

**PENGEMBANGAN *MASSIVE OPEN ONLINE COURSE*
(MOOC) MODUL *ENROLLMENT* dan *PAYMENT***



Disusun Oleh:

N a m a : Hendry Aryadi

NIM : 14523238

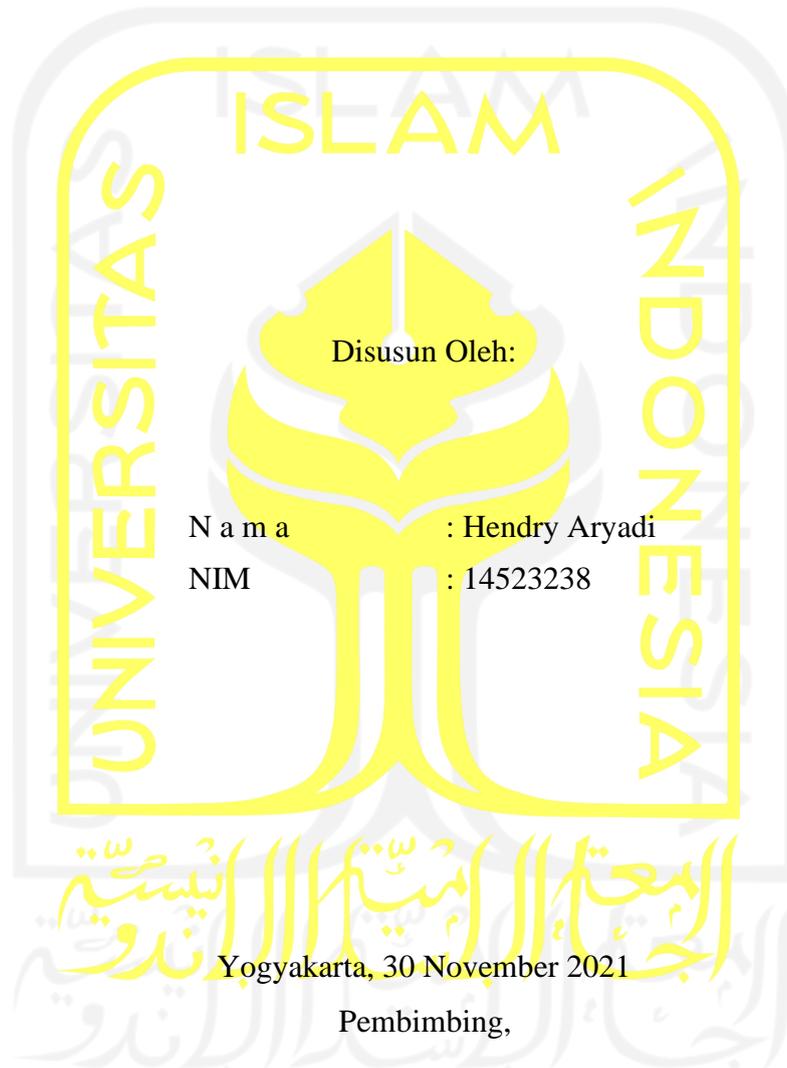
**PROGRAM STUDI INFORMATIKA - PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2021

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN *MASSIVE OPEN ONLINE COURSE*
(MOOC) MODUL *ENROLLMENT* dan *PAYMENT***

TUGAS AKHIR




(Kholid Haryono S.T., M.Kom.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGEMBANGAN *MASSIVE OPEN ONLINE COURSE*
(MOOC) MODUL *ENROLLMENT* dan *PAYMENT***

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk
memperoleh gelar Sarjana Komputer dari Program Studi Informatika
di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 30 November 2021

Tim Penguji

Kholid Haryono, S.T., M.Kom.

Anggota 1

Hari Setiaji, S.Kom., M.Eng.

Anggota 2

Chanifah Indah Ratnasari, S.Kom., M.Kom.

Mengetahui,

Ketua Program Studi Informatika - Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Hendry Aryadi

NIM : 14523238

Tugas akhir dengan judul:

**PENGEMBANGAN *MASSIVE OPEN ONLINE COURSE*
(MOOC) MODUL *ENROLLMENT* dan *PAYMENT***

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 30 November 2021



(Hendry Aryadi)

HALAMAN PERSEMBAHAN

Dengan rasa syukur kepada Allah subhanahu wata'ala, penulis mempersembahkan tugas akhir yang berjudul “Pengembangan *Massive Open Online Course* Modul *Enrollment* dan *Payment*”. Selama penyusunan dan pengerjaan tugas akhir, penulis mendapatkan banyak bantuan, bimbingan, serta dukungan dari berbagai pihak. sehingga pada kesempatan ini, penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Kedua orang tua, yang telah memberikan kasih sayang, motivasi dan do'a disetiap hari-hari yang saya lalui.
2. Bapak Kholid Haryono, S.T., M.Kom., selaku pembimbing yang telah meluangkan waktu, tenaga, dan selalu sabar dalam membimbing saya selama mengerjakan Tugas Akhir ini.
3. Reza Martha Bella Estanto Putri, S.E., yang selalu mendampingi, memberi semangat dan memberikan do'a kepada saya selama mengerjakan tugas akhir.
4. Teman-teman yang selalu berbagi suka maupun duka dengan sifat-sifat unik kalian dan semoga kita dapat tetap menjaga tali silaturahmi selamanya.

HALAMAN MOTO

“Maka sesungguhnya bersama kesulitan ada kemudahan. Sesungguhnya bersama kesulitan itu ada kemudahan.” (Al Qur’an: Surat Al Insyirah : 5 – 6)

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”

(Al Qur’an: Surat Al Baqarah : 286)



KATA PENGANTAR



Assalamu'alaikum Warahamatullahi Wabarakatuh

Alhamdulillah, penulis panjatkan kehadiran Allah Subhanahuwa ta'ala yang telah memberikan rahmat, hidayah, serta karunia-Nya dan tak lupa juga shalawat, serta salam kami haturkan kepada junjungan kita Nabi Muhammad SAW, sehingga laporan Tugas Akhir ini dapat terselesaikan. Tugas Akhir ini dibuat sebagai salah satu syarat yang harus dipenuhi untuk memperoleh gelar sarjana di Jurusan Informatika Universitas Islam Indonesia. Adapun Tugas Akhir yang penulis lakukan yaitu mengenai Pengembangan MOOC Modul *Enrollment* dan *Payment*.

Pelaksanaan Tugas Akhir ini merupakan salah satu mata kuliah wajib dari jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia. Tugas Akhir ini juga merupakan sarana bagi penulis untuk menambah wawasan, serta pengalaman dalam menerapkan keilmuan yang telah didapatkan selama belajar di bangku perkuliahan.

Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada:

1. Allah SWT, atas rahmat, taufik dan hidayah-Nya.
2. Nabi Muhammad SAW, sebagai suri tauladan bagi seluruh umat manusia dalam menghadapi dan menjalani kehidupan.
3. Orang Tua dan keluarga penulis atas segala do'a dan dukungannya.
4. Hendrik, S.T.,M.Eng., selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
6. Kholid Haryono S.T., M.Kom., selaku Dosen Pembimbing Tugas Akhir di Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
7. Sahabat dan teman-teman yang telah memberikan semangat, do'a dan membantu mengumpulkan data-data yang diperlukan dalam menyelesaikan Tugas Akhir.
8. Semua pihak yang telah banyak membantu dalam pengerjaan Tugas Akhir yang tidak dapat sebutkan satu per satu.

Penulis menyadari bahwa laporan ini masih belum sempurna, karena keterbatasan kemampuan dan pengalaman di lapangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi kesempurnaan laporan Tugas Akhir ini. Akhir kata, penulis berharap laporan ini dapat bermanfaat bagi semua pihak.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Yogyakarta, 30 November 2021

(Hendry Aryadi)



SARI

MOOC (*Masive Open Online Course*) adalah sebuah sistem pembelajaran berupa kursus online dalam skala yang besar di mana melalui sistem MOOC semua orang dapat kuliah gratis tanpa terikat jadwal dan bebas memilih mata kuliah yang diinginkan. MOOC juga berkembang dengan cepat seperti di Indonesia sendiri muncul beberapa MOOC seperti IndonesiAx, Ruangguru, CodeSaya, Fisipol UGM, MOOCs, Sibejo, dan lain sebagainya.

Pada MOOC sendiri khususnya pada modul *Enrollment* dan *Payment* merupakan tahap awal dari serangkaian proses yang ada, akan tetapi pada proses *enrollment* dan *payment* ditemukan masalah dimana ada calon peserta merasa kesulitan dikarenakan pada proses *enrollment* dan *payment* masih kurang efisien. Oleh karena itu pada penelitian Tugas Akhir ini dilakukan pengembangan MOOC Modul *Enrollment* dan *Payment* untuk menambah minat dan mempermudah akses pembayaran.

Pengembangan MOOC modul *enrollment* dan *payment* ini dilakukan dengan menggunakan metode purwarupa (*prototyping*) menurut Pressman. Pengembangan MOOC modul *enrollment* dan *payment* ini berbasis pada *platform website* dengan menggunakan Laravel sebagai *framework* PHP.

Kata Kunci: MOOC, *enrollment*, *payment*, *website*, *platform*, *prototyping*, *framework*.

GLOSARIUM

<i>Framework</i>	Sebuah kerangka kerja yang digunakan untuk mempermudah dalam membuat dan mengembangkan sistem.
Laravel	Sebuah <i>framework</i> PHP yang digunakan untuk membuat dan mengembangkan website.
Purwarupa	Contoh final produk.
<i>Prototyping</i>	Sebuah metode dalam mengembangkan perangkat lunak.



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO.....	vi
KATA PENGANTAR.....	vii
SARI.....	ix
GLOSARIUM	x
DAFTAR ISI	xi
DAFTAR TABEL	xiii
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metode Penelitian	4
1.7 Sistematika Penulisan	5
BAB II LANDASAN TEORI.....	7
2.1 Teori Dasar.....	7
2.1.1 <i>Massive Open Online Course (MOOC)</i>	7
2.1.2 <i>Enrollment</i>	7
2.1.3 <i>Payment</i>	7
2.2 <i>Prototyping</i>	8
2.2.1 Kelebihan Metode <i>Prototyping</i>	9
2.2.2 Kekurangan Metode <i>Prototyping</i>	9
2.3 <i>Use Case Diagram</i>	9
2.4 <i>Activity Diagram</i>	10
2.5 Pengujian <i>User Acceptance Test (UAT)</i>	11
2.6 Studi Awal.....	11
BAB III METODE PENELITIAN.....	16
3.1 Pengumpulan Data (komunikasi).....	16
3.1.1 Wawancara	16
3.1.2 Diskusi.....	16
3.2 Identifikasi Masalah	16
3.3 Perencanaan Secara Cepat.....	17
3.3.1 Analisis Kebutuhan Sistem	17
3.3.2 Analisis Kebutuhan Proses, Input, dan Output.....	17
3.4 Perancangan dan Desain <i>Prototype</i>	18
3.4.1 Perancangan <i>Use Case Diagram</i>	18
3.4.2 Perancangan <i>Activity Diagram</i>	21
3.5 Perancangan Basisdata	28
3.5.1 Relasi Tabel	29
3.5.2 Struktur Tabel Basisdata	29
3.6 Perancangan Antarmuka	36

	xii
3.6.1 Skenario <i>User Student</i>	37
3.6.2 Skenario <i>User Institusi</i>	44
3.6.3 Perbaikan Skenario <i>User Institusi</i>	51
BAB IV HASIL DAN PEMBAHASAN.....	55
4.1 Implementasi	55
4.1.1 Skenario <i>User Student</i>	55
4.1.2 Skenario <i>user institusi</i>	59
4.2 Pembahasan.....	62
4.2.1 Pengujian	62
4.2.2 Komparasi sistem	68
BAB V KESIMPULAN DAN SARAN	71
5.1 Kesimpulan	71
5.2 Saran.....	71
DAFTAR PUSTAKA	72
LAMPIRAN	74



DAFTAR TABEL

Tabel 3. 1 Analisis kebutuhan proses, input, dan output	17
Tabel 3. 2 Penjelasan <i>Use Case</i> Modul <i>Enrollment</i>	19
Tabel 3. 3 Penjelasan <i>usecase</i> modul <i>payment</i>	20
Tabel 3. 4 Penjelasan alur registrasi akun.....	21
Tabel 3. 5 Penjelasan alur kegiatan edit akun.....	22
Tabel 3. 6 Penjelasan alur kegiatan hapus akun	23
Tabel 3. 7 Penjelasan alur lihat akun	23
Tabel 3. 8 Penjelasan alur lihat data <i>enrollment</i>	24
Tabel 3. 9 Penjelasan alur tambah informasi pembayaran.....	25
Tabel 3. 10 Penjelasan alur edit informasi pembayaran	25
Tabel 3. 11 Penjelasan alur lihat informasi pembayaran	26
Tabel 3. 12 Penjelasan alur lihat data pembayaran.....	27
Tabel 3. 13 Penjelasan alur lihat riwayat pembayaran	28
Tabel 3. 14 Tabel <i>users</i>	29
Tabel 3. 15 Tabel <i>course</i>	30
Tabel 3. 16 Tabel <i>module</i>	31
Tabel 3. 17 Tabel <i>transaction</i>	31
Tabel 3. 18 Tabel <i>transaction course</i>	32
Tabel 3. 19 Tabel <i>setting</i>	32
Tabel 3. 20 Tabel <i>transaction midtrans</i>	33
Tabel 3. 21 Tabel <i>failed jobs</i>	34
Tabel 3. 22 Tabel <i>password reset</i>	34
Tabel 3. 23 Tabel <i>ratings</i>	35
Tabel 3. 24 Tabel <i>enquiries</i>	35
Tabel 3. 25 Tabel <i>session</i>	36
Tabel 3. 26 Tabel <i>personal akses token</i>	36
Tabel 3. 27 Hasil pengujian <i>user (Student)</i>	44
Tabel 3. 28 Hasil pengujian <i>user (institusi)</i>	50
Tabel 3. 29 Hasil pengujian perbaikan <i>user (institusi)</i>	54
Tabel 4. 1 pengujian <i>alpha</i>	63
Tabel 4. 2 Pengujian <i>beta</i>	63
Tabel 4. 3 Pengujian UAT <i>Student</i>	65

