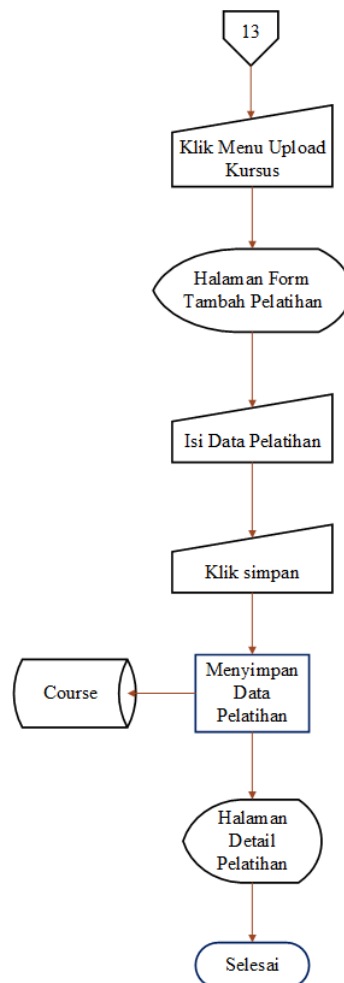
Proses Alur untuk menampilkan semua pelatihan pada sisi pengajar dapat dilihat pada gambar 4.19 Bagan Alir Menampilkan Semua Data Pelatihan Pengajar.



Gambar 4.22 Bagan Alir Menampilkan Semua Pelatihan Pengajar

#### 4.2.14 Proses Alur Membuat Pelatihan Oleh Pengajar

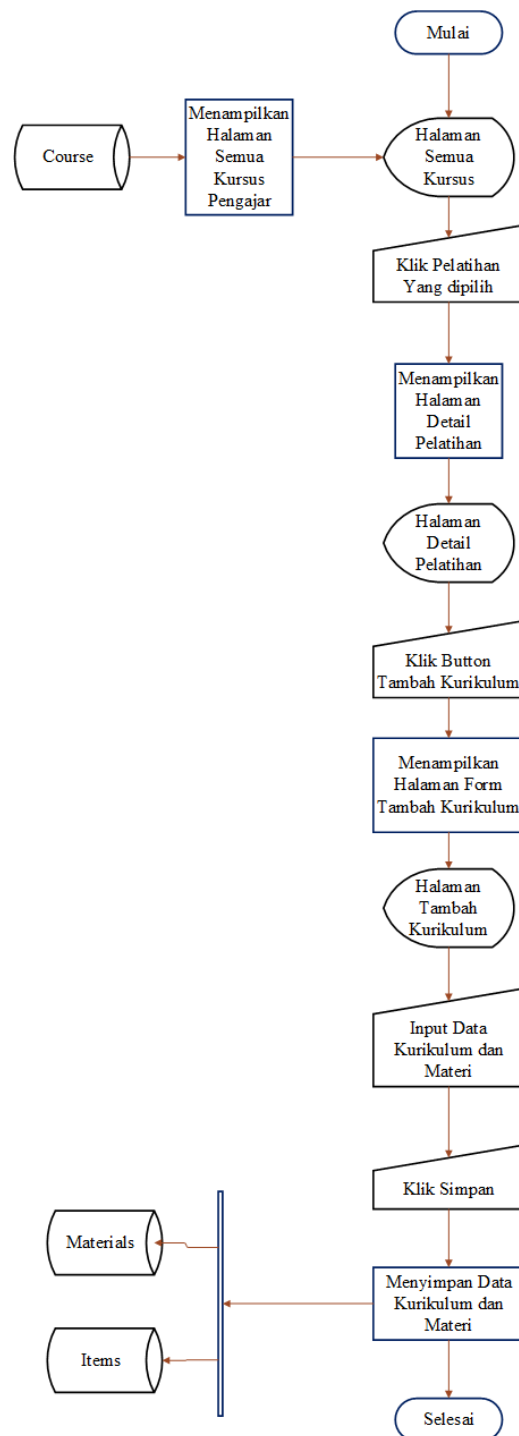
Proses Alur untuk membuat pelatihan pada sisi pengajar dapat dilihat pada gambar 4.20 Bagan Alir Membuat Pelatihan Pengajar.



Gambar 4.23 Bagan Alir Menambah Data Pelatihan Pengajar

#### 4.2.15 Proses Alur Menambah Data Materi Pelatihan Oleh Pengajar

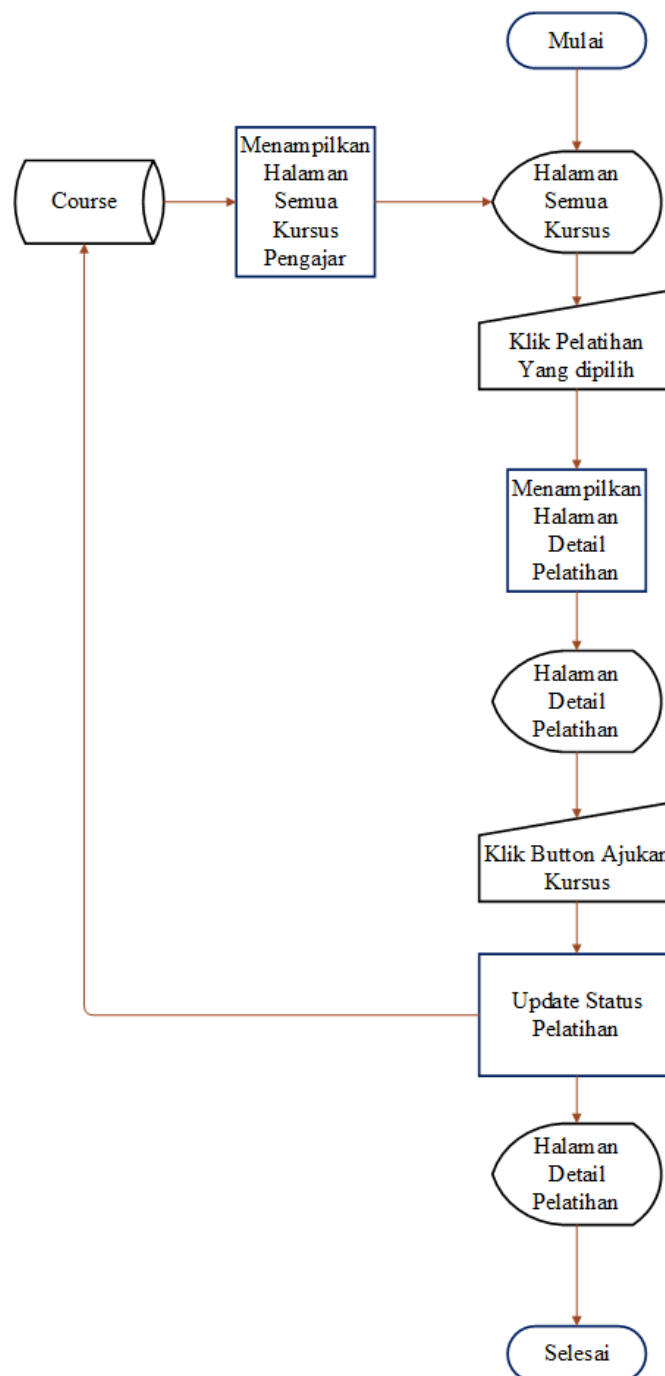
Proses Alur untuk mengajukan pelatihan pada sisi pengajar dapat dilihat pada gambar 4.21 Bagan Alir Menambah Data Materi Pelatihan Pengajar.



Gambar 4.24 Bagan Alir Menambah Data Materi Pelatihan Pengajar

#### 4.2.16 Proses Alur Mengajukan Pelatihan Oleh Pengajar

Proses Alur untuk mengajukan pelatihan pada sisi pengajar dapat dilihat pada gambar 4.22 Bagan Alir Mengajukan Pelatihan Pengajar.



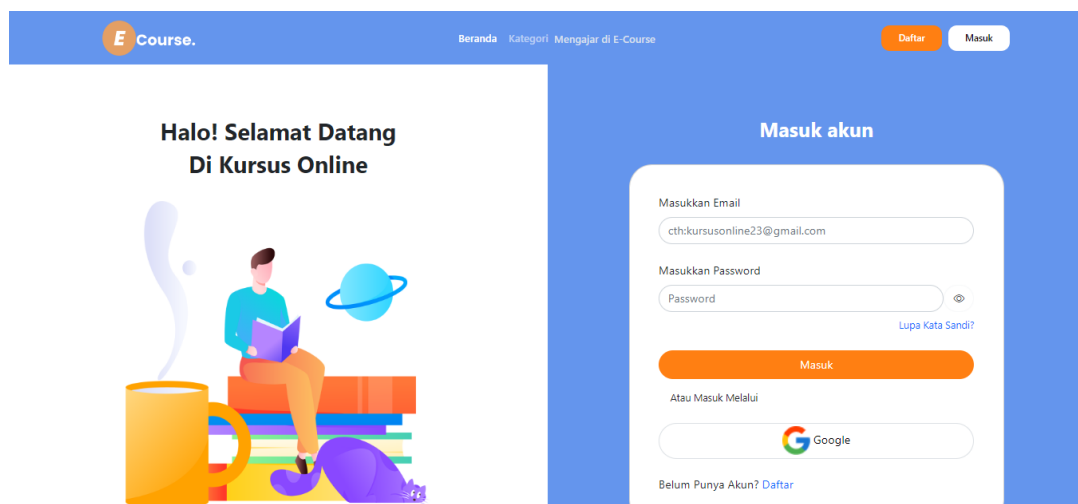
Gambar 4.25 Bagan Alir Mengajukan Pelatihan Pengajar