Tabel 4. 4 Pengujian UAT Institusi	66
Tabel 4. 5 Hasil pengujian UAT <i>student</i>	67
Tabel 4. 6 Hasil pengujian UAT institusi	68
Tabel 4. 7 Tabel Komparasi.....	69



DAFTAR GAMBAR

Gambar 2. 1 Alur Proses Metode Prototyping	8
Gambar 3. 1 <i>Use Case Diagram</i> Modul <i>Enrollment</i>	19
Gambar 3. 2 <i>Use Case Diagram</i> Modul <i>Payment</i>	20
Gambar 3. 3 <i>Activity Diagram</i> Registrasi Akun	21
Gambar 3. 4 <i>Activity Diagram</i> edit akun	22
Gambar 3. 5 <i>Activity diagram</i> hapus akun	22
Gambar 3. 6 <i>Activity diagram</i> lihat akun	23
Gambar 3. 7 <i>Activity diagram</i> lihat data <i>enrollment</i>	24
Gambar 3. 8 <i>Activity diagram</i> tambah informasi pembayaran	24
Gambar 3. 9 <i>Activity diagram</i> edit informasi pembayaran	25
Gambar 3. 10 <i>Activity diagram</i> lihat informasi pembayaran	26
Gambar 3. 11 <i>Activity diagram</i> lihat data pembayaran	27
Gambar 3. 12 <i>Activity diagram</i> lihat riwayat pembayaran	28
Gambar 3. 13 Relasi basisdata mooc	29
Gambar 3. 14 Halaman awal sistem	37
Gambar 3. 15 Halaman login	38
Gambar 3. 16 Halaman login google	38
Gambar 3. 17 Halaman login email	39
Gambar 3. 18 Halaman ganti <i>password</i>	39
Gambar 3. 19 Halaman ganti <i>password</i> email	40
Gambar 3. 20 Halaman <i>dashboard</i> student	40
Gambar 3. 21 Halaman riwayat pembayaran	41
Gambar 3. 22 Halaman kursus saya	41
Gambar 3. 23 Halaman detail kursus	42
Gambar 3. 24 Halaman cari kursus	42
Gambar 3. 25 Halaman metode pembayaran	42
Gambar 3. 26 Halaman kursus yang sudah dibeli	43
Gambar 3. 27 Halaman profil student	43
Gambar 3. 28 Halaman edit profile	44
Gambar 3. 29 Halaman <i>dashboard</i> institusi	45
Gambar 3. 30 Halaman <i>course</i> (institusi)	46
Gambar 3. 31 Halaman create <i>course</i>	46

Gambar 3. 32 Halaman edit <i>course</i>	47
Gambar 3. 33 Halaman detail <i>course</i>	47
Gambar 3. 34 Halaman detail student.....	48
Gambar 3. 35 Halaman student.....	48
Gambar 3. 36 Halaman <i>enrollment</i> (institusi).....	49
Gambar 3. 37 Halaman list <i>enrollment course</i>	49
Gambar 3. 38 Halaman <i>transaction</i> (institusi).....	50
Gambar 3. 39 Halaman <i>dashboard</i> institusi	51
Gambar 3. 40 Halaman list <i>course</i>	52
Gambar 3. 41 Halaman create <i>course</i>	52
Gambar 3. 42 Halaman edit <i>course</i>	53
Gambar 3. 43 Halaman detail <i>course</i>	53
Gambar 4. 1 Halaman awal sistem	55
Gambar 4. 2 student dialihkan untuk login.....	56
Gambar 4. 3 Student mengisi form reset <i>password</i>	56
Gambar 4. 4 Sistem menampilkan halaman daftar email	57
Gambar 4. 5 Halaman pilihan akun google	57
Gambar 4. 6 Halaman detail kursus berbayar	58
Gambar 4. 7 Halaman pembayaran kursus	58
Gambar 4. 8 Halaman <i>course</i> berbayar setelah dilakukan pembayaran	59
Gambar 4. 9 Halaman payment history	59
Gambar 4. 10 Halaman awal sistem (institusi)	60
Gambar 4. 11 Institusi melihata data student.....	60
Gambar 4. 12 Tombol hapus akun student	61
Gambar 4. 13 Insitusi mengelola <i>course</i>	61
Gambar 4. 14 Tombol lihat, edit, dan hapus kursus	61
Gambar 4. 15 Menu <i>enrollment</i> (institusi).....	62
Gambar 4. 16 Halaman cek transaksi <i>student</i>	62

BAB I PENDAHULUAN

1.1 Latar Belakang

MOOC (*Massive Open Online Course*) merupakan sebuah sistem pembelajaran berupa kursus online skala besar (*massive*), akan tetapi MOOC sedikit berbeda dengan kursus *online* pada umumnya, di mana MOOC lebih spesifik seperti dalam pengajarannya harus dilengkapi dengan manajemen pengetahuan, pembelajaran informal, pembelajaran sosial, dan prinsip-prinsip pembelajaran seumur hidup (Gené, Núñez, & Blanco, 2014). Kehadiran MOOC sendiri tentu menghadirkan sebuah tantangan, dimana pada kenyataannya dengan kemudahan yang diberikan MOOC ada beberapa pelajar ditemukan mengirimkan tugas dengan jawaban yang sama selain itu juga ada ditemukan pelajar belum siap dengan materi kelas perkuliahan (Ng et al., 2012).

MOOC pada dasarnya memiliki 6 modul seperti *enrollment*, *payment*, *course*, *grading*, *certificate*, dan *ceremony*. Modul awal yang dimiliki oleh MOOC yaitu modul *Enrollment* dan *Payment*, bahkan bagian ini cukup krusial dimana pada proses ini merupakan awal dan penentu bahwa peserta akan jadi mendaftar atau tidak. Hal ini dikarenakan saat pelajar mendaftar pada MOOC tidak secara otomatis dapat mengakses seluruh kursus yang tersedia, melainkan pelajar harus mendaftar dalam kursus tersebut satu persatu. Selanjutnya pada proses *payment*, pelajar akan diminta untuk menyelesaikan pembayaran dikarenakan ada beberapa kursus yang berbayar. Jika pada salah satu proses di atas ada yang tidak terpenuhi maka pendaftaran peserta pada kursus tersebut akan dibatalkan. Beberapa fakta yang ada yaitu kemudahan dalam melakukan pembayaran (*payment*) mendorong masyarakat untuk meningkatkan konsumsinya (Sitorus, 2006), dalam kasus ini berarti peningkatan intensitas *enrollment*.

Model *enrollment* yang umum dipakai ada 3 macam, yaitu manual *enrollment*, *guest access*, dan *self enrollment*. Dari ketiga model tersebut, masing-masing memiliki kelebihan dan kekurangan, seperti pada model *manual enrollment* yang mengharuskan admin untuk mendaftarkan semua pelajar (pendaftar) ke dalam sistem, lalu pada model *guest access*, admin akan memberikan identitas sementara tanpa proses mendaftar sehingga pelajar dapat masuk ke dalam sistem, akan tetapi fitur yang didapatkan hanya sedikit dan yang terakhir yaitu *self enrollment*, pada model ini admin akan memberikan sebuah form yang berisi data diri untuk

diisi oleh pelajar, sehingga jika pelajar ingin masuk ke sistem, maka diharuskan melakukan pengisian data terlebih dahulu (review.com, 2018). Pada model *self enrollment* dapat juga mendaftar menggunakan pihak ketiga, akan tetapi timbul juga tantangan seperti kecenderungan pengunjung untuk login dengan menggunakan akun tertentu, sesuai survey yang dilakukan oleh Blue Research terhadap lebih dari 600 pengguna internet pada tahun 2011 mengungkapkan bahwa 66% orang menginginkan adanya login sosial pada suatu situs, sedangkan yg 34% tidak. Pengguna juga lebih suka menggunakan login sosial (41%) daripada membuat akun baru pada suatu situs (24%) atau menggunakan akun tamu (35%) (Olson, 2011).

Selanjutnya pada proses *payment*, secara umum ada 3 metode pembayaran, yaitu dengan menggunakan web *merchant*, penyedia jasa *payment gateway*, dan campuran. Penerapan metode pembayaran dengan menggunakan web *merchant* memiliki beberapa resiko, seperti *merchant* yang terbatas, pencurian data pelanggan, manipulasi transaksi bahkan dapat sampai peretasan sistem. Akan tetapi penerapan metode pembayaran menggunakan *payment gateway* juga sedikit rumit, khususnya pada saat implementasi, di mana salah satu syaratnya adalah *hosting* yang digunakan diharuskan cukup tangguh sehingga tidak *down* saat ramai diakses. Selain itu ada juga masalah, dimana jenis pembayaran yang disediakan *payment gateway* dan *e-wallet* dari web *merchant* masih menyulitkan pengguna, seperti kepemilikan paypal, visa, *credit card*, dan penggunaan *e-wallet* yang belum merata, terlebih lagi pada daerah yang akses internetnya masih sulit dan pengguna yang gagap teknologi (Hatane Samuel, 2006), Mengingat alat pembayaran seperti visa, paypal, dan *credit card* sendiri adalah salah satu alat pembayaran yang tingkat ketersediaan dan penggunaan paling luas di semua negara. Sedangkan pada metode pembayaran campuran, pengelolaan pembayarannya menggunakan web *merchant*, akan tetapi dibantu juga dengan *payment gateway*, sehingga dapat meminimalisir resiko yang timbul dari masing – masing metode.

Penelitian ini bertujuan untuk menawarkan model *enrollment* dan *payment* yang sesuai dari berbagai MOOC yang sudah ada, menjadi sebuah prototype. Dalam mengembangkan MOOC pada modul *enrollment* dan *payment* untuk menambah minat peserta dan mempermudah akses pembayaran, penulis akan menggunakan metode *prototyping* menurut Roger S Pressman (Pressman, 2010) sebagai acuan penelitian.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka peneliti merumuskan masalah sebagai berikut, bagaimana mengembangkan MOOC pada modul *Enrollment* dan *payment* untuk menambah minat peserta dan mempermudah akses pembayaran?

1.3 Batasan Masalah

Adapun beberapa batasan masalah yang penulis cantumkan dalam penelitian tugas akhir ini adalah sebagai berikut:

- a. Pengembangan MOOC modul *enrollment* dan *payment* yang dibuat hanya pada *platform* Website.
- b. Pembayaran yang dibuat merupakan contoh dan bukan pembayaran sebenarnya.
- c. Metode pendaftaran dan pembayaran yang dipilih untuk digunakan bersumber dari 6 *platform* MOOC acuan, yaitu *Coursera*, *edX*, *FutureLearn*, *Udacity*, *Udemy*, dan *IndonesiaX*.
- d. Institusi merangkap tugas tutor sebagai validator kursus sebelum *dipost* pada website

1.4 Tujuan Penelitian

Tujuan penelitian pada tugas akhir ini adalah membuat sebuah *prototype* MOOC modul *enrollment* dan *payment* yang dapat menambah minat peserta dan mempermudah akses pembayaran.

1.5 Manfaat Penelitian

Manfaat dari penelitian tugas akhir ini adalah sebagai berikut:

- a. Mendapatkan pengetahuan dalam membuat MOOC pada modul *enrollment* dan *payment*.
- b. Mendapatkan pengetahuan mengenai proses bisnis MOOC, khususnya pada modul *enrollment* dan *payment*.
- c. Meningkatkan kualitas MOOC modul *enrollment* dan *payment*.
- d. Membuat MOOC modul *enrollment* dan *payment*.

1.6 Metode Penelitian

Metode penelitian yang akan dilakukan dalam penelitian tugas akhir ini adalah dengan menggunakan metode pengembangan perangkat lunak berbasis purwarupa (*prototyping*) menurut (Pressman, 2010). Metode ini dipilih dikarenakan metode ini cocok untuk mengembangkan sebuah perangkat yang akan dikembangkan kembali. Berdasarkan metode *prototyping* (Pressman, 2010) sendiri terdiri dari 5 tahap yaitu, Pengumpulan data (komunikasi), Perencanaan secara cepat (analisis kebutuhan), Pemodelan dan perancangan, Pembentukan *prototype*, Penyerahan sistem/perangkat lunak ke pengguna (evaluasi dan pengujian).

1) Pengumpulan Data (komunikasi).

Pada tahap ini penulis melakukan pengumpulan data yang dibutuhkan dalam penelitian. Data tersebut bersumber dari hasil riset sistem yang diambil dari internet, kemudian dilanjutkan dengan menggunakan kuesioner dan wawancara kepada pihak-pihak terkait.

2) Perencanaan secara cepat (analisis Kebutuhan).

Hasil dari tahapan pengumpulan data digunakan sebagai masukan dalam melaksanakan tahapan perencanaan. Pada tahap ini penulis melakukan analisis kebutuhan sistem, yang terdiri dari analisis kebutuhan proses dan analisis *input-output*. Sehingga didapatkan berbagai kebutuhan yang diperlukan dalam pemodelan dan merancang sistem yang akan dibuat.

3) Pemodelan dan Perancangan.

Setelah analisis kebutuhan selesai, penulis melanjutkan ke tahap pemodelan dan perancangan. Pada tahap ini akan dibuat desain antarmuka. Pada pembuatan desain antarmuka, penulis menggunakan *tools* Mockplus dan JustInMind. Kemudian hasil dari proses pemodelan ini akan digunakan untuk membuat *prototype* sistem MOOC.

4) Pembentukan *Prototype*.

Pada tahap ini penulis membuat sebuah iterasi yang berisi runtutan pengerjaan *prototype* sistem MOOC *enrollment* dan *payment* dari awal hingga akhir dan setelah selesai, penulis akan melakukan testing kepada pengguna akhir.

5) Penyerahan sistem/perangkat lunak ke pengguna (evaluasi dan Pengujian)

Prototype yang sudah jadi tersebut akan diuji oleh penulis terlebih dahulu, jika sudah sesuai maka penulis akan melanjutkan pengujian ke pengguna akhir seperti mahasiswa dan tenaga pendidik. Pada pengujian ini, pengguna akhir akan diminta memberikan tanggapan, apakah sistem yang dibuat sudah sesuai atau belum. Jika belum sesuai, maka penulis akan melakukan perbaikan dan melakukan pengujian kembali hingga sistem yang dibuat sesuai.

1.7 Sistematika Penulisan

Sistematika penulisan laporan bertujuan untuk memudahkan dalam memahami isi yang ada ada laporan tugas akhir ini. Pada laporan tugas akhir ini dibuat dengan sistematika sebagai berikut.

BAB I PENDAHULUAN

Berisi tentang pembahasan masalah umum yang meliputi latar belakang yang menyebabkan munculnya kebutuhan pengembangan MOOC modul *enrollment* dan *payment*, rumusan masalah, manfaat penelitian, dan sistematika penulisan selama kegiatan penelitian dilakukan.

BAB II LANDASAN TEORI

Berisi tentang uraian teori-teori yang digunakan sebagai dasar selama melakukan penelitian antara lain seperti pengertian *massive open online course*, pengertian *enrollment*, pengertian *payment* adapun pemodelan yang digunakan yaitu dengan menggunakan metode *prototyping*.

BAB III METODOLOGI

Berisi pembahasan seputar cara dan prosedur yang digunakan untuk menjawab masalah penelitian dengan cara analisis kebutuhan, perancangan sistem, perancangan basisdata, serta perancangan desain antarmuka.

BAB IV HASIL DAN PEMBAHASAN

Berisi tentang hasil dari proses pembuatan sistem yang telah dilakukan. Pembahasan yang dilakukan yaitu seperti pembahasan tentang implementasi sistem dan pengujian sistem.

BAB V KESIMPULAN

Berisi kesimpulan dari keseluruhan penelitian yang telah dilakukan, serta saran rekomendasi berdasarkan hasil penelitian.



BAB II

LANDASAN TEORI

2.1 Teori Dasar

Menjelaskan tentang teori – teori dasar yang berkaitan dengan pembuatan sistem *Massive Open Online Course*. Adapun teori yang akan dibahas adalah sebagai berikut.

2.1.1 *Massive Open Online Course (MOOC)*

Massive open online course adalah kursus terbuka secara online dan masal, yang terbuka untuk umum, dapat diikuti oleh siapa saja, kapan saja dan dimana saja, bahkan sebenarnya MOOC juga menyediakan layanan dalam hal konten, desain, poin akses, cara aplikasi, dan definisi keberhasilan (Emigawaty, 2017). Dalam perkembangannya metode pembelajaran jarak jauh mulai bermunculan, dimulai dari kelas kuliah online OCW (*Open Course Ware*) dan yg terbaru yaitu MOOC. Awal berkembangnya MOOC ini sejak tahun 2008 dan mulai populer di tahun 2012, berawal dari model pembelajaran jarak jauh dimana pusat pembelajaran mengirimkan materi melalui pos, hingga sekarang pembelajaran dilakukan secara online.

2.1.2 *Enrollment*

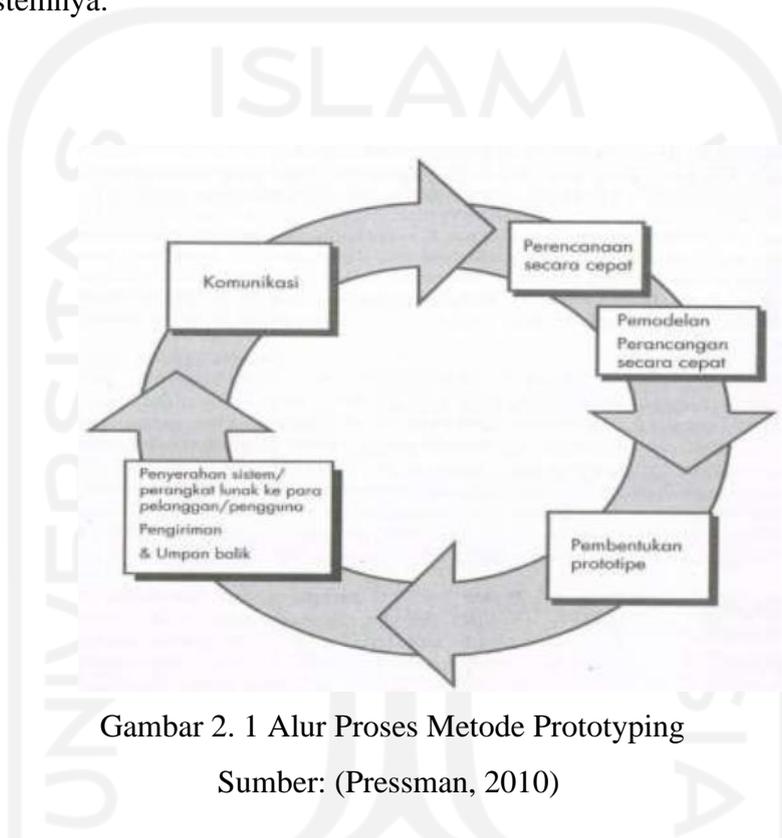
Menurut DEPDIKBUD *enrollment* atau pendaftaran adalah proses, cara, perbuatan mendaftar (mendaftarkan) seperti pencatatan nama, alamat, dan sebagainya dalam daftar. Sehingga pendaftaran/*enrollment* pada MOOC disini adalah sebuah proses pencatatan identitas seseorang untuk disimpan guna keperluan di masa mendatang. Penerapannya pada MOOC ada yang melakukan pencatatan mandiri dari data yang masuk dan ada juga yang menggunakan bantuan pihak ketiga dalam melakukan pendataan, sehingga pendaftar hanya perlu memberikan izin agar data identitas pada pihak ketiga dapat diakses pada MOOC yang bersangkutan. Adapun pihak ketiga yang banyak digunakan pada MOOC saat ini yaitu akun google, facebook, Microsoft, linkedin, dan appleid.

2.1.3 *Payment*

Menurut UU no.23 pasal 1(1999:6) pembayaran mencakup seperangkat aturan, lembaga dan mekanisme yang digunakan untuk melakukan pemindahan dana guna memenuhi suatu kewajiban yang timbul dari suatu kegiatan ekonomi. Ada pula yang menyatakan bahwa pembayaran adalah proses penukaran mata uang dengan barang, jasa atau informasi (Chan,

2009:108). Sehingga dapat disimpulkan bahwa pembayaran adalah kegiatan yang dilakukan dalam pemindahan mata uang menjadi barang, jasa ataupun informasi dari pembayar ke penerimanya, baik secara langsung maupun melalui media perantara lain. MOOC juga menerapkan *payment* pada sistemnya, ada yang mengelola secara mandiri dan ada juga yang menggunakan bantuan *payment gateway* dalam pengelolaan *payment* nya. Beberapa MOOC besar seperti *coursera*, *edx*, dan *futurelearn* bahkan mengombinasikan beberapa *payment gateway* pada sistemnya.

2.2 Prototyping



Gambar 2. 1 Alur Proses Metode Prototyping

Sumber: (Pressman, 2010)

Pada Gambar 2. 1 menjelaskan mengenai alur pengembangan sistem dengan menggunakan metode *Prototyping-Oriented Software*, prosesnya sendiri terdiri dari 5 proses yaitu, pengumpulan data (komunikasi), perencanaan secara cepat (analisis kebutuhan), pemodelan dan perancangan, pembentukan *prototype*, dan penyerahan sistem/perangkat lunak ke pengguna (evaluasi dan pengujian). Pada tahap pertama, dilakukan pengumpulan data dengan melakukan komunikasi antar tim pengembang dengan para *stakeholder*. Kemudian akan diidentifikasi spesifikasi kebutuhan apa saja yang diperlukan dalam mengembangkan sistem. Selanjutnya yaitu pemodelan rancangan, proses ini dilakukan dalam bentuk rancangan cepat, dimana rancangan ini berfokus pada representasi sistem yang akan dibuat (*user interface*). Setelah proses perancangan selesai, maka akan dilanjutkan pada pembuatan *prototype* sistem. Setelah *prototype* sistem selesai dibuat, selanjutnya akan diserahkan kepada

para *stakeholder* terkait, dimana mereka akan diminta untuk melakukan evaluasi dan memberikan umpan balik terhadap *prototype* tersebut, yang nantinya akan digunakan untuk memperbaiki sistem agar sesuai dengan kebutuhan. Proses evaluasi dan memberikan umpan balik ini dapat berlangsung lebih dari sekali, sampai kebutuhan yang diperlukan dalam pengembangan *prototype* terpenuhi.

2.2.1 Kelebihan Metode *Prototyping*

Kelebihan metode *prototyping* adalah sebagai berikut:

1. Setiap adanya perbaikan pada proses *prototype* merupakan hasil masukan dari pelanggan, sehingga lebih reliabel.
2. Memiliki komunikasi yang baik antara pengembang dengan pelanggan.
3. Pelanggan berperan aktif dalam pengembangan sistem.
4. Sistem yang dibangun memiliki kualitas yang diinginkan, karena sesuai dengan kebutuhan yang ada.

2.2.2 Kekurangan Metode *Prototyping*

Kekurangan metode *prototyping* adalah sebagai berikut:

1. Semua kebutuhan sistem harus diketahui terlebih dahulu.
2. Memiliki resiko yang tergolong tinggi untuk masalah – masalah yang tidak terstruktur.
3. Hubungan antara pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.

2.3 Use Case Diagram

Use Case Diagram adalah sebuah model UML yang digunakan untuk mendokumentasikan peran-peran dari *user* dan cara mereka berinteraksi dengan sistem yang akan dikembangkan (Satzinger, Jackson & Burd, 2012:78). Pada *Use Case*, kita mengidentifikasi suatu entitas yang disebut *actor*, yang berhubungan dengan sistem. Tujuan dari *use case* sendiri adalah untuk mengidentifikasi bagaimana sistem akan digunakan oleh *user*. Dalam *use case diagram* terdapat 4 komponen, antara lain adalah sebagai berikut:

1. *System Boundary*

System boundary adalah sebuah batas yang mengelilingi *Use Case*, *actor*, dan *communication*.

2. Actor

Actor adalah sebuah entitas baik itu manusia, atau mesin yang berinteraksi dengan sistem untuk mengerjakan tugas – tugas tertentu.

3. Use Case

Use Case adalah rangkaian proses yang menggambarkan perilaku interaksi antara *actor* dan sistem.

4. Communication

Communication adalah sebuah hubungan antara *use case* dengan *actor*.

2.4 Activity Diagram

Activity Diagram adalah suatu diagram yang digunakan untuk menggambarkan alur aktivitas pada tiap fungsionalitas sistem yang akan dikembangkan. Adapun komponen yang terdapat pada *activity diagram* adalah sebagai berikut:

1. Start Point

Start point adalah simbol yang berfungsi untuk menunjukkan titik awal dari dimulainya sebuah alur aktivitas.

2. End Point

End point adalah simbol yang berfungsi untuk menunjukkan titik akhir dari sebuah alur aktivitas.

3. Activities

Activities adalah simbol yang berfungsi untuk menunjukkan aktivitas yang dilakukan dalam sistem.

4. Fork

Fork adalah simbol yang berfungsi untuk menunjukkan sebuah percabangan aktivitas menjadi beberapa aktivitas.

5. Join

Join adalah simbol yang berfungsi untuk menunjukkan penggabungan beberapa aktivitas menjadi satu aktivitas, atau dengan kata lain kebalikan dari *fork*.

6. Decision

Decision adalah sebuah simbol yang berfungsi untuk menunjukkan dimana sebuah keputusan perlu dibuat pada diagram aktivitas.

7. *Swimlane*

Swimlane adalah sebuah kerangka yang berfungsi untuk mengelompokan aktivitas yang terjadi sesuai dengan organisasi yang bertanggung jawab atas serangkaian aktivitas tersebut.

2.5 Pengujian *User Acceptance Test (UAT)*

User Acceptance Test (UAT) adalah suatu proses pengujian yang dilakukan pada pengguna akhir dengan tujuan untuk mengetahui apakah sistem yang dibuat sudah sesuai dengan kebutuhan pengguna akhir atau belum (Aliya, 2021). Pada pelaksanaannya pengujian akan dilakukan bersamaan dengan pengujian *alfa* dan *beta* test, dimana hal ini dilakukan guna mencari error maupun permasalahan lain pada sistem sebelum dirilis ke pengguna akhir. Adapun tahapan dalam pengujian UAT adalah sebagai berikut:

1. *Planning*, penguji merencanakan jadwal pengujian.
2. *Design test case*, melakukan desain uji coba kasus yang akan diujikan pada pengguna akhir.
3. *Selection of testing team*, pemilihan penguji agar sesuai dengan sistem yang akan diuji.
4. *Executing test case documenting*, proses pengujian pada pengguna akhir dan membuat laporan dari hasil pengujian tersebut.
5. *Sign off*, tahap pengujian telah selesai dilakukan.

2.6 Studi Awal

Pada tahap ini penulis melakukan pengumpulan data awal dengan cara membandingkan *platform* yang ada di seluruh dunia dan mengambil 6 terbaik. 6 *platform* MOOC terbaik terdiri dari *Coursera*, edX, FutureLearn, Udacity, Udemy, dan IndonesiaX. *Platform* MOOC tersebut dipilih atas pertimbangan beberapa hal seperti jumlah peserta yg mendaftar, jumlah kelas/kursus yang ditawarkan dan jumlah mitra yang bekerja sama dengan *platform* MOOC tersebut. Dalam membantu menentukan *platform* mana saja yang akan menjadi acuan, penulis menggunakan data review yang sudah dilakukan oleh (mooclab.club, 2018) dan (review.com, 2018) sebagai tambahan referensi, kemudian penulis mendiskusikan kembali dan mendapatkan 6 *platform* terbaik. Pada (mooclab.club, 2018) hanya menyediakan data mentah dari 57 *platform* diseluruh dunia, yang terdiri dari 23 negara termasuk Indonesia. Sedangkan pada (review.com, 2018) menyediakan data yang berisi penilaian kualitas dari beberapa *platform*

terbaik di dunia. Pada aspek penilaiannya (review.com, 2018) menggunakan 5 kategori yang menjadi fitur penting atau disebut sebagai *Core Metric* pada MOOC. *Core metric* tersebut adalah sebagai berikut.