### 4.3 Uji Coba Sistem

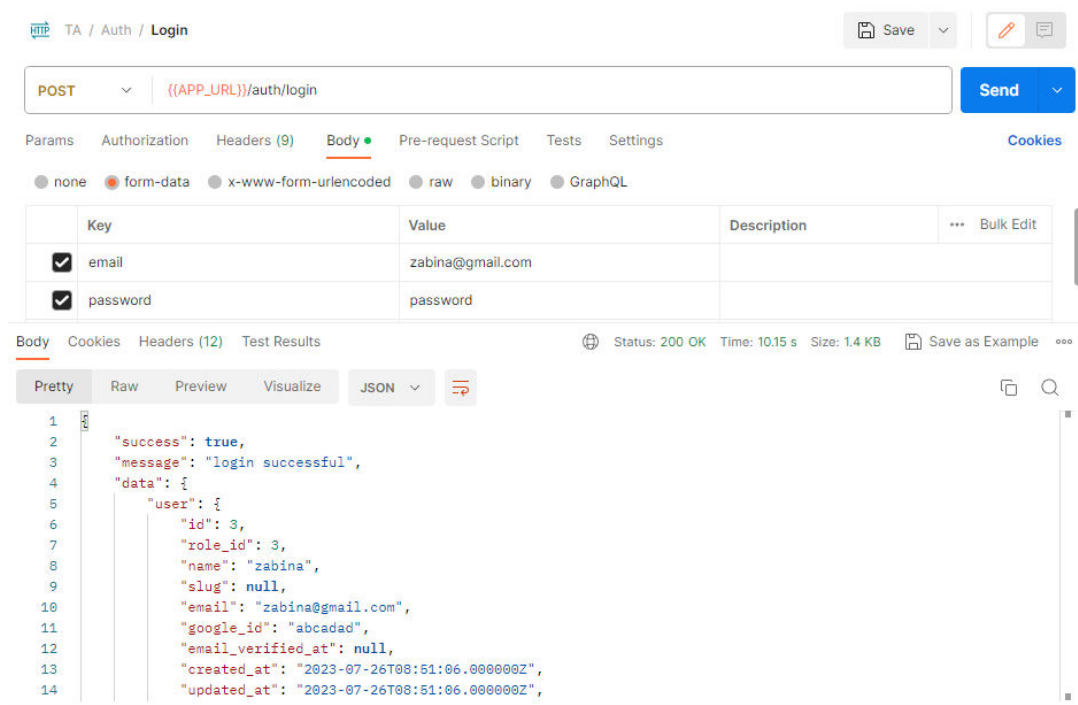
Pada bagian uji coba ini berisi pengujian E-Course dengan menggunakan metode *Black Box Testing*. *Black Box Testing* adalah pengujian yang dilakukan tanpa mengetahui struktur internal kode, hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian ini dilakukan oleh *back-end developer intern* yang meliputi pengujian REST-API E-Course.

#### 4.3.1 Pengujian Fitur *Login*

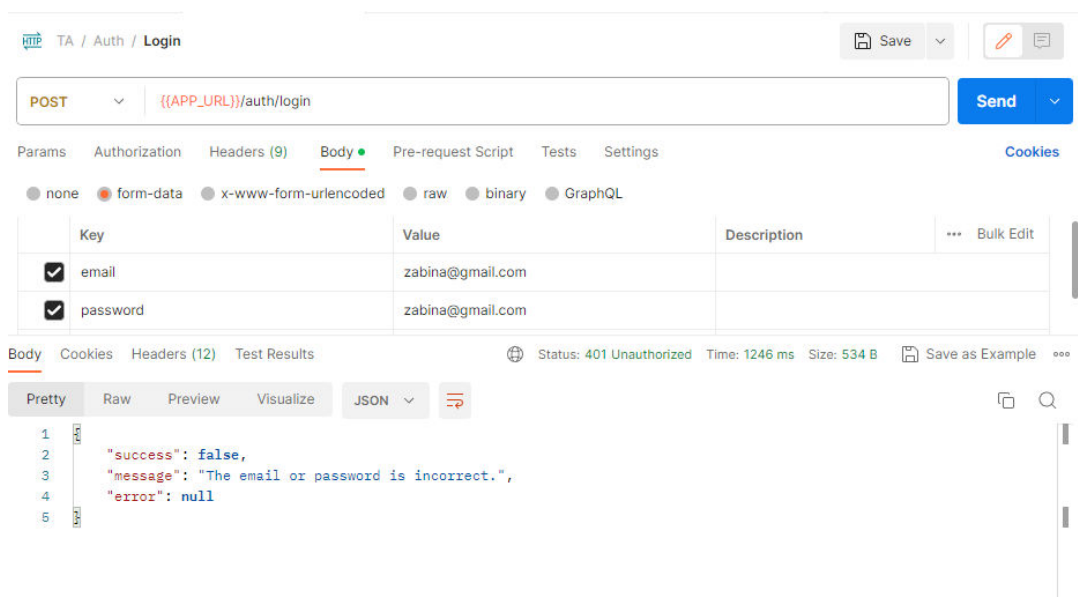
Pengujian fitur *login* dilakukan oleh pengguna sesuai dengan hak akses masing – masing pengguna. Terdapat tiga hak akses, yaitu : peserta/siswa, pengajar, dan admin. Fitur *login* dapat dilihat pada gambar 4.23. Pengujian menggunakan *postman* dapat dilihat pada gambar 4.24, gambar 4.25 dan gambar 4.26. Hasil pengujian fitur *login* dapat dilihat pada tabel 4.9.