1) *Credentialing* (Kredensial)

Metric ini membandingkan nilai selain pengetahuan dan keterampilan yang akan didapatkan setelah menyelesaikan MOOC di salah satu *platform*.

2) *Course Diversity* (Keberagaman Kursus)

Metric ini menilai banyaknya jumlah kursus, keberagaman kursus, dan juga kemampuan sistem dalam menemukan kursus yang dapat membuat peserta tertarik untuk mendaftar.

3) *Course Features* (Fitur Kursus)

Metric ini digunakan untuk mengevaluasi fitur-fitur yang terdapat pada kursus tersebut.

4) *Social Features* (Fitur Sosial)

Metric ini digunakan untuk menilai kemampuan dalam berinteraksi dengan peserta lain yang menggunakan *platform* tersebut.

5) *Partner Institution* (Mitra Institusi)

Metric ini melakukan penilaian dari segi keberagaman dan kualitas dari institusi yang menjadi mitra *platform* tersebut.

Berikut merupakan tabel review *platform* MOOC yang dapat dilihat pada Tabel 2. 1 *Platform* MOOC

Tabel 2. 1 *Platform* MOOC

No	Nama <i>Platform</i>	Pengguna	Kelas	Mitra	Score Review (review.com)
1	<i>Coursera</i>	>= 35 juta	>= 2954	181 Univ dan Perusahaan	8.8
2	EdX	>= 14 juta	>=1900	130 Univ dan Perusahaan	7.4
3	FutureLearn	>= 7,1 juta	532	-	6.4
4	Udacity	>=53 ribu	200	-	0.4
5	Udemy	>=24 juta	>=80000	-	-
6	IndonesiaX	-	>25	-	-

Seperti terlihat pada Tabel 2. 1 *Platform* MOOC, dapat disimpulkan bahwa *Coursera* menempati posisi pertama sebagai *platform* terbaik dengan jumlah pengguna lebih dari 35 juta, lebih dari 2954 kelas dan 181 mitra universitas dan perusahaan (terbanyak diantara 6 *platform* lainnya). Selain itu, *coursera* juga mendapatkan nilai *review* terbaik, sementara *edX* berada pada posisi kedua, dimana perlu diketahui bahwa *Coursera* dan *edX* merupakan pelopor *platform* MOOC, sehingga cukup wajar jika *Coursera* dan *edX* menempati posisi teratas sebagai *platform* terbaik. Selanjutnya pada posisi ketiga ditempati oleh *FutureLearn*, *platform* ini merupakan MOOC berbasis Inggris, sehingga sebagian besar peserta berasal dari Eropa. Kemudian pada posisi ke empat ditempati oleh *Udacity*, sedikit berbeda dengan MOOC lainnya, *Udacity* lebih detail dan berfokus menghadirkan kursus yang bertema teknologi, sehingga kategori kursusnya menjadi kurang beragam (kurang lengkap). Sementara pada posisi kelima ditempati oleh *Udemy*, walaupun menempati posisi kelima, *Udemy* lebih unggul dibandingkan *platform* lainnya pada jumlah total kelas yang dibuka, yaitu lebih dari 80.000. Lalu yang terakhir yaitu *IndonesiaX*, sesuai Namanya, MOOC ini berasal dari Indonesia. *IndonesiaX* sendiri memiliki jumlah kursus yang tergolong sedikit, akan tetapi kualitas kursus yang ditawarkan tidak kalah berkualitas dengan MOOC lainnya.

Perlu diketahui, dalam *review* yang dilakukan (*review.com*, 2018) *Udemy* dan *IndonesiaX* tidak masuk dalam kategori *platform* terbaik, *Udemy* masuk dalam list dikarenakan pertimbangan dari jumlah kursus yang ditawarkan, sedangkan *IndonesiaX* masuk dalam list dikarenakan memiliki banyak mitra, tutor yang berkualitas. Sehingga penulis mendapatkan pengetahuan seputar MOOC di Indonesia. Setelah menganalisis *platform* tersebut secara umum, penulis kemudian fokus pada proses yang paling awal dilakukan yaitu *enrollment* dan *payment*. Untuk mempermudah dalam membandingkan modul *enrollment* dan *payment* dari masing – masing *platform* di atas maka dibuatlah tabel seperti terlihat pada Tabel 2. 2 *Komparasi Platform MOOC*.

Tabel 2. 2 *Komparasi Platform MOOC*

<i>Platform</i>	<i>Enrollment</i>	<i>Payment</i>
<i>Coursera</i>	<i>Coursera</i> menggunakan metode pendaftaran <i>self enrollment</i> menggunakan email. Adapun metode login menggunakan akun pihak ketiga, dapat dengan	<i>Coursera</i> menyediakan pembayaran dengan kartu kredit dan paypal. <i>Payment gateway</i> yang digunakan adalah

	menggunakan akun facebook, google, dan apple	authorize.net, razorpay, dan paypal. Pembayaran dilakukan satu persatu
edX	edX menggunakan metode pendaftaran <i>self enrollment</i> menggunakan email. adapun metode login menggunakan akun pihak ketiga, dapat dengan akun facebook, google, Microsoft, dan apple	edX menyediakan pembayaran dengan kartu kredit dan paypal. Payment gateway yang digunakan yaitu cybersourcevisa (bagian dari authorize.net) dan paypal. Pembayaran dilakukan satu persatu
FutureLearn	FutureLearn menggunakan metode pendaftaran <i>self enrollment</i> menggunakan email. Adapun metode login menggunakan akun pihak ketiga dapat dengan akun facebook dan google.	FutureLearn menyediakan pembayaran dengan kartu kredit dan paypal. Payment gateway yang digunakan yaitu stripe dan paypal. Pembayaran dilakukan satu persatu
Udacity	Udacity menggunakan metode pendaftaran <i>self enrollment</i> menggunakan email. Adapun metode login menggunakan akun pihak ketiga dapat dengan akun google ataupun facebook.	Udacity menyediakan metode pembayaran menggunakan kartu kredit, paypal, dan affirm.
Udemy	Udemy menggunakan metode pendaftaran <i>self enrollment</i> menggunakan email. Adapun metode login	Udemy menyediakan metode pembayaran menggunakan kartu kredit, paypal, ovo,

	menggunakan akun pihak ketiga dapat dengan akun facebook, google, dan apple.	alfamart, indomart, doku wallet, dan transfer ke bank.
IndonesiaX	IndonesiaX menggunakan metode pendaftaran <i>self enrollment</i> menggunakan email. Adapun metode login pihak ketiga dapat menggunakan facebook, google, dan linkedin.	IndonesiaX menyediakan metode pembayaran menggunakan Kartu kredit, Transfer bank, dan E-wallet gopay Payment gateway yang digunakan yaitu midtrans. Pembayaran dilakukan satu persatu.

Dari hasil komparasi diatas ditemukan bahwa semua *platform* menggunakan metode *self enrollment* menggunakan *email* adapun cara login dengan menggunakan akun pihak ketiga penulis menemukan 6 dari 6 *platform* menyediakan login facebook dan google, 3 dari 6 menggunakan apple dan ada juga beberapa login akun pihak ketiga yang digunakan pada *platform* diatas seperti microsoft dan linkedin.

Sedangkan pada modul *payment* 6 dari 6 *platform* menyediakan metode pembayaran menggunakan kartu kredit, 5 dari 6 *platform* menggunakan paypal dan ada juga beberapa metode lain yg disediakan oleh beberapa *platform* yaitu dapat dengan affirm, alfamart, indomart, doku wallet, ovo, gopay, dan transfer bank.

BAB III

METODE PENELITIAN

3.1 Pengumpulan Data (komunikasi)

Pada bab ini berisi penguraian secara rinci dari sistem informasi MOOC yang akan dibuat, khususnya pada modul *enrollment* dan *payment*. hal ini bertujuan untuk mengidentifikasi dan mengevaluasi permasalahan dan kebutuhan yang ada sehingga dapat menemukan usulan penyelesaian masalah yang sesuai. Adapun tahapan analisis yang dilakukan ini adalah sebagai berikut:

3.1.1 Wawancara

Pada tahap ini dilakukan pembagian kuesioner dan wawancara dengan mahasiswa, tenaga pendidik dan staff akademik yang pernah menggunakan atau setidaknya mengetahui MOOC. Pertanyaan yang ditanyakan bersifat umum dan hanya untuk mengetahui pengetahuan dasar narasumber akan MOOC. Untuk jumlah narasumber yang di targetkan dalam mengisi kuesioner tersebut berjumlah 100 responden.

3.1.2 Diskusi

Selanjutnya semua informasi yang didapat akan disampaikan kepada tim yang nantinya akan dilakukan pencarian solusi untuk permasalahan tersebut. Diskusi ini tidak hanya berfokus pada hasil wawancara dan angket saja melainkan tim juga menggunakan informasi yang ada di internet agar solusi yang ditawarkan dapat menyelesaikan masalah yang ada. Diskusi ini di ikuti oleh semua bagian tim MOOC untuk membahas kebutuhan yang dibutuhkan sistem.

3.2 Identifikasi Masalah

Ditahap ini dilakukan identifikasi masalah dengan menganalisa review-review dari setiap pengguna yang pernah mencoba atau menggunakan *platform* MOOC. Penulis mengumpulkan review-review tersebut dari internet seperti artikel dan jurnal yang membahas MOOC. Penulis juga menggunakan kuesioner dan wawancara dalam mengumpulkan review terhadap *platform* MOOC tersebut. Dari hasil analisis review tersebut penulis mendapatkan wawasan atau pengetahuan tentang bagaimana mengembangkan MOOC modul *enrollment* dan *payment*.

3.3 Perencanaan Secara Cepat

Dalam pengembangan MOOC modul *enrollment* untuk menambah minat peserta dan mempermudah akses pembayaran. Penulis akan melakukan proses analisis yang terdiri dari beberapa analisis kebutuhan, seperti analisis kebutuhan sistem, analisis kebutuhan masukan, analisis kebutuhan output, dan analisis kebutuhan proses.

3.3.1 Analisis Kebutuhan Sistem

Untuk menjalankan modul *enrollment* dan *payment* dalam pengembangan MOOC ini dibutuhkan perangkat lunak yang dapat mendukung agar dapat berjalan dengan baik tanpa terkendala, diantaranya adalah sebagai berikut:

- a. StarUML.
- b. Justinmind.
- c. *Framework Laravel*.
- d. Ngrok.
- e. Visual Studio Code, sebagai *text editor*.
- f. *Web Browser*.

3.3.2 Analisis Kebutuhan Proses, Input, dan Output

Pada tahap ini penulis akan melakukan analisis proses, input, dan output sebagai dasar untuk mengetahui masukan, proses, dan juga output yang diperlukan pada sistem. Adapun proses yang akan dijelaskan pada analisis dibawah ini merupakan gabungan dari beberapa *platform* yang dijadikan referensi awal, hal ini dilakukan guna memberikan sebuah kemudahan proses bagi *user*. Berikut adalah penjelasan mengenai analisis proses, input, dan output yang ditujukan pada Tabel 3. 1.

Tabel 3. 1 Analisis kebutuhan proses, input, dan output

Proses	Input	Output	Keterangan
Mengelola data enrollment	Nama, email dan <i>password</i>	Menampilkan form daftar akun MOOC	Proses mengelola data enrollment ini dimiliki semua <i>platform</i> MOOC dengan role <i>student</i> dan pada proses ini berisi bermacam proses mulai dari pendaftaran akun sampai mengedit data akun
Mengelola data pembayaran	Nama kursus, detail, jumlah, jenis	menampilkan form pembayaran kursus dan	proses mengelola pembayaran dimiliki semua

	pembayaran, waktu, dan tanggal	juga metode yang disediakan <i>platform</i> MOOC	<i>platform</i> MOOC dengan role institusi dan pada proses ini berisi proses pengelolaan pembayaran dikarenakan kursus yang tersedia ada yang berbayar. Pengelolaan pembayaran ini terdiri dari bermacam proses seperti tambah, edit, dan hapus data pembayaran, serta pengelolaan metode yang disediakan pada <i>Platform</i> MOOC
Melihat informasi pembayaran	Nama, tanggal, jumlah, status transaksi, nomer transaksi, nama kursus	Menampilkan form pembayaran dan menampilkan riwayat pembayaran	Proses melihat informasi pembayaran merupakan bagian dari proses mengelola data pembayaran, dimana pada proses ini role yang terlibat adalah <i>student</i> dan institusi
Melihat data enrollment	Nama, email, alamat, foto, tanggal daftar	Menampilkan form data <i>enrollment</i> dari <i>student</i> yang tercatat pada <i>platform</i> MOOC	Pada proses melihat data <i>enrollment</i> , <i>user</i> dengan role <i>student</i> dan institusi dapat mengakses data <i>enrollment</i> hanya saja data yang ditampilkan pada role <i>student</i> dan institusi sedikit berbeda

3.4 Perancangan dan Desain *Prototype*

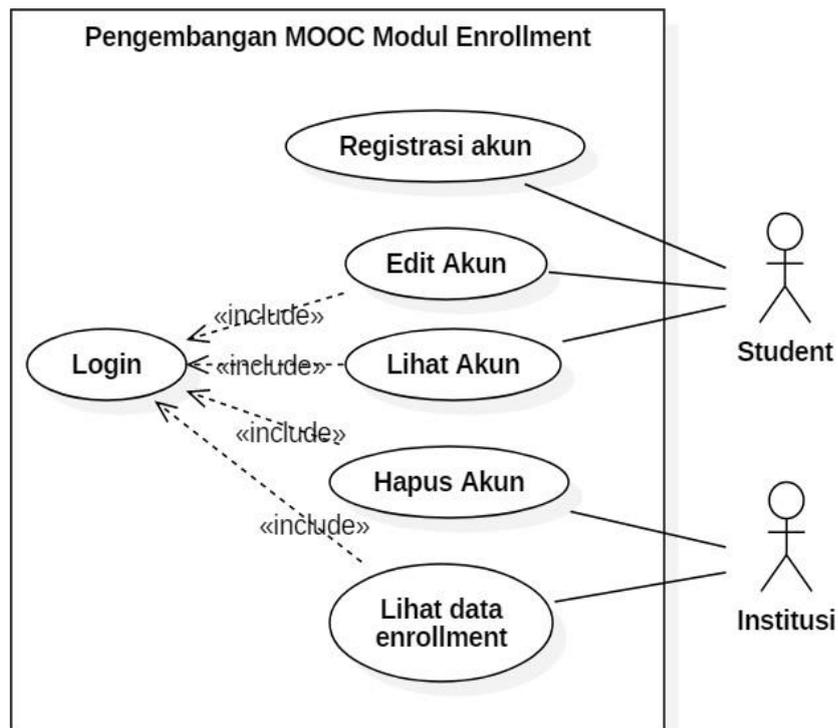
Setelah melakukan identifikasi masalah dan menganalisis, langkah selanjutnya adalah merancang dan mendesain *prototype*. Untuk sistemnya sendiri penulis akan mengembangkan dua modul sesuai dengan judul yang diambil yaitu modul *enrollment* dan modul *payment*.