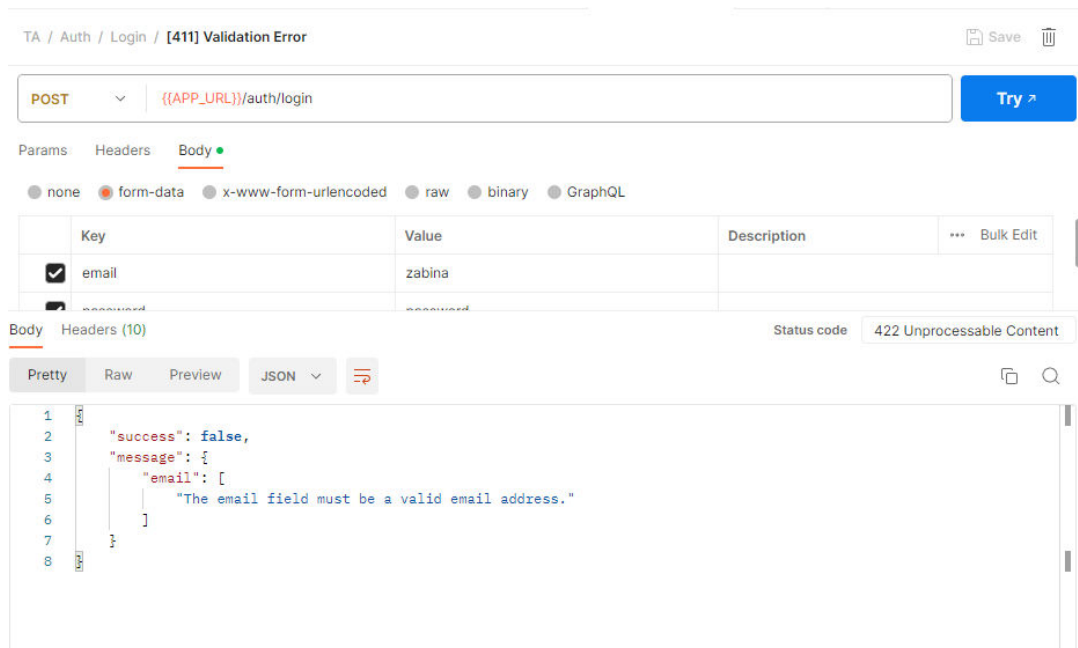
Gambar 4.26 Fitur *Login*



Gambar 4.27 Skema Pengujian Berhasil Login Menggunakan Postman



Gambar 4.28 Skema Pengujian Gagal Login 1 Menggunakan Postman



Gambar 4.29 Skema Pengujian Gagal Login 2 Menggunakan Postman

Tabel 4.10 Hasil Pengujian Fitur *Login*

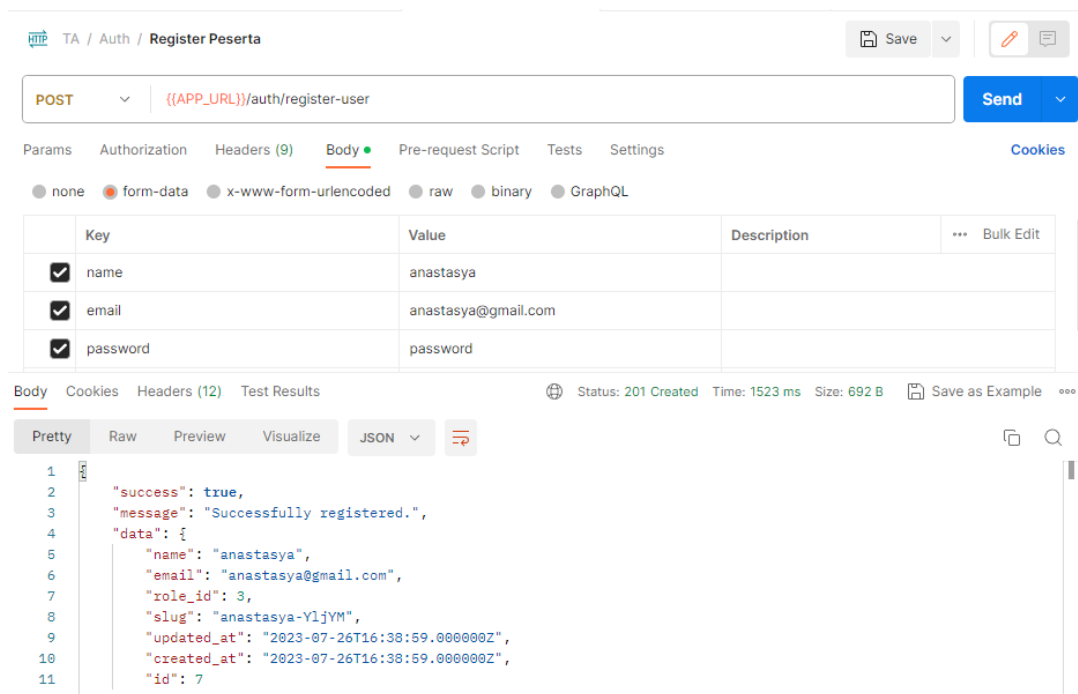
| No. | Nama Fitur             | <i>Input</i>                                  | <i>Output</i>                                            | Sesuai | Tidak Sesuai |
|-----|------------------------|-----------------------------------------------|----------------------------------------------------------|--------|--------------|
| 1.  | <i>Login</i> ke sistem | <i>email</i> dan <i>password</i> sesuai       | Menuju ke halaman sesuai dengan hak akses                | ✓      |              |
|     |                        | <i>email</i> dan <i>password</i> tidak sesuai | Terdapat pesan <i>username</i> dan <i>password</i> salah | ✓      |              |
|     |                        | <i>Email tidak valid</i>                      | Terdapat Pesan Email tidak valid                         | ✓      |              |

#### 4.3.2 Pengujian Fitur Registrasi

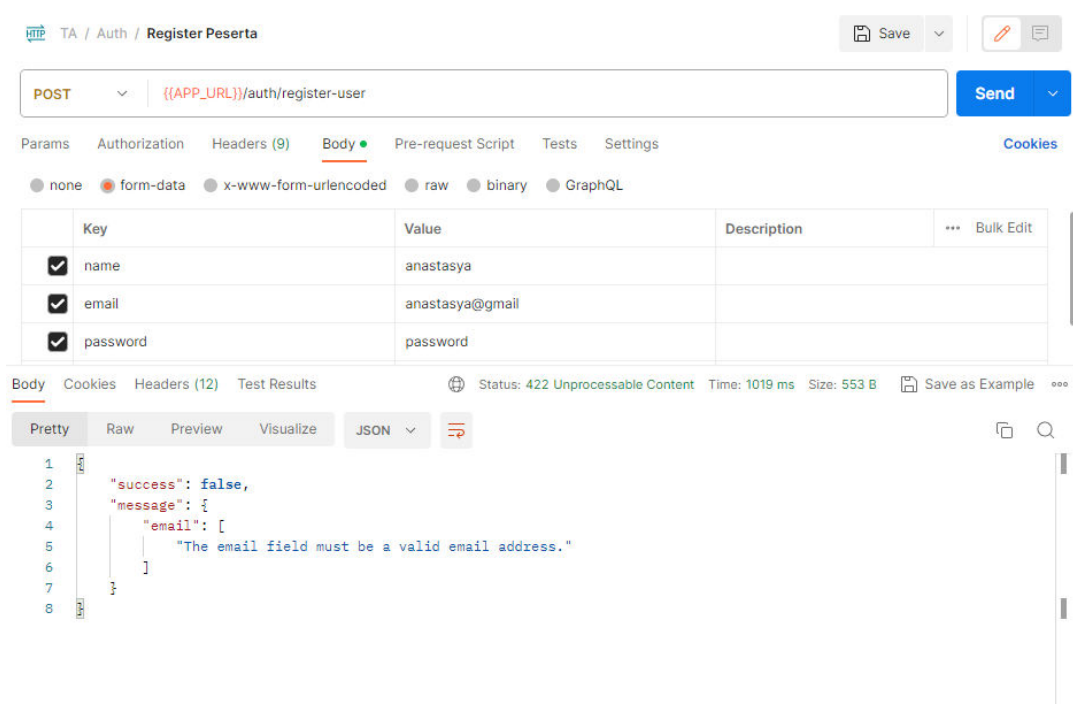
Pengujian fitur registrasi dilakukan oleh pengguna sesuai dengan hak akses masing – masing pengguna. Terdapat tiga hak akses, yaitu : peserta/siswa, pengajar, dan admin. Fitur login dapat dilihat pada gambar 4.27. Pengujian menggunakan

postman dapat dilihat pada gambar 4.28 dan gambar 4.29. Hasil pengujian fitur registrasi dapat dilihat pada tabel 4.10

Gambar 4.30 Fitur Registrasi



Gambar 4.31 Skema Pengujian Berhasil Menggunakan Postman



Gambar 4.32 Skema Pengujian Gagal Menggunakan Postman

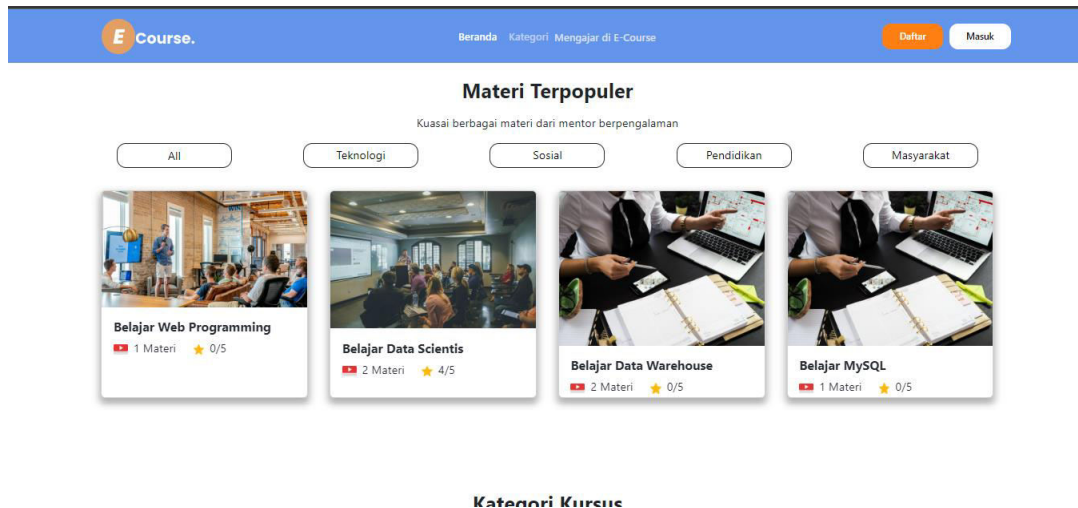
Tabel 4.11 Hasil Pengujian Proses Registrasi

| No. | Nama Fitur | <i>Input</i>                                 | <i>Output</i>                                     | Sesuai | Tidak Sesuai |
|-----|------------|----------------------------------------------|---------------------------------------------------|--------|--------------|
| 1.  | Registrasi | Nama, <i>Email</i> dan <i>password</i> valid | Registrasi Berhasil dan data User berhasil dibuat | ✓      |              |
|     |            | <i>Email</i> tidak valid                     | Terdapat pesan bahwan email tidak valid           | ✓      |              |

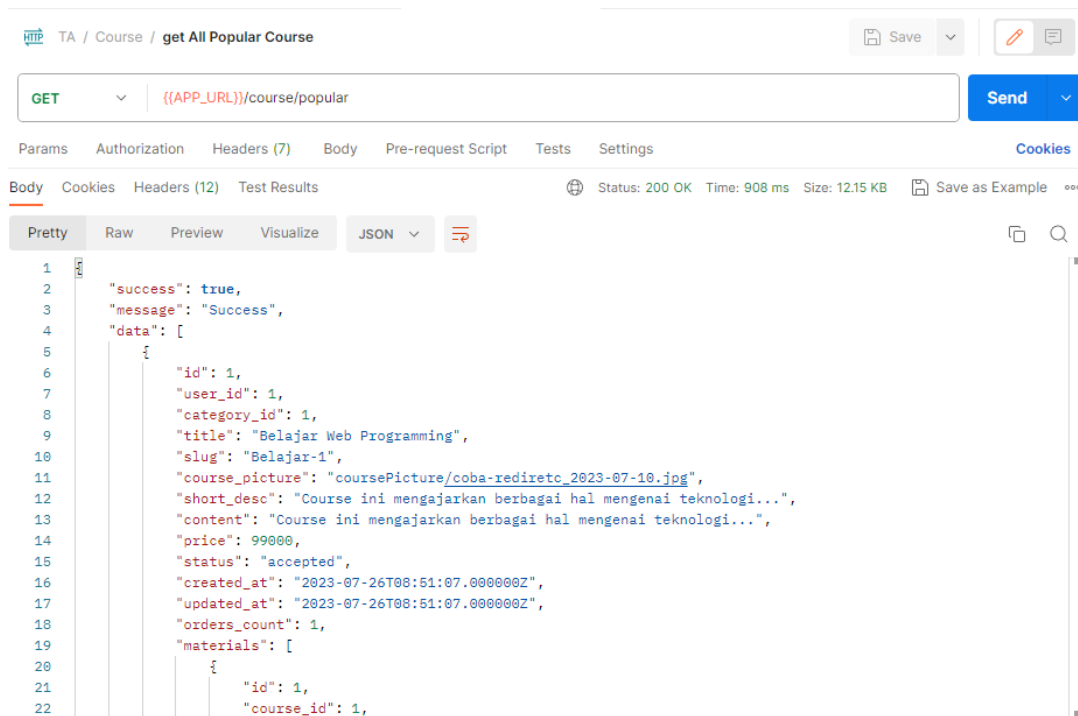
### 4.3.3 Pengujian Penggunaan API untuk Menampilkan Data Pelatihan

Pengujian penggunaan API untuk menampilkan data pelatihan digunakan pada sisi siswa/pengajar agar dapat melihat semua pelatihan yang telah disetujui oleh admin. Fitur list pelatihan dapat dilihat pada gambar 4.30. Pengujian menggunakan

postman dapat dilihat pada gambar 4.31. Hasil pengujian fitur registrasi dapat dilihat pada tabel 4.11



Gambar 4.33 Fitur List Pelatihan



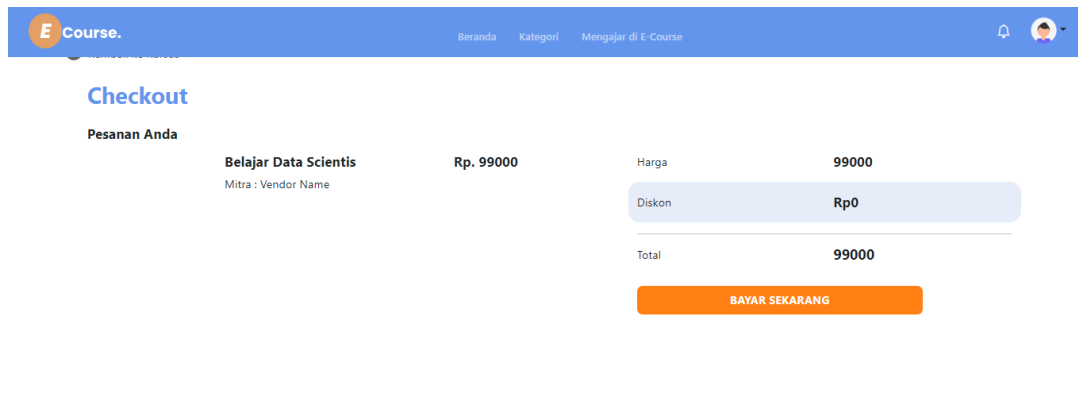
Gambar 4.34 Pengujian API Pelatihan Menggunakan API

Tabel 4.12 Hasil Pengujian API Pelatihan

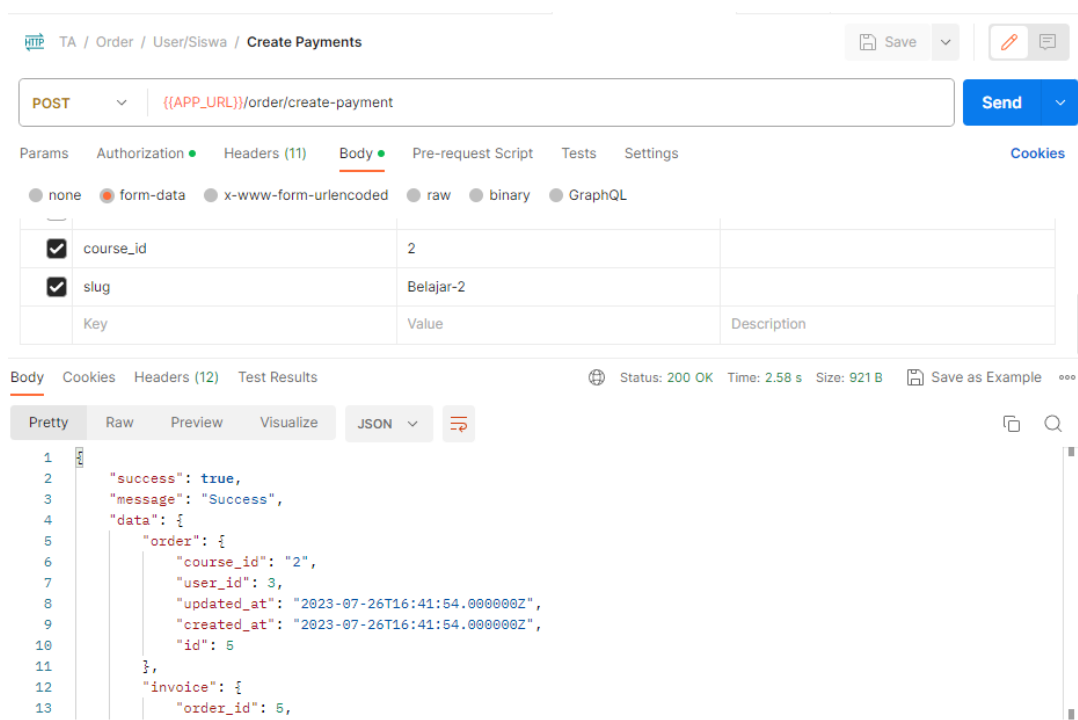
| No. | Nama Fitur                                                | <i>Input</i> | <i>Output</i>                                           | Sesuai | Tidak Sesuai |
|-----|-----------------------------------------------------------|--------------|---------------------------------------------------------|--------|--------------|
| 1.  | Menampilkan data pelatihan yang telah di verifikasi admin | -            | Dapat mengambil semua pelatihan yang telah diverifikasi | ✓      |              |

#### 4.3.4 Pengujian Proses Pembelian Pelatihan

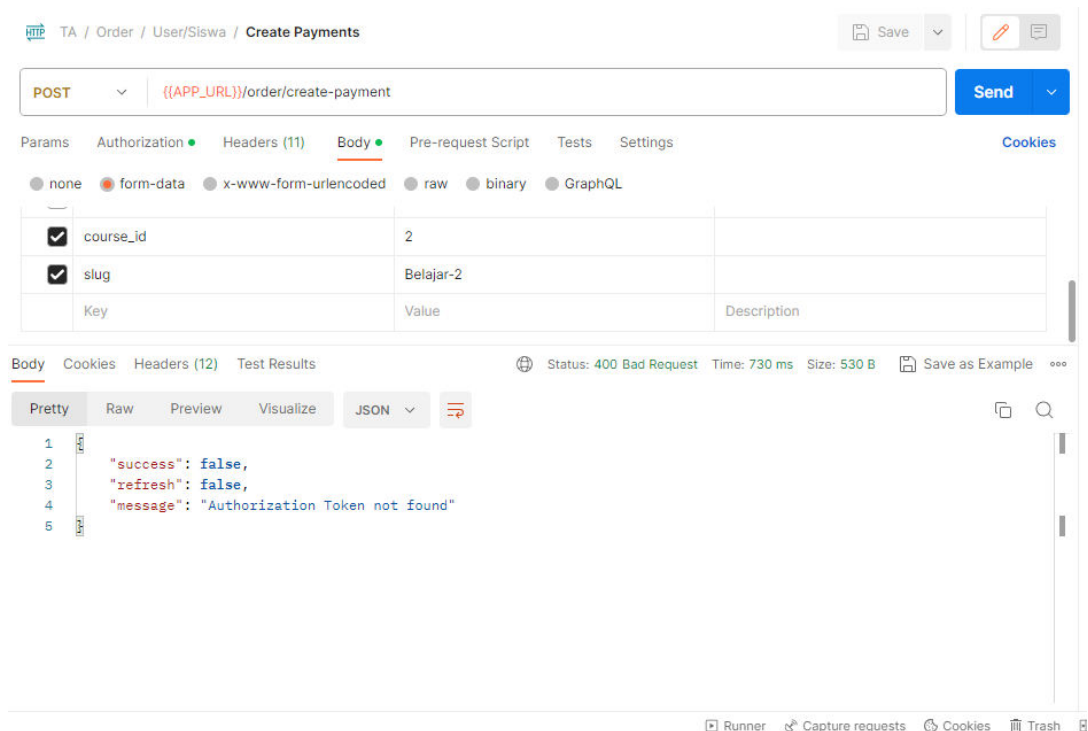
Pengujian proses pembelian pelatihan dilakukan oleh siswa/peserta melalui halaman checkout tanpa mengirim data pada form. Pengujian ini dilakukan apakah peserta dapat membeli kursus dengan benar dan memastikan bahwa database transaction dapat berjalan lancar. Fitur pembelian pelatihan dapat dilihat pada gambar 4.32. Pengujian menggunakan postman dapat dilihat pada gambar 4.33, gambar 4.34 dan gambar 4.35. Hasil pengujian fitur pembelian dapat dilihat pada tabel 4.12