3.4.1 Perancangan *Use Case Diagram*

Perancangan *Use case diagram* digunakan untuk menggambarkan interaksi – interaksi yang terjadi antara *actor* dengan sistem yang dikembangkan. Adapun *use case* yang penulis buat berfokus pada modul *enrollment* dan *payment* saja.

3.4.1.1 *Use Case Modul Enrollment*

Use case modul *enrollment* ini berisi tentang informasi seputar interaksi-interaksi antara *actor* dengan sistem pada modul *enrollment*. Berikut *use case* yang terdapat pada modul *enrollment* dapat dilihat pada Gambar 3. 1. Gambar 3. 1 *Use Case Diagram* Modul



Gambar 3. 1 Use Case Diagram Modul Enrollment

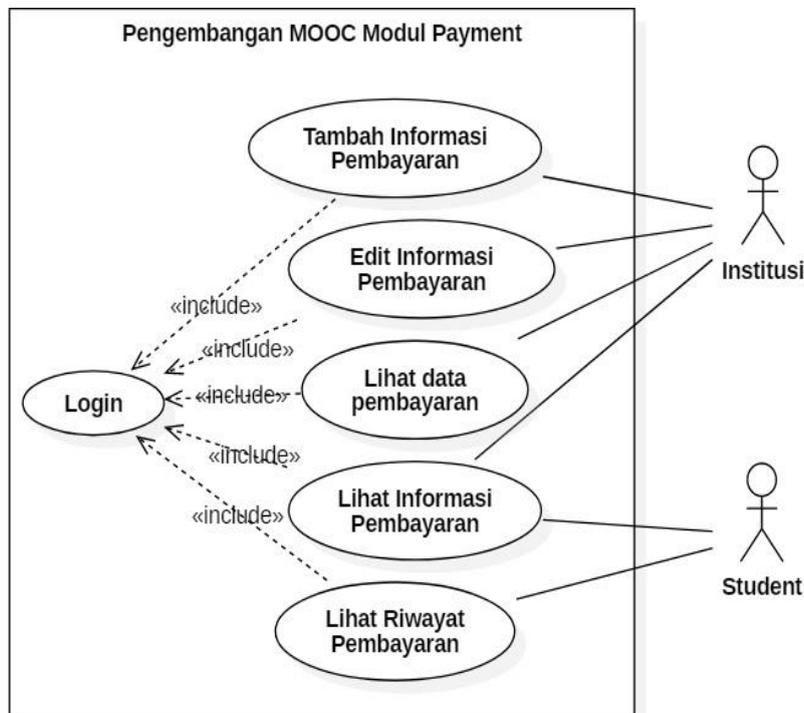
Pada Gambar 3. 2 terdapat aktor *student* dan *institusi*, adapun masing-masing *use case* diharuskan login terlebih dahulu kecuali registrasi akun, hal ini dikarenakan registrasi akun merupakan tahapan paling awal, di mana data yang dibutuhkan oleh sistem untuk login tersedia disini. Berikut penjelasan dari setiap *use case diagram* modul *enrollment*. Pada untuk lebih jelasnya dapat dilihat pada Tabel 3. 2

Tabel 3. 2 Penjelasan Use Case Modul Enrollment

No	Use Case	Deskripsi
UC01	Registrasi Akun	User dengan <i>role</i> peserta dapat mendaftarkan akun ke dalam sistem
UC02	Edit Akun	User dengan <i>role</i> peserta dapat mengedit informasi akun yang sudah didaftarkan sebelumnya ke dalam sistem.
UC03	Hapus Akun	User dengan <i>role</i> institusi dapat menghapus akun yang sudah terdaftar.
UC04	Lihat Akun	User dengan <i>role</i> peserta dapat melihat informasi seputar akun yang sudah didaftarkan oleh peserta tersebut
UC05	Lihat Data Enrollment	User dengan <i>role</i> institusi dapat melihat rekapan data <i>enrollment</i> dari peserta yang sudah tercatat pada <i>database</i> .

3.4.1.2 Use Case Modul Payment

Use case modul *payment* ini berisi tentang informasi seputar interaksi-interaksi antara *actor* dengan sistem pada modul *payment*. Berikut *use case* yang terdapat pada modul *payment* dapat dilihat pada Gambar 3. 2.



Gambar 3. 2 Use Case Diagram Modul Payment

Berikut penjelasan dari setiap *use case diagram* modul *payment*, untuk lebih jelasnya dapat dilihat pada Tabel 3. 3.

Tabel 3. 3 Penjelasan *usecase* modul *payment*

No	Use Case	Deskripsi
UC01	Tambah Informasi Pembayaran	User dengan <i>role</i> institusi dapat menambah informasi pembayaran dapat mendaftarkan akun ke dalam sistem
UC02	Edit Informasi Pembayaran	User dengan <i>role</i> institusi dapat mengedit informasi pembayaran yang sudah didaftarkan sebelumnya ke dalam sistem.
UC03	Lihat Informasi Pembayaran	User dengan <i>role</i> institusi dan peserta dapat melihat informasi pembayaran seputar kursus yang diambil oleh peserta.
UC04	Lihat Data Pembayaran	User dengan <i>role</i> institusi dapat melihat rekapan data <i>payment</i> yang telah dilakukan oleh para peserta
UC05	Lihat Riwayat Pembayaran	User dengan <i>role</i> <i>student</i> dapat melihat riwayat pembayaran yang sudah dibayarkan pada akun terkait.

3.4.2 Perancangan Activity Diagram

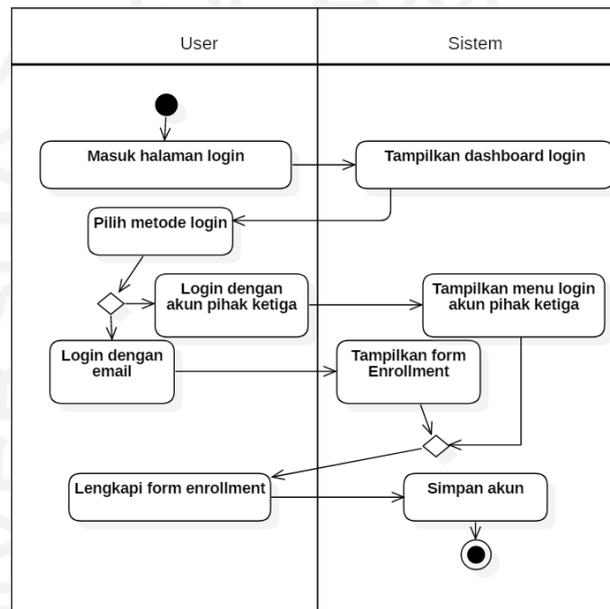
Perancangan *activity diagram* ini digunakan untuk menjelaskan alur kegiatan dari jalannya sistem yang dijelaskan pada *use case diagram*. Pada bagian *Activity diagram* ini penulis akan membuat *activity diagram* yang ada pada modul *enrollment* dan *payment*.

3.4.2.1 Activity Diagram Modul Enrollment

Pada tahap ini akan menjelaskan tiap – tiap alur yang ada pada *usecase* modul *enrollment*.

1. Activity diagram registrasi akun

Alur kerja sistem pada *use case* registrasi akun dapat dilihat pada Gambar 3. 3.



Gambar 3. 3 Activity Diagram Registrasi Akun

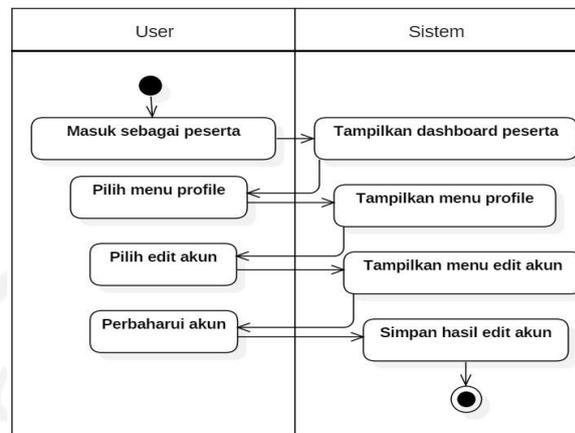
Adapun penjelasan *activity diagram* pada Gambar 3. 3 dapat dilihat pada Tabel 3. 4.

Tabel 3. 4 Penjelasan alur registrasi akun

No	Aktor	Aktivitas
1	User	User masuk ke halaman <i>login</i>
2	Sistem	Sistem menampilkan <i>dashboard login</i>
3	User	User memilih metode <i>login</i> , apakah menggunakan akun pihak ketiga atau dengan mendaftar menggunakan <i>email</i> .
4	Sistem	Sistem menampilkan form sesuai yang dipilih <i>user</i> , jika <i>user</i> memilih menggunakan login dengan akun pihak ketiga, maka sistem akan menampilkan menu untuk login melalui akun pihak ketiga, jika <i>user</i> memilih mendaftar menggunakan email, maka sistem akan menampilkan form <i>enrollment</i> .
5	User	User melengkapi form pendaftaran.
6	Sistem	Menyimpan data akun dari <i>user</i> .

2. Activity diagram edit akun

Alur kerja sistem pada *use case* edit akun dapat dilihat pada Gambar 3. 4.



Gambar 3. 4 Activity Diagram edit akun

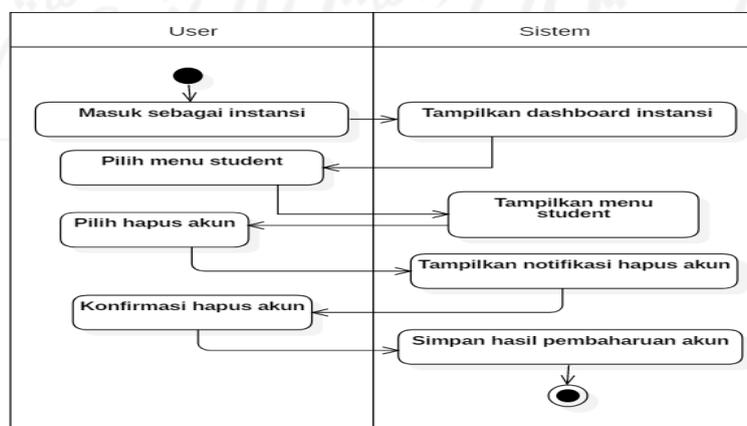
Penjelasan *activity diagram* pada Gambar 3. 4 Gambar 3. 4 Activity Diagram dapat dilihat pada Tabel 3. 5.

Tabel 3. 5 Penjelasan alur kegiatan edit akun

No	Aktor	Aktivitas
1	User	User masuk sebagai peserta
2	Sistem	Sistem menampilkan <i>dashboard</i> peserta
3	User	User memilih menu <i>profile</i>
4	Sistem	Sistem menampilkan menu <i>profile</i>
5	User	User memilih edit akun
6	Sistem	Sistem menampilkan menu edit akun
7	User	User memperbaharui akun
8	Sistem	Sistem menyimpan hasil pembaharuan akun

3. Activity diagram hapus akun

Alur kerja sistem pada *use case* hapus akun dapat dilihat pada Gambar 3. 5.



Gambar 3. 5 Activity diagram hapus akun

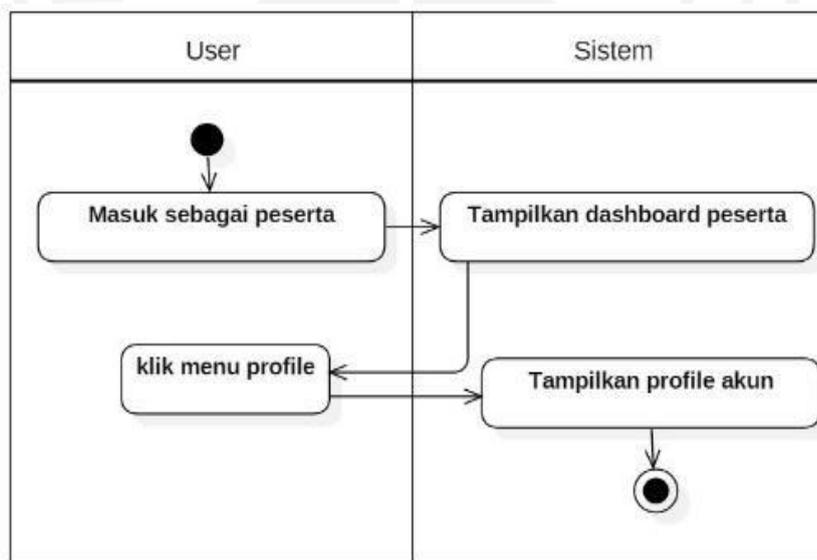
Penjelasan *activity diagram* pada Gambar 3. 5 dapat dilihat pada Tabel 3. 6.