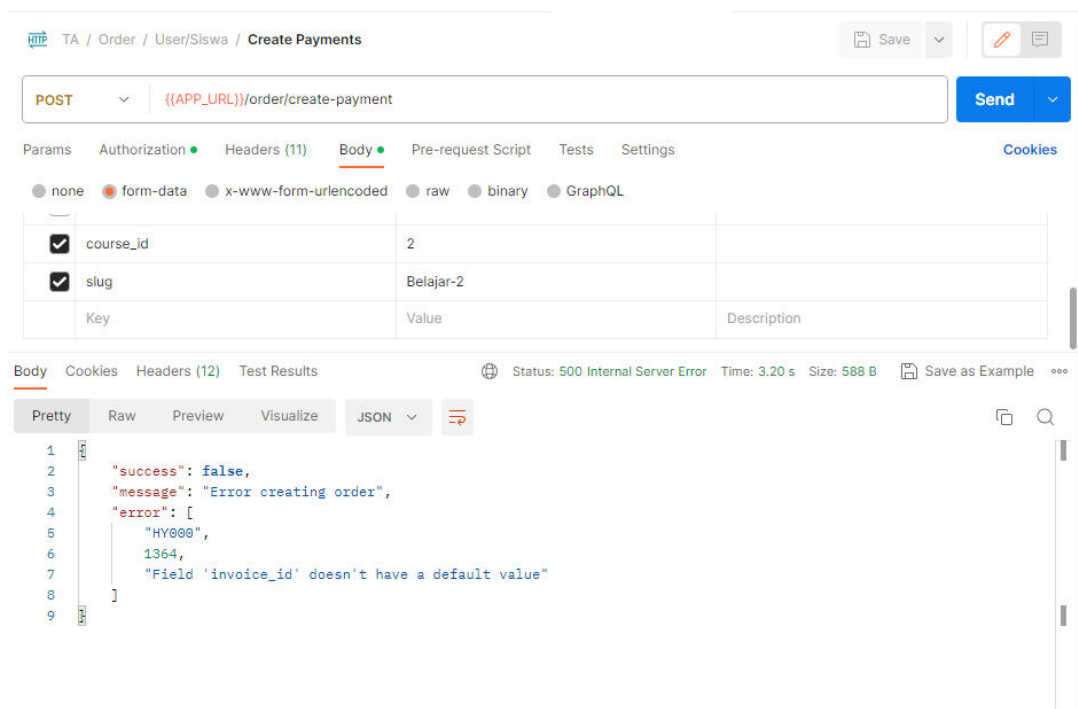
Gambar 4.35 Fitur Pembelian Pelatihan



Gambar 4.36 Skema Pengujian Pembelian Berhasil Menggunakan Postman



Gambar 4.37 Skema Pengujian Pembelian Gagal 1 Menggunakan Postman



Gambar 4.38 Skema Pengujian Pembelian Gagal 2 Menggunakan Postman

Tabel 4.13 Hasil Pengujian Proses Pembelian Pelatihan

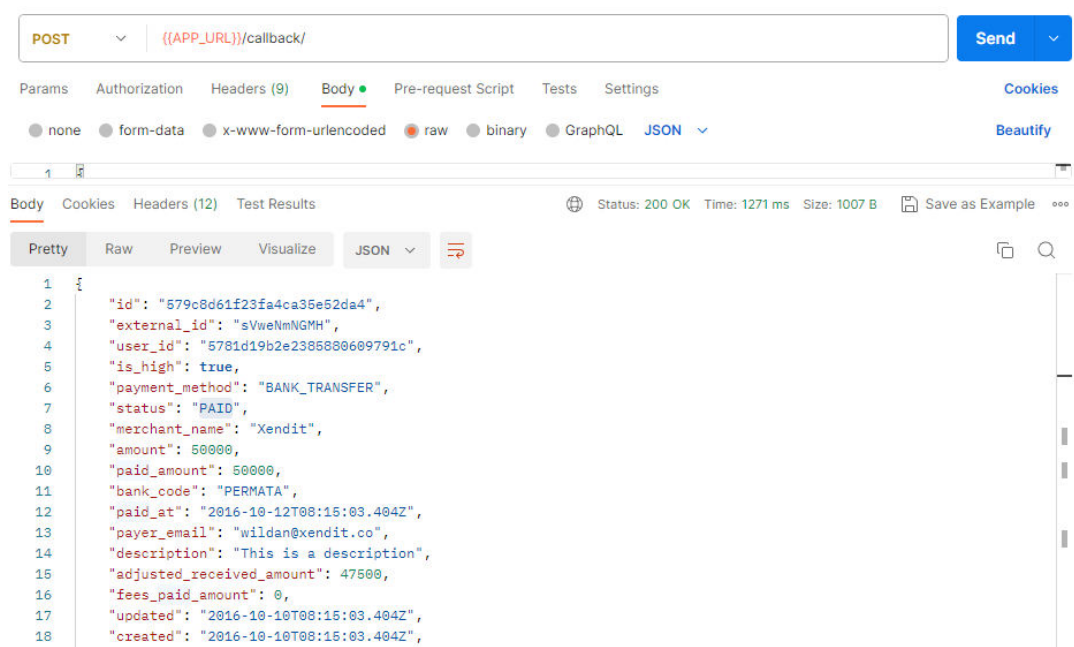
| No. | Nama Fitur          | <i>Input</i>                       | <i>Output</i>                                                            | Sesuai | Tidak Sesuai |
|-----|---------------------|------------------------------------|--------------------------------------------------------------------------|--------|--------------|
| 1.  | Pembelian Pelatihan | <i>Slug</i> Pelatihan, Sudah Login | Berhasil Mendapatkan Link Pembayaran, dan data pembayaran sukses terbuat | ✓      |              |
|     |                     | <i>Slug</i> Pelatihan, Belum Login | Terdapat pesan “Authorization Token not found”                           | ✓      |              |

#### 4.3.5 Pengujian Proses *Callback*

Proses *callback* merupakan proses dimana sistem dapat menerima pemberitahuan jika terdapat pembaruan data dari *payment gateway* xendit. Pengujian proses *callback* dilakukan menggunakan postman untuk memastikan bahwa sistem dapat menerima pembaruan data pada database sesuai dengan request dari xendit. Data sebelum dilakukan pengujian dapat dilihat pada gambar 4.36 dan proses pengujian dapat dilihat pada gambar 4.37 Pengujian menggunakan postman dapat dilihat pada gambar 4.33. Hasil pengujian fitur pembelian dapat dilihat pada tabel 4.13.



Gambar 4.39 Data Pembayaran Berstatus Pending Sebelum Pengujian



Gambar 4.40 Pengujian Callback

Pada proses pengujian diatas, data pembayaran berubah status menjadi paid

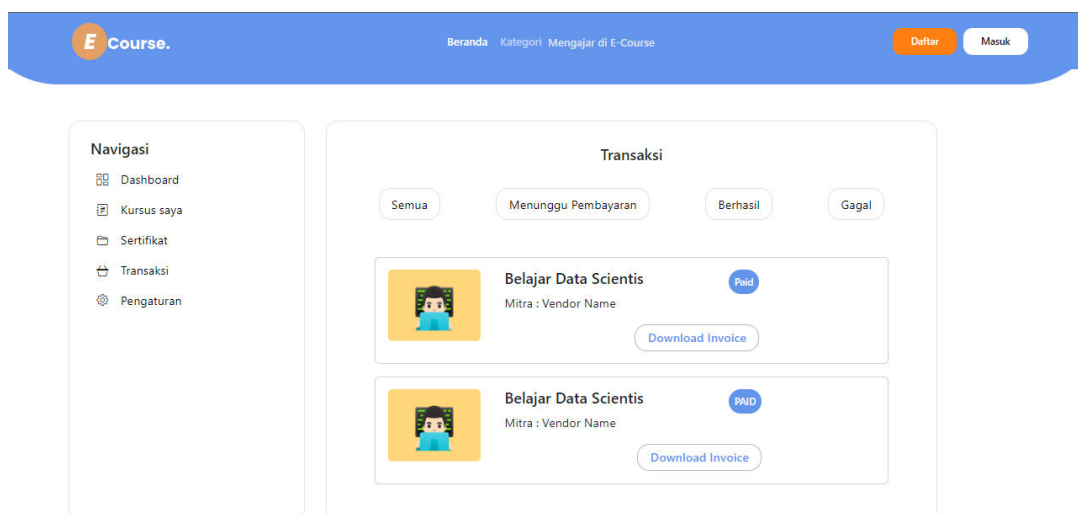
Ketika sistem menerima callback pembayaran dari *payment gateway* sesuai dengan `external_id` pada tabel payments.

Tabel 4.14 Hasil Pengujian Proses Pembelian Pelatihan

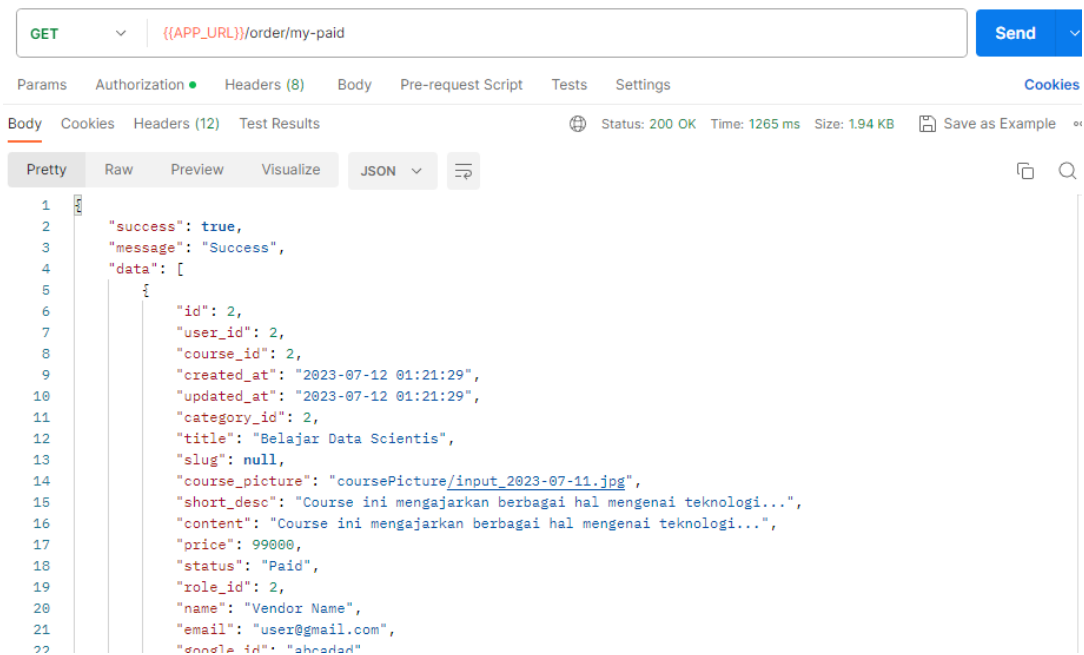
| No. | Nama Fitur      | <i>Input</i>          | <i>Output</i>                                              | Sesuai | Tidak Sesuai |
|-----|-----------------|-----------------------|------------------------------------------------------------|--------|--------------|
| 1.  | Proses Callback | Data JSON dari xendit | Berhasil merubah status paymen sesuai request dari xendit. | ✓      |              |