Tabel 3. 6 Penjelasan alur kegiatan hapus akun

No	Aktor	Aktivitas
1	User	User masuk sebagai instansi
2	Sistem	Sistem menampilkan <i>dashboard</i> instansi yang login
3	User	Instansi pilih menu <i>student</i>
4	Sistem	Sistem menampilkan menu <i>student</i>
5	User	User memilih menu hapus akun
6	Sistem	Sistem menampilkan notifikasi hapus akun
7	User	User memilih konfirmasi penghapusan akun
8	Sistem	Sistem menyimpan pembaharuan akun

4. *Activity diagram* lihat akun

Alur kerja sistem pada *use case* lihat akun dapat dilihat pada Gambar 3. 6.



Gambar 3. 6 *Activity diagram* lihat akun

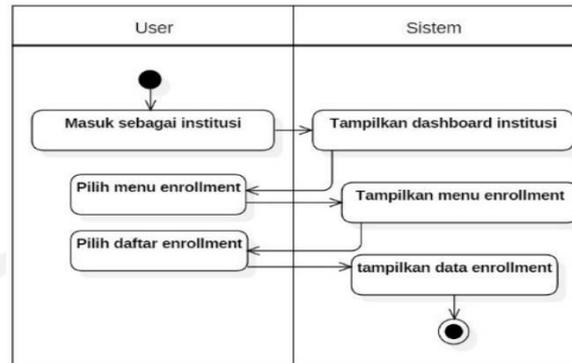
Penjelasan alur *activity diagram* pada Gambar 3. 6 dapat dilihat pada Tabel 3. 7.

Tabel 3. 7 Penjelasan alur lihat akun

No	Aktor	Aktivitas
1	User	User masuk sebagai peserta
2	Sistem	Sistem menampilkan <i>dashboard</i> peserta
3	User	User memilih menu <i>profile</i>
4	Sistem	Sistem menampilkan data <i>profile</i> akun yang bersangkutan

5. Activity diagram lihat data enrollment

Alur kerja sistem pada *use case* lihat data enrollment dapat dilihat pada Gambar 3. 7.



Gambar 3. 7 Activity diagram lihat data enrollment

Penjelasan alur *activity diagram* pada Gambar 3. 7 dapat dilihat pada Tabel 3. 8.

Tabel 3. 8 Penjelasan alur lihat data enrollment

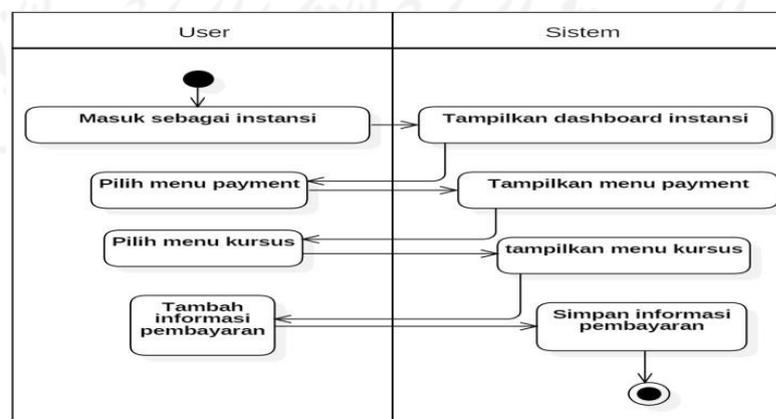
No	Aktor	Aktivitas
1	User	User masuk sebagai instansi
2	Sistem	Sistem menampilkan <i>dashboard</i> instansi
3	User	User memilih menu <i>enrollment</i>
4	Sistem	Sistem menampilkan menu <i>enrollment</i>
5	User	User memilih menu daftar <i>enrollment</i>
6	Sistem	Sistem menampilkan data <i>enrollment</i>

3.4.2.2 Activity Diagram Modul Payment

Pada tahap ini akan menjelaskan tiap – tiap alur yang ada pada *usecase* modul *payment*.

1. Activity diagram tambah informasi pembayaran

Alur kerja sistem pada *use case* tambah informasi pembayaran dapat dilihat pada Gambar 3. 8.



Gambar 3. 8 Activity diagram tambah informasi pembayaran

Penjelasan alur *activity diagram* pada Gambar 3. 8 dapat dilihat pada Tabel 3. 9.

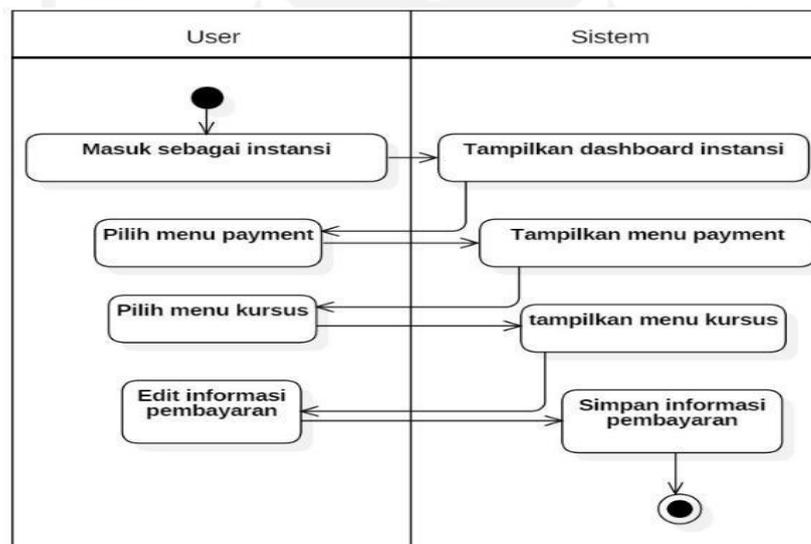
Tabel 3. 9 Penjelasan alur tambah informasi pembayaran

No	Aktor	Aktivitas
1	User	User masuk sebagai instansi
2	Sistem	Sistem menampilkan <i>dashboard</i> instansi
3	User	User menu <i>payment</i>
4	Sistem	Sistem menampilkan menu <i>payment</i>
5	User	User memilih menu kursus
6	Sistem	Sistem menampilkan menu kursus
7	User	User menambah informasi pembayaran
8	Sistem	Sistem menyimpan informasi pembayaran

2. *Activity diagram* edit informasi pembayaran

Alur kerja sistem pada *use case* edit informasi pembayaran dapat dilihat pada Gambar 3.

9.



Gambar 3. 9 *Activity diagram* edit informasi pembayaran

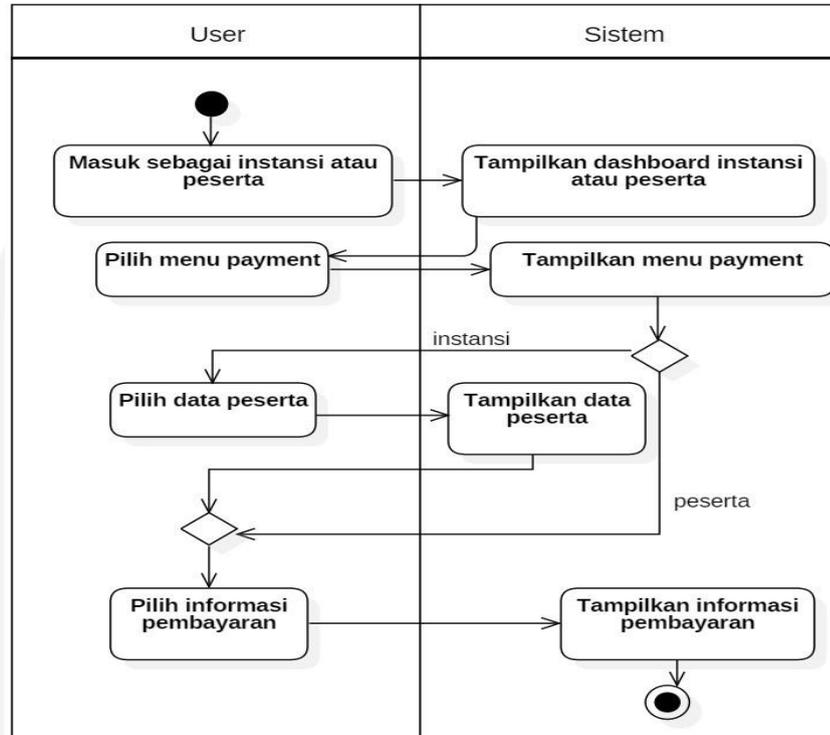
Penjelasan alur *activity diagram* pada Gambar 3. 9 dapat dilihat pada Tabel 3. 10.

Tabel 3. 10 Penjelasan alur edit informasi pembayaran

No	Aktor	Aktivitas
1	User	User masuk sebagai instansi
2	Sistem	Sistem menampilkan <i>dashboard</i> instansi
3	User	User memilih menu <i>payment</i>
4	Sistem	Sistem menampilkan menu <i>payment</i>
5	User	User memilih menu kursus
6	Sistem	Sistem menampilkan menu kursus
7	User	User edit informasi pembayaran
8	Sistem	Simpan informasi pembayaran

3. Activity diagram lihat informasi pembayaran

Alur kerja sistem pada *use case* lihat informasi pembayaran dapat dilihat pada Gambar 3. 10.



Gambar 3. 10 Activity diagram lihat informasi pembayaran

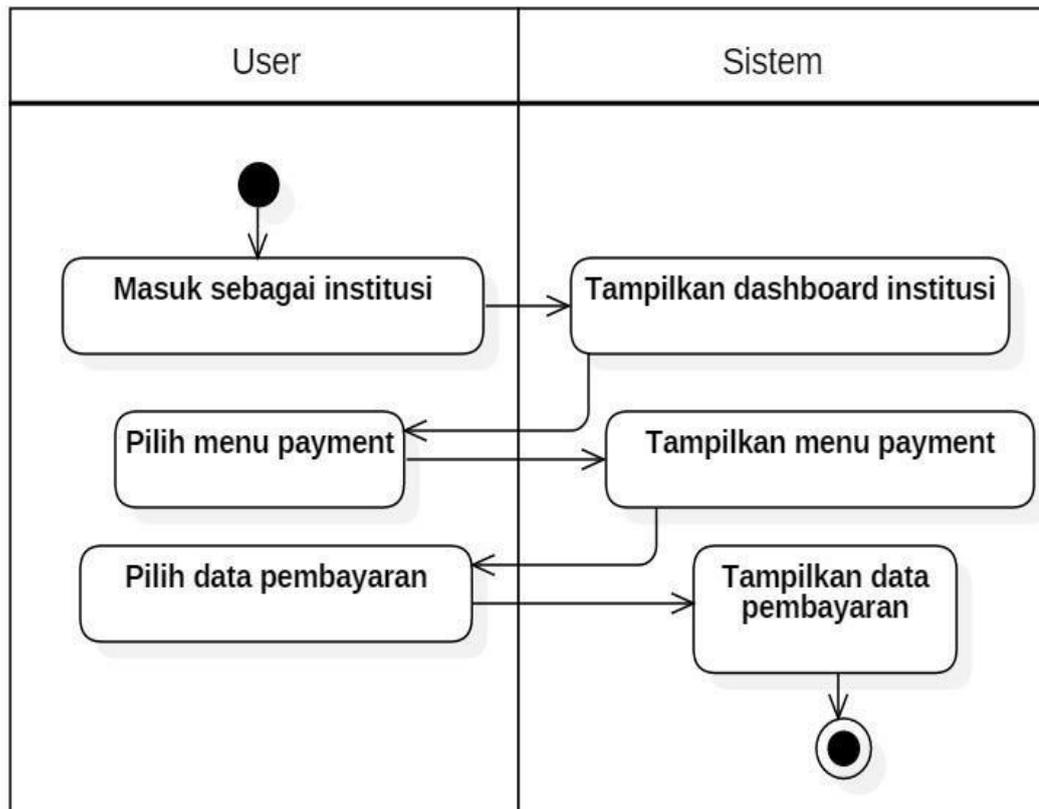
Penjelasan alur *activity diagram* pada Gambar 3. 10 dapat dilihat pada Tabel 3. 11.

Tabel 3. 11 Penjelasan alur lihat informasi pembayaran

No	Aktor	Aktivitas
1	User	User masuk sebagai instansi atau peserta
2	Sistem	Sistem menampilkan <i>dashboard</i> sesuai <i>user</i> yang login
3	User	User dengan <i>role</i> instansi dan peserta memilih menu <i>payment</i>
4	Sistem	Sistem menampilkan menu <i>payment</i>
5	User	User dengan <i>role</i> instansi memilih data peserta
6	Sistem	Sistem menampilkan data peserta
7	User	User dengan <i>role</i> instansi dan peserta memilih menu informasi pembayaran
8	Sistem	Sistem menampilkan data informasi pembayaran

4. Activity diagram lihat data pembayaran

Alur kerja sistem pada *use case* lihat data pembayaran dapat dilihat pada Gambar 3. 11.



Gambar 3. 11 *Activity diagram* lihat data pembayaran

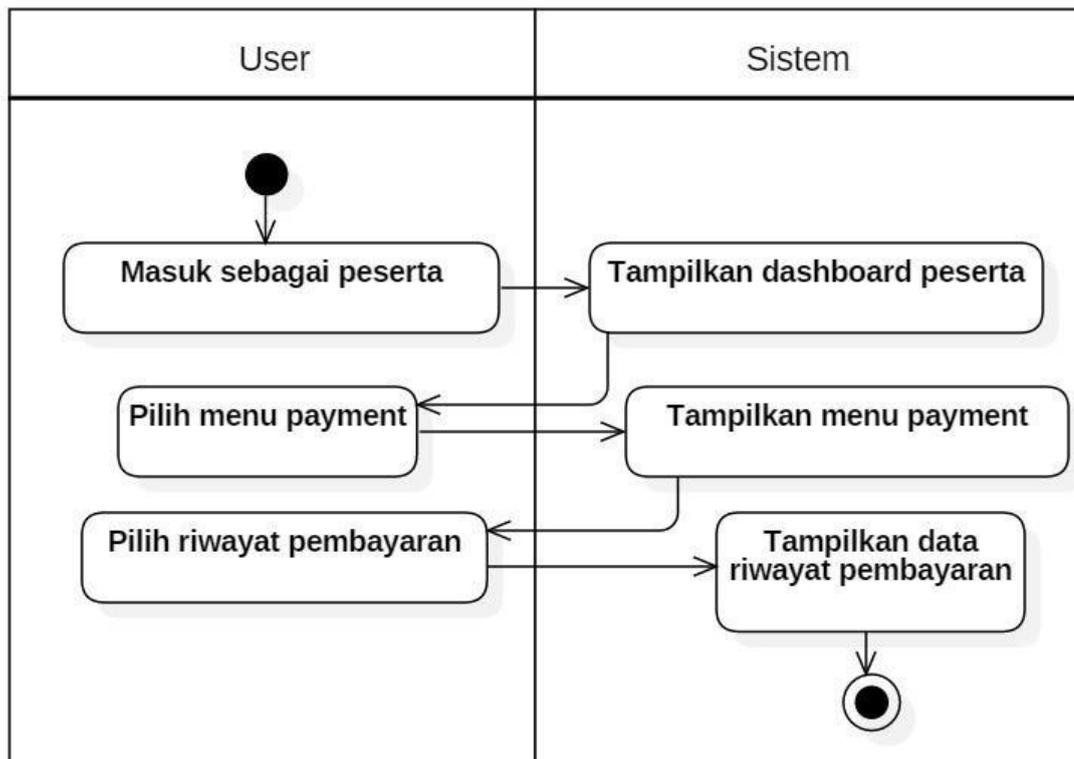
Penjelasan alur *activity diagram* pada Gambar 3. 11 dapat dilihat pada Tabel 3. 12.

Tabel 3. 12 Penjelasan alur lihat data pembayaran

No	Aktor	Aktivitas
1	<i>User</i>	<i>User</i> masuk sebagai instansi
2	Sistem	Sistem menampilkan <i>dashboard</i> institusi
3	<i>User</i>	<i>User</i> memilih menu <i>payment</i>
4	Sistem	Sistem menampilkan menu <i>payment</i>
5	<i>User</i>	<i>User</i> memilih menu data pembayaran
6	Sistem	Sistem menampilkan data pembayaran

5. *Activity diagram* lihat riwayat pembayaran

Alur kerja sistem pada *use case* lihat riwayat pembayaran dapat dilihat pada Gambar 3. 12.



Gambar 3. 12 *Activity diagram* lihat riwayat pembayaran

Penjelasan alur *activity diagram* pada Gambar 3. 12 dapat dilihat pada Tabel 3. 13.

Tabel 3. 13 Penjelasan alur lihat riwayat pembayaran

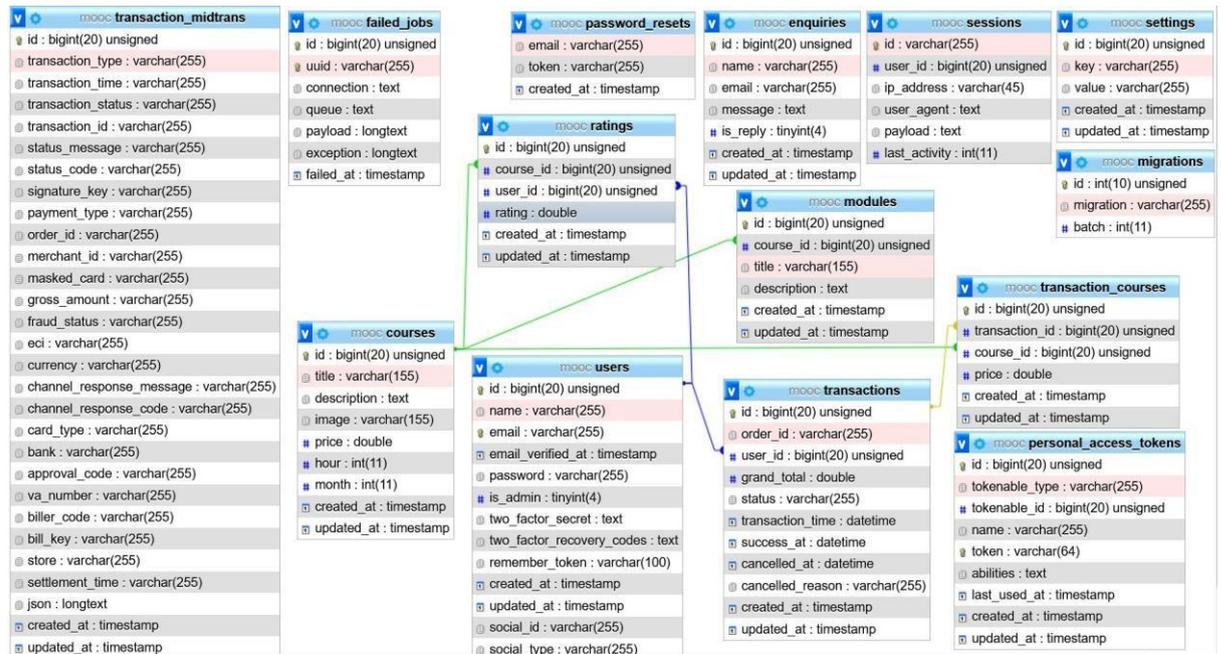
No	Aktor	Aktivitas
1	User	User masuk sebagai peserta
2	Sistem	Sistem menampilkan <i>dashboard</i> peserta
3	User	User memilih menu <i>payment</i>
4	Sistem	Sistem menampilkan menu <i>payment</i>
5	User	User memilih menu riwayat pembayaran
6	Sistem	Sistem menampilkan data riwayat pembayaran

3.5 Perancangan Basisdata

Perancangan basisdata pada sistem MOOC modul *enrollment* dan *payment* terdiri dari 2 proses, yaitu membuat sebuah relasi tabel dan penjelasan struktur tabel dari keseluruhan basisdata yang ada pada sistem.

3.5.1 Relasi Tabel

Relasi tabel merupakan suatu model yang digunakan untuk menjelaskan hubungan antar tabel dan atribut data pada sebuah database. Untuk lebih jelasnya, dapat dilihat pada gambar Gambar 3. 13.



Gambar 3. 13 Relasi basisdata mooc

3.5.2 Struktur Tabel Basisdata

Struktur tabel basisdata merupakan struktur detail tabel yang berada pada basisdata. Struktur tabel juga menunjukkan lebih detail role setiap tabel. Berikut struktur tabel pada basisdata di sistem ini.

1. Tabel *users*

Berikut merupakan tabel *user* yang terdiri dari 14 kolom, untuk detailnya dapat dilihat pada Tabel 3. 14.

Tabel 3. 14 Tabel *users*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	Primary Key
name	varchar	255	
email	varchar	255	
email_verified_at	timestamp	-	
password	varchar	255	
is_admin	Tinyint	4	
phone	varchar	16	
address	text	-	

Nama Kolom	Tipe	Panjang	Keterangan
photo	varchar	200	
social_id	varchar	255	
social_type	varchar	255	
remember_token	varchar	100	
created_at	timestamp	-	
updated_at	timestamp	-	

Tabel ini digunakan untuk menyimpan data dari semua *user*, baik institusi maupun *student* yang sudah mendaftar pada sistem ini. Saat *student* awal login pada sistem data yang dicatat hanya *username*, email, dan *password* saja, selanjutnya jika ingin melengkapi data diri ada pada bagian profile. Adapun data yang disimpan disini merupakan data umum dari *student* seperti nama, email, *password*, alamat, nomer telpon, dan foto. Selain itu, pada tabel *user* juga terapat atribut *social_type* yang menunjukkan cara *user* bergabung ke sistem apakah melalui google atau langsung melalui sistem MOOC.

2. Tabel *course*

Berikut ini adalah tabel *course* yang terdiri dari 9 kolom, untuk detailnya dapat dilihat pada Tabel 3. 15.

Tabel 3. 15 Tabel *course*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
title	varchar	255	
description	text		
image	varchar	155	
<i>price</i>	double		
hour	int	11	
month	int	11	
created_at	timestamp	-	
updated_at	timestamp	-	

Tabel ini digunakan untuk menyimpan data *course* yang telah dibuat oleh *user* institusi. Adapun data yang ada pada tabel ini sama seperti pada *platform* mooc acuan, dimana memiliki data judul, deskripsi singkat, gambar, harga, perkiraan waktu yang dibutuhkan untuk menyelesaikan kursus, dan yang terakhir yaitu pencatatan waktu dibuatnya kursus.

3. Tabel *module*

Berikut adalah tabel dari *module* yang terdiri dari 6 kolom, untuk detailnya dapat dilihat pada Tabel 3. 16.

Tabel 3. 16 Tabel *module*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
course_id	bigint	20	<i>Foreign key (id, course)</i>
title	varchar	255	
description	text		
created_at	timestamp	-	
updated_at	timestamp	-	

Tabel ini digunakan untuk menyimpan data modul dari detail kursus, dimana data modul ini merupakan materi yang akan didapatkan oleh *student* saat mendaftar pada kursus tersebut.

4. Tabel *transaction*

Pada tabel *transaction* ini memiliki 11 kolom yang berisi data seputar transaksi, untuk detailnya dapat dilihat pada Tabel 3. 17.

Tabel 3. 17 Tabel *transaction*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
order_id	varchar	255	
user_id	bigint	20	<i>foreign key (id, user)</i>
grand_total	double		
status	varchar	255	
transaction time	datetime		
success_at	datetime		
cancelled_at	datetime		
cancelled_reason	varchar	255	
created_at	timestamp		
update_at	timestamp		

Pada tabel ini menampung data dari transaksi yang dilakukan oleh *student* saat mendaftar pada kursus, baik yang gratis maupun yang membayar. Pada tabel ini berisi data seperti nomer pemesanan, nomer *user*, status transaksi hingga waktu

transaksi. Selain itu, data transaksi yang dibatalkan juga dicatat meliputi waktu dibatalkan dan alasan dibatalkan. Data transaksi ini akan ditampilkan pada sistem agar instansi dapat mengetahui status perkembangan transaksi dan total transaksi.

5. Tabel *transaction course*

Tabel *transaction course* ini terdiri dari 6 kolom seputar data pada detail pembayaran kursus, untuk detailnya dapat dilihat pada Tabel 3. 18.

Tabel 3. 18 Tabel *transaction course*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
<i>transaction_id</i>	bigint	20	
<i>course_id</i>	bigint	20	
<i>price</i>	double		
<i>created_at</i>	timestamp	-	
<i>updated_at</i>	timestamp	-	

Pada tabel ini menampung daftar *course* pada tiap transaksi. Data didapatkan dari halaman keranjang belanja yang ditambahkan oleh *student*. Tabel ini menampung beberapa data seperti *transaction id*, *course id*, *price* dari *course*, serta pencatatan waktu ditambakkannya *course* ke keranjang belanja tersebut.

6. Tabel *setting*

Tabel *setting* ini terdiri dari 5 kolom seperti *id*, *key*, *value*, *created at*, dan *update at*. Untuk detailnya dapat dilihat pada Tabel 3. 19.

Tabel 3. 19 Tabel *setting*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
key	varchar	255	
value	varchar	255	
<i>created_at</i>	timestamp	-	
<i>updated_at</i>	timestamp	-	

Pada tabel ini berisi data *setting* yang terdiri dari beberapa data seperti *id setting*, *key*, *value*, dan pencatatan waktu akses dengan menggunakan *timestamp*. Data *setting* digunakan untuk mengontrol beberapa hal yang umumnya tidak

memerlukan tabel khusus pada sistem seperti konten about us, konten maintenance, dan lain-lain.

7. Tabel *transaction* midtrans

Tabel *transaction* midtrans ini terdiri dari 31 kolom, untuk detailnya dapat dilihat pada Tabel 3. 20.

Tabel 3. 20 Tabel *transaction* midtrans

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
<i>transaction_type</i>	varchar	255	
<i>transaction_time</i>	varchar	255	
<i>transaction_status</i>	varchar	255	
statustransaction_id	varchar	255	
<i>transaction_status</i>	varchar	255	
<i>transaction_id</i>	varchar	255	
status_message	varchar	255	
status_code	varchar	255	
signature_key	varchar	255	
payment_type	varchar	255	
order_id	varchar	255	
merchant_id	varchar	255	
masked_card	varchar	255	
gross_amount	varchar	255	
fraud_status	varchar	255	
eci	varchar	255	
currency	varchar	255	
channel_response_message	varchar	255	
channel_response_code	varchar	255	
card_type	varchar	255	
bank	varchar	255	
approval_code	varchar	255	
va_number	varchar	255	
biller_code	varchar	255	
bill_key	varchar	255	
store	varchar	255	
settlement_time	varchar	255	
json	longtext		
created_at	timestamp		
update_at	timestamp		

Tabel ini merupakan tabel untuk menyimpan notifikasi transaksi dari pihak midtrans agar institusi dapat mengakses data transaksi yang sebenarnya dari midtrans. Data yang ada di tabel ini tidak akan ditampilkan di sistem akan tetapi lebih bersifat sebagai log sistem atau histori transaksi dari sistem.

8. Tabel *failed jobs*

Tabel *failed jobs* ini terdiri dari 7 kolom, untuk detailnya dapat dilihat pada Tabel 3. 21.

Tabel 3. 21 Tabel *failed jobs*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
uuid	varchar	255	
connection	text		
queue	text		
payload	longtext		
exception	longtext		
failed_at	timestamp		

Tabel ini menampung beberapa data seperti id, uuid, *connection*, *queue*, *payload*, *exception*, dan *failed at* menggunakan pencatatan waktu. Tabel ini berfungsi sebagai log *error* dari fungsi-fungsi yang berjalan di balik layar seperti notifikasi email, dan lain-lain. Tabel ini merupakan default bawaan dari Laravel.

9. Tabel *password reset*

Berikut merupakan tabel dari *password reset* yang terdiri dari 3 kolom, detail tabelnya dapat dilihat pada Tabel 3. 22.

Tabel 3. 22 Tabel *password reset*

Nama Kolom	Tipe	Panjang	Keterangan
email	varchar	255	
token	varchar	255	
created_at	timestamp		

Tabel ini menampung beberapa data seperti email, token, dan pencatatan waktu menggunakan *time stamp*. Tabel ini merupakan default bawaan Laravel dan berfungsi untuk mengatur proses reset *password*.