#### 4.3.6 Pengujian API Histori Pembayaran

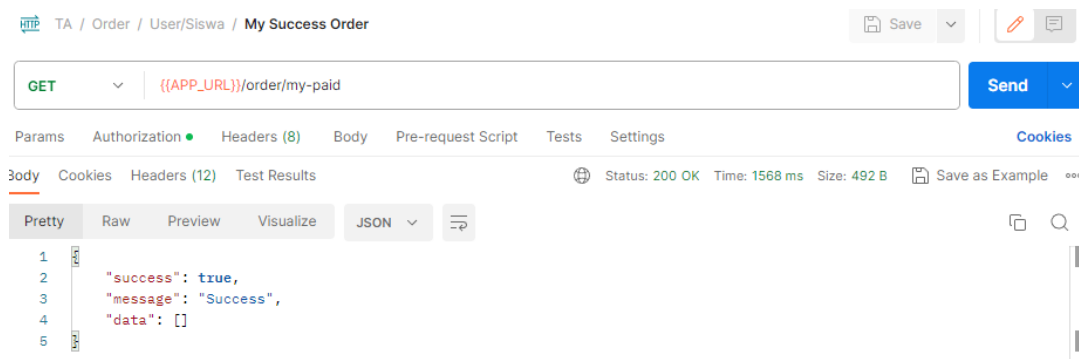
API histori pembayaran digunakan untuk dapat mengambil data pembayaran dari masing masing user yang login sesuai dengan status pembayaran yang diinginkan. Pada proses pengujian API histori pembayaran ini akan dilakukan pengujian histori pembayaran yang selesai dibayar dalam dua kondisi yaitu ketika terdapat data yang valid dan ketika tidak ditemukan data yang valid. Fitur histori pembayaran dapat dilihat pada gambar 4.38. Pengujian ketika terdapat data yang valid dapat dilihat pada gambar 4.39 dan pengujian ketika tidak terdapat data yang valid dapat dilihat pada gambar 4.40. Hasil pengujian API histori pembayaran dapat dilihat pada tabel 4.14.



Gambar 4.41 Fitur Histori Pembayaran



Gambar 4.42 Pengujian Ketika Terdapat Data yang Valid



Gambar 4.43 Pengujian Ketika Tidak Terdapat Data yang Valid

Tabel 4.15 Hasil Pengujian Proses Pembelian Pelatihan

| No. | Nama Fitur                | Input                   | Output                                    | Sesuai | Tidak Sesuai |
|-----|---------------------------|-------------------------|-------------------------------------------|--------|--------------|
| 1.  | Proses Histori Pembayaran | Bearer Token User Login | Data Histori Pembayaran Sesuai User Login | ✓      |              |

#### 4.3 Analisis Hasil Pengujian

Berdasarkan pengujian terhadap *backend learning management system* E-Course, dapat disimpulkan bahwa pengembangan *back-end* sistem berjalan dengan baik dan sesuai dengan kebutuhan data serta dapat mengatasi permasalahan yang telah dianalisis pada bab sebelumnya, antara lain :

1. Pada proses pelatihan konvensional, proses pelatihan hanya dapat dilakukan pada rentang waktu dan tempat tertentu, Dengan adanya sistem E-Course ini, proses pembelajaran dapat dilakukan kapan saja dan dimana saja secara efektif.
2. Proses pencatatan dan perekapan data kurang efektif karena harus mencatat pada beberapa kertas dan merekap data satu persatu sehingga menghabiskan tenaga dan waktu yang cukup banyak.
3. Proses Pembayaran pada pelatihan konvensional memerlukan proses yang cukup Panjang. Dengan adanya sistem E-Course ini proses pembayaran dapat dipersingkat karena metode hingga proses pembayaran ditangani oleh *payment gateway* xendit.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Tugas akhir ini disusun untuk dapat mengembangkan *backend learning management system E-Course* mulai dari proses pendaftaran, pelaksanaan pelatihan, hingga proses pengelolaan data pelatihan yang dapat memecahkan masalah pada pelaksanaan pelatihan konvensional. Berdasarkan hasil pengelolaan data dan analisis yang dibahas pada bab – bab sebelumnya, dapat diambil kesimpulan sebagai berikut:

1. Pembuatan *backend learning management system E-Course* ini diawali dengan menganalisa sistem kerja pelatihan konvensional dengan menggunakan docflow.
2. Kemudian untuk membangun *backend learning management system E-Course* yang dapat mengatasi permasalahan sebelumnya yakni dengan mengidentifikasi penyebab dari suatu masalah dan menganalisa apa yang sesungguhnya terjadi dalam proses tersebut dengan menggunakan Fishbone Diagram. Selanjutnya melakukan perancangan desain sistem. Yaitu dengan merancang desain proses berupa DFD yang terdiri dari 4 proses utama. Merancang desain database berupa CDM yang terdiri dari 19 entitas dan PDM yang terdiri dari 19 tabel namun lingkup pengerjaan pada tugas akhir ini hanya menggunakan 12 tabel, kemudian dilakukan generate pada aplikasi basis data. Serta membuat desain input dan output setiap proses yang ada. Aplikasi ini dibangun menggunakan pemrograman website yang berbasis

framework Laravel untuk backend dan frontend admin serta React.js untuk frontend siswa/pengajar.

## **5.2 Saran**

Dari hasil E-Course ini maka, penulis sarankan untuk kedepannya sistem ini perlu dikembangkan lebih lanjut sehingga dapat menjadi sistem yang dapat menangani proses kegiatan pelatihan atau pembelajaran secara digital dengan lebih baik dan masih perlu banyak pengembangan lebih lanjut agar dapat menyesuaikan dengan perkembangan teknologi dan kebutuhan baru kedepannya.

## DAFTAR PUSTAKA

- Chusna, N. L. (2019). Pembelajaran E-learning. In Prosiding Seminar Nasional Pendidikan KALUNI (Vol. 2, No. 1, pp. 113-117).
- Haryanto, S. (2018). Kelebihan dan kekurangan E-learning berbasis schoology (studi PTK dalam pembejaran mata kuliah academic listening). Seminar Nasional GEOTIK 2018.
- Jogiyanto, HM. 2005. Analisa dan Desain Sistem. Penerbit Andi: Yogyakarta.
- Whitten. Jeffrey L, Lonnie D. Bantley, Kevin C. Dittman. 2004. System Analysis and Design Methods. edisi 1. Yogyakarta : Andi.
- Whitten, Jeffery L., Lonnie D. Bentley, Kevin C. Dittman. 2004. “Metode Desain dan Analisis Sistem Edisi Enam. Yogyakarta. Mc Graw Hill-Andi.

**LAMPIRAN I**  
**DOKUMEN API DOCUMENTATION**

API Documentation

# E-Course API

## Daftar Isi

### Daftar Isi

|                                         |    |
|-----------------------------------------|----|
| <b>Daftar Isi</b> .....                 | 1  |
| <b>Login API</b> .....                  | 3  |
| <b>Login</b> .....                      | 3  |
| Headers .....                           | 3  |
| Request Body .....                      | 3  |
| Responses .....                         | 3  |
| Example Responses 200 .....             | 4  |
| Example Responses 401 .....             | 5  |
| <b>Register API</b> .....               | 5  |
| <b>Register</b> .....                   | 5  |
| Headers .....                           | 5  |
| Request Body .....                      | 5  |
| Responses .....                         | 6  |
| Example Responses 200 .....             | 6  |
| Example Responses 422 .....             | 6  |
| <b>Category API</b> .....               | 7  |
| <b>Category</b> .....                   | 7  |
| Headers .....                           | 7  |
| Responses .....                         | 7  |
| Example Responses 200 .....             | 8  |
| <b>Get All Course API</b> .....         | 8  |
| <b>Get All Course</b> .....             | 8  |
| Headers .....                           | 8  |
| Responses .....                         | 9  |
| Example Responses 200 .....             | 9  |
| <b>Get Course By category API</b> ..... | 10 |



|                                     |    |
|-------------------------------------|----|
| <b>Get Course By Category</b> ..... | 10 |
| Headers .....                       | 11 |
| Responses .....                     | 11 |
| Example Responses 200 .....         | 11 |
| <b>Get Detail Course API</b> .....  | 13 |
| <b>Get Detail Course</b> .....      | 13 |
| Headers .....                       | 13 |
| Responses .....                     | 13 |
| Example Responses 200 .....         | 13 |
| <b>Create Payment API</b> .....     | 15 |
| <b>Payment</b> .....                | 15 |
| Headers .....                       | 15 |
| Request Body .....                  | 15 |
| Responses .....                     | 15 |
| Example Responses 200 .....         | 16 |
| Example Responses 400 .....         | 17 |
| Example Responses 500 .....         | 17 |
| <b>callback API</b> .....           | 17 |
| <b>callback</b> .....               | 17 |
| Headers .....                       | 18 |
| Example Request .....               | 18 |
| Example Responses 200 .....         | 19 |



### Login API

#### Login

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk login  
Apps : E-Course

|          |                        |
|----------|------------------------|
| Endpoint | {{APP_URL}}/auth/login |
| Method   | <b>POST</b>            |

#### Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |

#### Request Body

| Attribute | Type   | Required | Description                         |
|-----------|--------|----------|-------------------------------------|
| email     | string | y        | Email yang digunakan user login     |
| password  | string | y        | Password yang digunakan untuk login |

#### Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data user login                      |





## Example Responses 401

```
{
  "success": false,
  "message": "The email or password is incorrect.",
  "error": null
}
```

**Register API****Register**

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk Registrasi

Apps : E-Course

|          |                                |
|----------|--------------------------------|
| Endpoint | {{APP_URL}}/auth/register-user |
| Method   | <b>POST</b>                    |

## Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |

## Request Body

| Attribute | Type   | Required | Description   |
|-----------|--------|----------|---------------|
| Name      | string | y        | Nama User     |
| Email     | string | y        | Email user    |
| Password  | string | y        | Password user |



## Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data Registrasi User                 |

## Example Responses 200

```
{
  "success": true,
  "message": "Successfully registered.",
  "data": {
    "name": "rahmad",
    "email": "rahmad@gmail.com",
    "role_id": 3,
    "slug": "rahmad-NKx48",
    "updated_at": "2023-07-08T13:19:32.000000Z",
    "created_at": "2023-07-08T13:19:32.000000Z",
    "id": 8
  }
}
```

## Example Responses 422

```
{
  "success": false,
  "message": {
    "email": [
      "The email field must be a valid email address."
    ]
  }
}
```



```

    ]
  }
}

```

### Category API

#### Category

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk mengambil data kategori pelatihan

Apps : E-Course

|          |                      |
|----------|----------------------|
| Endpoint | {{APP_URL}}/category |
| Method   | GET                  |

#### Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |

#### Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data Kategori Pelatihan              |



## Example Responses 200

```
{
  "success": true,
  "message": "Success",
  "data": [
    {
      "id": 1,
      "title": "Teknologi",
      "description": "Course ini mengajarkan berbagai hal mengenai teknologi...",
      "category_picture": "ini Gambar.jpg",
      "created_at": "2023-07-08T13:46:07.000000Z",
      "updated_at": "2023-07-08T13:46:07.000000Z",
      "courses": [
```

**Get All Course API****Get All Course**

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk mengambil data pelatihan yang aktif

Apps : E-Course

|          |                     |
|----------|---------------------|
| Endpoint | {{APP_URL}}/course/ |
| Method   | Get                 |

## Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |



## Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data Registrasi User                 |

## Example Responses 200

```
{
  "success": true,
  "message": "Success",
  "data": [
    {
      "id": 1,
      "user_id": 1,
      "category_id": 1,
      "title": "Belajar Web Programming",
      "slug": "Belajar-1",
      "course_picture": "buat foro.png",
      "short_desc": "Course ini mengajarkan berbagai hal mengenai teknologi...",
      "content": "Course ini mengajarkan berbagai hal mengenai teknologi...",
      "price": 99000,
      "status": 1,
      "created_at": "2023-07-08T13:20:04.000000Z",
      "updated_at": "2023-07-08T13:20:04.000000Z",
      "materials": [
        {
          "id": 1,
          "course_id": 1,
          "title": "bab 1",
          "slug": "bab-1",

```



```

    "created_at": "2023-07-08T13:20:04.000000Z",
    "updated_at": "2023-07-08T13:20:04.000000Z",
    "items": [
      {
        "id": 1,
        "material_id": 1,
        "name": "Video Materi 1",
        "slug": null,
        "source": "www.youtube.com",
        "created_at": "2023-07-08T13:20:04.000000Z",
        "updated_at": "2023-07-08T13:20:04.000000Z"
      }
    ]
  },
  "user": {
    "id": 1,
    "name": "admin",
    "email": "admin@gmail.com"
  },
  "reviews": []
},

```

### Get Course By category API

#### Get Course By Category

**Purpose** : Bagian Ini Menjelaskan API yang digunakan untuk mengambil data pelatihan berdasarkan category yang aktif

**Apps** : E-Course

|          |                                 |
|----------|---------------------------------|
| Endpoint | {{APP_URL}}/course/popular/{id} |
|----------|---------------------------------|



|        |     |
|--------|-----|
| Method | Get |
|--------|-----|

## Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |

## Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data Registrasi User                 |

## Example Responses 200

```
{
  "success": true,
  "message": "Success",
  "data": [
    {
      "id": 1,
      "user_id": 1,
      "category_id": 2,
      "title": "Belajar Web Programming",
      "slug": "Belajar-1",
      "course_picture": "buat foro.png",
      "short_desc": "Course ini mengajarkan berbagai hal mengenai teknologi...",
      "content": "Course ini mengajarkan berbagai hal mengenai teknologi..."
    }
  ]
}
```



```
"price": 99000,
"status": 1,
"created_at": "2023-07-08T13:20:04.000000Z",
"updated_at": "2023-07-08T13:20:04.000000Z",
"materials": [
  {
    "id": 1,
    "course_id": 1,
    "title": "bab 1",
    "slug": "bab-1",
    "created_at": "2023-07-08T13:20:04.000000Z",
    "updated_at": "2023-07-08T13:20:04.000000Z",
    "items": [
      {
        "id": 1,
        "material_id": 1,
        "name": "Video Materi 1",
        "slug": null,
        "source": "www.youtube.com",
        "created_at": "2023-07-08T13:20:04.000000Z",
        "updated_at": "2023-07-08T13:20:04.000000Z"
      }
    ]
  }
],
"user": {
  "id": 1,
  "name": "admin",
  "email": "admin@gmail.com"
},
"reviews": []
},
```



### Get Detail Course API

#### Get Detail Course

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk mengambil data pelatihan berdasarkan category yang aktif

Apps : E-Course

|          |                                |
|----------|--------------------------------|
| Endpoint | {APP_URL}/course/detail/{slug} |
| Method   | Get                            |

#### Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |

#### Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data Registrasi User                 |

#### Example Responses 200

```
{
  "success": true,
  "message": "Success",
  "data": {
```



```
{
  "id": 6,
  "user_id": 3,
  "category_id": 3,
  "title": "Belajar Oracle",
  "slug": "Belajar-6",
  "course_picture": "buat foro.png",
  "short_desc": "Course ini mengajarkan berbagai hal mengenai teknologi...",
  "content": "Course ini mengajarkan berbagai hal mengenai teknologi...",
  "price": 99000,
  "status": 0,
  "created_at": "2023-07-08T17:19:57.000000Z",
  "updated_at": "2023-07-08T17:19:57.000000Z",
  "materials": [
    {
      "id": 8,
      "course_id": 6,
      "title": "bab 4",
      "slug": "bab-8",
      "created_at": "2023-07-08T17:19:57.000000Z",
      "updated_at": "2023-07-08T17:19:57.000000Z",
      "items": []
    }
  ],
  "user": {
    "id": 3,
    "name": "zabina",
    "email": "zabina@gmail.com"
  },
  "reviews": []
}
```



### Create Payment API

#### Payment

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk membuat data pembelian pelatihan

Apps : E-Course

|          |                                  |
|----------|----------------------------------|
| Endpoint | {{APP_URL}}/order/create-payment |
| Method   | <b>POST</b>                      |

#### Headers

| Key           | Value            | Description                  |
|---------------|------------------|------------------------------|
| Authorization | Bearer xxxx      | Bearer token dari user login |
| Content-Type  | application/json |                              |

#### Request Body

| Attribute | Type   | Required | Description    |
|-----------|--------|----------|----------------|
| Slug      | string | y        | Slug Pelatihan |

#### Responses

| Attribute | Data Type | Description                          |
|-----------|-----------|--------------------------------------|
| Success   | string    | status Respon API (true atau false ) |
| message   | string    | Pesan Dari respon                    |
| data      | object    | Data Registrasi User                 |



## Example Responses 200

```
{
  "success": true,
  "message": "Success",
  "data": {
    "order": {
      "course_id": "2",
      "user_id": 4,
      "updated_at": "2023-07-08T22:08:22.000000Z",
      "created_at": "2023-07-08T22:08:22.000000Z",
      "id": 10
    },
    "invoice": {
      "discount": "0",
      "amount": "55000",
      "order_id": 10,
      "updated_at": "2023-07-08T22:08:22.000000Z",
      "created_at": "2023-07-08T22:08:22.000000Z",
      "id": 5
    },
    "payment": {
      "note": "notee",
      "status": "PENDING",
      "invoice_id": 5,
      "external_id": "ge87VkY3kc",
      "updated_at": "2023-07-08T22:08:23.000000Z",
      "created_at": "2023-07-08T22:08:23.000000Z",
      "id": 4
    }
  }
}
```



## Example Responses 400

```
{
  "success": false,
  "refresh": false,
  "message": "Authorization Token not found"
}
```

## Example Responses 500

```
{
  "success": false,
  "message": "Error creating order",
  "error": [
    "HY000",
    1364,
    "Field 'invoice_id' doesn't have a default value"
  ]
}
```

**Callback API****Callback**

Purpose : Bagian Ini Menjelaskan API yang digunakan untuk mendapat data pembaruan dari payment gateway xendit

Apps : E-Course

|          |                      |
|----------|----------------------|
| Endpoint | {{APP_URL}}/callback |
| Method   | <b>POST</b>          |



## Headers

| Key          | Value            | Description |
|--------------|------------------|-------------|
| Content-Type | application/json |             |

## Example Request

```
{
  "id": "579c8d61f23fa4ca35e52da4",
  "external_id": "NJ0Wvx3CLP",
  "user_id": "5781d19b2e2385880609791c",
  "is_high": true,
  "payment_method": "BANK_TRANSFER",
  "status": "PAID",
  "merchant_name": "Xendit",
  "amount": 50000,
  "paid_amount": 50000,
  "bank_code": "PERMATA",
  "paid_at": "2016-10-12T08:15:03.404Z",
  "payer_email": "wildan@xendit.co",
  "description": "This is a description",
  "adjusted_received_amount": 47500,
  "fees_paid_amount": 0,
  "updated": "2016-10-10T08:15:03.404Z",
  "created": "2016-10-10T08:15:03.404Z",
  "currency": "IDR",
  "payment_channel": "PERMATA",
  "payment_destination": "888888888888"
}
```



## Example Responses 200

```
{
  "id": "579c8d61f23fa4ca35e52da4",
  "external_id": "NJ0Wvx3CLP",
  "user_id": "5781d19b2e2385880609791c",
  "is_high": true,
  "payment_method": "BANK_TRANSFER",
  "status": "PAID",
  "merchant_name": "Xendit",
  "amount": 50000,
  "paid_amount": 50000,
  "bank_code": "PERMATA",
  "paid_at": "2016-10-12T08:15:03.404Z",
  "payer_email": "wildan@xendit.co",
  "description": "This is a description",
  "adjusted_received_amount": 47500,
  "fees_paid_amount": 0,
  "updated": "2016-10-10T08:15:03.404Z",
  "created": "2016-10-10T08:15:03.404Z",
  "currency": "IDR",
  "payment_channel": "PERMATA",
  "payment_destination": "888888888888"
```



## LAMPIRAN II

### DOKUMEN HASIL PENGUJIAN

#### DOKUMEN KUISIONER PENGUJIAN

##### KUISIONER PENGUJIAN BACK-END E-COURSE DI PT.OTAK KANAN BERBASIS LARAVEL

Sifat : Tertutup  
 Tujuan : Pengujian REST-API Back-End E-Course Berbasis Laravel  
 Menggunakan Postman.  
 Tanggal Pengisian : 22 Juli 2023

#### A. Identitas Responden

1. Nama : ZABINA ANASTASYA NURHALIZA
2. Jabatan : Back-End Developer E-Course

#### B. Petunjuk Pengisian

Berdasarkan pemeriksaan REST-API dari perangkat lunak yang dilakukan Bapak/Ibu, berilah tanda centang (✓) pada alternatif jawaban yang tersedia.

Keterangan:

1. Sesuai: Beri tanda centang (✓) pada kolom jika REST-API berjalan sesuai skenario.
2. Tidak Sesuai: Beri tanda centang (✓) pada kolom jika REST-API tidak berjalan sesuai skenario.

#### 1. Pengujian Endpoint Login “/auth/login”

| No. | Skenario Pengujian                                            | Hasil yang diharapkan                                                                                             | Hasil  |              |
|-----|---------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|--------|--------------|
|     |                                                               |                                                                                                                   | Sesuai | Tidak Sesuai |
| 1.  | Pengujian endpoint “/auth/login” dengan data yang valid       | Berhasil mengembalikan data <i>user</i> yang <i>login</i> beserta bearer token                                    | ✓      |              |
| 2.  | Pengujian endpoint “/auth/login” dengan data yang tidak valid | Gagal mengembalikan data <i>user login</i> dan mengembalikan pesan “ <i>The email or password is incorrect.</i> ” | ✓      |              |

|    |                                                                                     |                                                                                                                             |   |  |
|----|-------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|---|--|
| 3. | Pengujian endpoint <i>'/auth/login'</i> dengan <i>email</i> yang bukan <i>email</i> | Gagal mengembalikan data <i>user login</i> dan mengembalikan pesan <i>"The email field must be a valid email address.."</i> | ✓ |  |
|----|-------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|---|--|

2. Pengujian Endpoint Registrasi *"/auth/register-user"*

| No. | Skenario Pengujian                                                                          | Hasil yang diharapkan                                                                                                                             | Hasil  |              |
|-----|---------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|--------|--------------|
|     |                                                                                             |                                                                                                                                                   | Sesuai | Tidak Sesuai |
| 1.  | Pengujian endpoint <i>'/auth/register-user'</i> dengan data yang valid                      | Berhasil mengembalikan data <i>user</i> yang melakukan registrasi                                                                                 | ✓      |              |
| 2.  | Pengujian endpoint <i>'/auth/register-user'</i> dengan email yang tidak valid               | Gagal mengembalikan data <i>user</i> yang melakukan registrasi dan mengembalikan pesan <i>"The email field must be a valid email address"</i>     | ✓      |              |
| 3.  | Pengujian endpoint <i>'/auth/register-user'</i> dengan email yang telah digunakan           | Gagal mengembalikan data <i>user</i> yang melakukan registrasi dan mengembalikan pesan <i>"The email has already been taken."</i>                 | ✓      |              |
| 4.  | Pengujian endpoint <i>'/auth/register-user'</i> dengan password yang kurang dari 8 karakter | Gagal mengembalikan data <i>user</i> yang melakukan registrasi dan mengembalikan pesan <i>"The password field must be at least 8 characters."</i> | ✓      |              |

3. Pengujian *Endpoint* Data Pelatihan

| No. | Skenario Pengujian                                                                                                                              | Hasil yang diharapkan                                                                      | Hasil  |              |
|-----|-------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|--------|--------------|
|     |                                                                                                                                                 |                                                                                            | Sesuai | Tidak Sesuai |
| 1.  | Pengujian endpoint <code>*/course/popular</code> untuk mengambil data pelatihan terpopuler                                                      | Mengembalikan data pelatihan terpopuler beserta relasi dengan materi, user dan juga review | ✓      |              |
| 2.  | Pengujian endpoint <code>*/course/popular/2</code> untuk mengambil data pelatihan terpopuler berdasarkan kategori                               | Mengembalikan data pelatihan terpopuler beserta relasi dengan materi, user dan juga review | ✓      |              |
| 3.  | Pengujian endpoint <code>*/course/detail/{slug}</code> untuk mengambil data detail pelatihan berdasarkan slug menggunakan slug yang valid       | Mengembalikan data detail pelatihan beserta relasi dengan materi, user dan juga review     | ✓      |              |
| 4.  | Pengujian endpoint <code>*/course/detail/{slug}</code> untuk mengambil data detail pelatihan berdasarkan slug menggunakan slug yang tidak valid | Gagal mengembalikan data pelatihan dan mengembalikan pesan "Course not found"              | ✓      |              |

4. Pengujian *Endpoint* Pembelian Pelatihan “/order/create-payment”

| No. | Skenario Pengujian                                                                                                 | Hasil yang diharapkan                                                                                                                                         | Hasil  |              |
|-----|--------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|--------------|
|     |                                                                                                                    |                                                                                                                                                               | Sesuai | Tidak Sesuai |
| 1.  | Pengujian endpoint <i>*/order/create-payment*</i> untuk melakukan pembelian pelatihan                              | Berhasil melakukan transaksi untuk mendapatkan <i>link</i> pembayaran. Dan mengembalikan data dari 3 tabel, <i>order</i> , <i>invoice</i> dan <i>payments</i> | ✓      |              |
| 2.  | Pengujian endpoint <i>*/order/create-payment*</i> untuk melakukan pembelian pelatihan dengan data yang kosong      | Gagal melakukan transaksi pembelian pelatihan dan mengembalikan pesan “ <i>The course_id field is required.</i> ”                                             | ✓      |              |
| 3.  | Pengujian endpoint <i>*/order/create-payment*</i> untuk melakukan pembelian pelatihan dengan data yang tidak valid | Gagal melakukan transaksi pembelian pelatihan dan mengembalikan pesan “ <i>Course not found</i> ”                                                             | ✓      |              |
| 4.  | Pengujian endpoint <i>*/order/create-payment*</i> untuk melakukan pembelian sebelum melakukan <i>login</i>         | Gagal melakukan transaksi pembelian pelatihan dan mengembalikan pesan “ <i>Authorization Token not found</i> ”                                                | ✓      |              |
| 5.  | Pengujian endpoint <i>*/order/create-payment*</i> untuk melakukan pembelian                                        | Gagal melakukan transaksi pembelian pelatihan dan                                                                                                             | ✓      |              |

|  |                                              |                                                     |  |  |
|--|----------------------------------------------|-----------------------------------------------------|--|--|
|  | namun terdapat kegagalan di tengah transaksi | mengembalikan pesan " <i>Error creating order</i> " |  |  |
|--|----------------------------------------------|-----------------------------------------------------|--|--|

5. Pengujian *Endpoint* Callback *"/callback/"*

| No. | Skenario Pengujian                                                               | Hasil yang diharapkan                                                                                      | Hasil  |              |
|-----|----------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|--------|--------------|
|     |                                                                                  |                                                                                                            | Sesuai | Tidak Sesuai |
| 1.  | Pengujian endpoint <i>"/callback/"</i> dengan data <i>external_id</i> yang valid | Berhasil merubah data status pembayaran, dan berhasil membuat data pada table <i>course_active</i>         | ✓      |              |
| 2.  | Pengujian endpoint <i>"/callback/"</i> dengan data <i>external_id</i> yang valid | Gagal melakukan perubahan data status pembayaran, dan mengembalikan pesan " <i>external_id not found</i> " | ✓      |              |

Saran:

Surabaya, 22 Juli 2023


( ZABINA ANASTASYA  
NURHALIZA )