10. Tabel *ratings*

Berikut merupakan tabel *ratings* yang terdiri dari 6 kolom, untuk detailnya dapat dilihat pada Tabel 3. 23.

Tabel 3. 23 Tabel *ratings*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
course_id	bigint	20	
user_id	bigint	20	
rating	double		
created_at	timestamp		
updated_at	timestamp		

Tabel ini berisi beberapa data seperti id, *course id*, *user id*, *rating*, *created at*, dan *update at* dalam mencatat waktu dengan menggunakan *timestamp*. Tabel ini berfungsi untuk menyimpan data ratings dari tiap *user* terhadap *course-course* tertentu. Nilai suatu *course* akan terakumulasi dari rating-rating yang diberikan *user* terhadap *course* tersebut.

11. Tabel *enquiries*

Tabel *enquiries* ini berisi 7 kolom, untuk detailnya dapat dilihat pada Tabel 3. 24.

Tabel 3. 24 Tabel *enquiries*

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
name	varchar	255	
email	varchar	255	
message	text		
is_reply	tinyint	4	
created_at	timestamp		
updated_at	timestamp		

Tabel ini menampung data pada halaman *contact us*, dimana *student* maupun calon *student* dapat menanyakan pertanyaan dengan cara mengisi form pada halaman tersebut, adapun data yang digunakan pada tabel ini yaitu id, *name*, *email*, *message*, *reply*, *created at*, dan *update at* menggunakan *timestamp*.

12. Tabel *sessions*

Tabel *session* ini terdiri dari 6 kolom, untuk detailnya dapat dilihat pada Tabel 3. 25.

Tabel 3. 25 Tabel *session*

Nama Kolom	Tipe	Panjang	Keterangan
id	varchar	255	<i>Primary Key</i>
<i>user_id</i>	bigint	20	
<i>ip_address</i>	varchar	45	
<i>user_agent</i>	text		
<i>payload</i>	text		
<i>last_activity</i>	int	11	

Tabel ini berisi data yang digunakan untuk mengatur lama sesi login saat tidak ada interaksi, adapun data yang ada pada tabel ini yaitu *id session*, *id user*, *ip address*, *user agent*, *payload*, dan *last activity*.

13. Tabel personal akses token

Tabel personal akses token berisi 9 kolom, adapun detail tabelnya dapat dilihat pada Tabel 3. 26.

Tabel 3. 26 Tabel personal akses token

Nama Kolom	Tipe	Panjang	Keterangan
id	bigint	20	<i>Primary Key</i>
<i>tokenable_type</i>	varchar	255	
<i>tokenable_id</i>	bigint	20	
<i>name</i>	varchar	255	
<i>token</i>	varchar	64	
<i>abilities</i>	text		
<i>lastu_used_at</i>	timestamp		
<i>created_at</i>	timestamp		
<i>updated_at</i>	timestamp		

Tabel ini berisi beberapa data seperti *id*, *tokenable type*, *tokenable id*, *name*, *token*, *abilities*, *last used at*, *created at*, dan *update at* menggunakan *timestamp*. Tabel ini merupakan default bawaan dari Laravel yang berfungsi untuk mengatur autentikasi *user* dengan akses token tertentu.

3.6 Perancangan Antarmuka

Setelah menganalisis segala kebutuhan yang diperlukan, selanjutnya penulis melakukan desain antarmuka/UI *prototype*. Proses desain dilakukan dengan membuat *mock-up user interface* (UI) *prototype* dan dalam membuat *mock-up* penulis akan menggunakan *tools* Justinmind. Dalam proses desain ini penulis akan mengelompokkan alur proses berdasarkan

user sebagai skenario untuk memudahkan perbaikan saat desain yang dibuat kurang sesuai dengan kebutuhan *user*.

3.6.1 Skenario *User Student*

Skenario *user student* ini berisi seluruh alur proses yang terjadi pada *user student*, mulai dari saat awal memasuki sistem, sampai selesai melakukan pembayaran.

3.6.1.1 Desain antarmuka *user student*

Pada tahapan ini penulis akan melakukan pemodelan desain antarmuka pada *user student*.

a. Desain antarmuka halaman awal sistem

Pada halaman ini sistem akan menampilkan halaman awal sistem. Jadi pada saat pertama kali membuka sistem, maka baik *user student* maupun institusi akan dihadapkan dengan halaman ini. Tampilan didalamnya pun terdiri dari beberapa menu seperti rekomendasi kursus terbaik, cari lebih banyak kursus, serta login pada sistem. Detail desain halaman awal sistem dapat dilihat pada Gambar 3.14

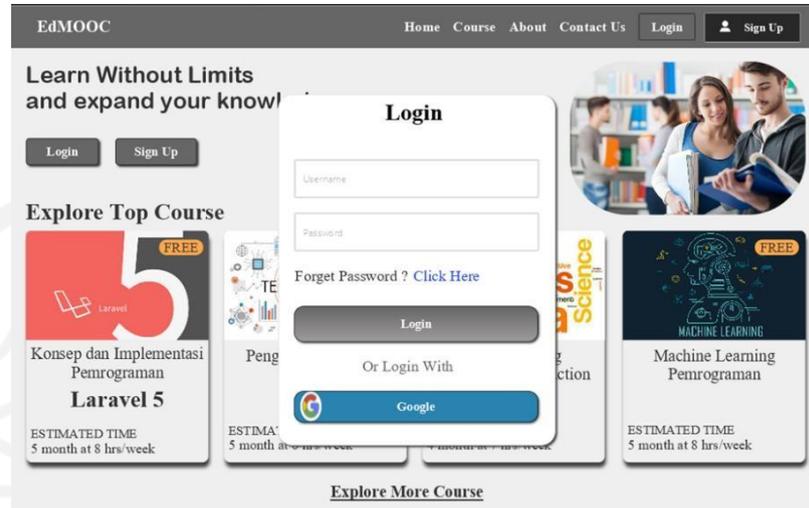


Gambar 3. 14 Halaman awal sistem

b. Desain antarmuka halaman login

Pada halaman ini akan menampilkan pop up untuk login ke dalam sistem, dapat dengan mendaftarkan manual dengan mengisi form email, atau dapat juga dengan menggunakan akun google, Adapun untuk menampilkan pop up

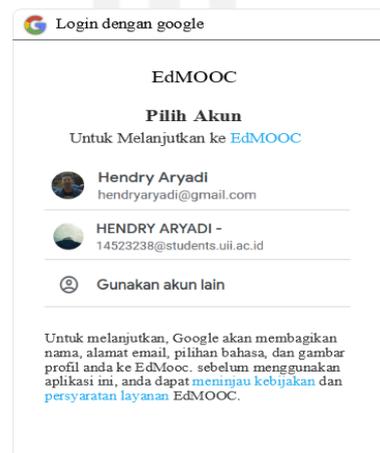
login dapat dengan mengklik tombol login yang ada pada tengah halaman maupun pojok kanan halaman. Untuk lebih jelasnya desain halaman login dapat dilihat pada Gambar 3. 15.



Gambar 3. 15 Halaman login

c. Desain antarmuka halaman pilihan login

Pada halaman ini akan menampilkan beberapa pilihan untuk login ke sistem, *student* dapat memilih untuk mendaftar dengan mengisi form email, atau dengan login google. Untuk lebih jelasnya dapat dilihat pada Gambar 3. 16 dan Gambar 3. 17.



English (United States) ▾ Help Privacy Terms

Gambar 3. 16 Halaman login google

Gambar 3. 17 Halaman login email

d. Desain antarmuka halaman reset *password*

Pada halaman ini sistem akan menampilkan halaman reset *password* (kata sandi), baik saat belum login maupun saat sedang login. Jika belum login, tombol reset *password* ada pada halaman pop up, sehingga saat diklik maka sistem akan menampilkan halaman reset *password*, tepat setelah *student* memasukan email yang *password*nya lupa tersebut, maka sistem akan mengirimkan form ganti *password* lewat email tersebut. Akan tetapi jika sudah login, *student* hanya perlu menuju profil dan klik ganti *password*. Untuk lebih jelasnya dapat dilihat pada Gambar 3. 18 dan Gambar 3. 19.

Gambar 3. 18 Halaman ganti *password*

Gambar 3. 19 Halaman ganti *password* email

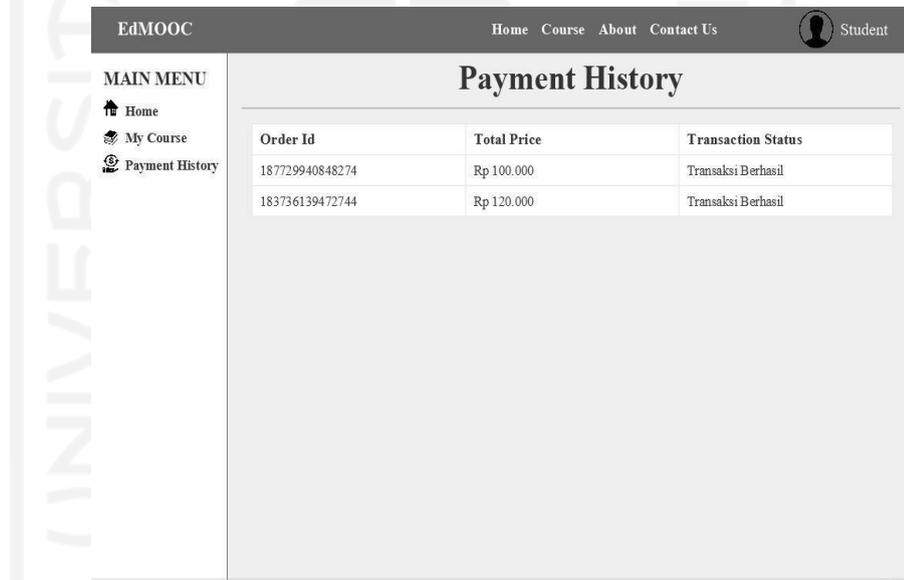
e. Desain antarmuka halaman *dashboard student*

Pada halaman ini adalah halaman awal saat *student* berhasil login ke sistem, pada halaman ini sistem akan menampilkan data dari akun tersebut seperti jumlah kursus yang di ikuti, jumlah kursus yang sudah diselesaikan, dan juga ada kursus yang terakhir diikuti (*enroll*). Sistem juga memberikan beberapa pilihan menu seperti kursus saya, riwayat pembayaran. Seperti pada Gambar 3. 20, Gambar 3. 22, dan Gambar 3. 21.

Gambar 3. 20 Halaman *dashboard* student



Gambar 3. 22 Halaman kursus saya



Gambar 3. 21 Halaman riwayat pembayaran

f. Desain antarmuka halaman cari kursus

Pada halaman cari kursus, sistem akan menampilkan data dari bermacam kursus yang ada pada sistem MOOC. Saat *student* mengklik salah satu kursus, maka sistem akan menampilkan detail kursus tersebut. *Student* juga dapat melihat metode pembayaran dari kursus berbayar yaitu dengan cara mengklik tombol *enroll* yang ada tertera harganya, maka sistem akan menampilkan halaman metode pembayaran dan saat telah dibayar, maka detail materi kursus berbayar akan terbuka. Untuk lebih jelasnya dapat dilihat pada Gambar 3. 24, Gambar 3. 23, Gambar 3. 25, dan Gambar 3. 26.

Gambar 3. 24 Halaman cari kursus

List Modules	Total Syllabus
Module 1	2
Module 2	4

Gambar 3. 23 Halaman detail kursus

Gambar 3. 25 Halaman metode pembayaran

The screenshot shows the EdMOOC interface for a course. The top navigation bar includes 'EdMOOC', 'Home', 'Course', 'About', 'Contact Us', and a 'Student' profile icon. A left sidebar contains a 'MAIN MENU' with 'Home', 'My Course', and 'Payment History'. The main content area displays the course title 'Proses Mining Data Science In Action' and a 'Deskripsi Kursus' section with a word cloud image and a date of '20 November 2021'. Below this is a 'List Modules' table.

List Modules	Total Syllabus
Module 1 Pengenalan Materi lanjutan Quis	3
Module 2 Pengenalan Materi lanjutan Quis Tugas Besar	4

Gambar 3. 26 Halaman kursus yang sudah dibeli

g. Desain antarmuka halaman profil *student*

Pada halaman ini, sistem menampilkan beberapa menu yang memuat data *student*, seperti lihat profil *student* dan edit profil *student*. Halaman ini dapat diakses *student* dengan cara klik icon *student* di pojok kanan atas. Untuk lebih jelasnya dapat dilihat pada Gambar 3. 27 dan Gambar 3. 28.

The screenshot shows the 'My Profile' page in the EdMOOC system. The top navigation bar is identical to the previous image. The main content area is titled 'My Profile' and features a profile picture placeholder (a square with an 'X'). To the right of the placeholder, the following information is displayed: Username: **Hendryaryadi**, Full Name: **Hendry Aryadi**, Phone Number: +628772842****, and Address: Jalan kaliurang km 14.5, Ngaglik Yogyakarta. At the bottom of the profile section, there is an 'Edit My Profile' button.

Gambar 3. 27 Halaman profil student

Gambar 3. 28 Halaman edit profile

3.6.1.2 Pengujian *prototype*

Setelah selesai membuat prototype, selanjutnya penulis menguji prototype kepada pengguna akhir. Metode pengujian ini menggunakan metode sampling acak yang terdiri dari mahasiswa dan tenaga pendidik yang sebelumnya telah mengisi kuesioner pada tahap pengumpulan data. Hasil dari pengujian yang dilakukan dengan menggunakan metode wawancara dapat dilihat pada Tabel 3. 27.

Tabel 3. 27 Hasil pengujian *user (Student)*

Responden	Hasil Evaluasi	Error/bug
R1	Secara keseluruhan sudah bagus. Pada alur proses login, daftar kursus, detail kursus, kelola akun, dan kelola pembayaran sudah sesuai dengan kebutuhan awal dan tidak ditemukan bug/error.	-
R2	Penerapan fitur dan alur proses pada <i>prototype</i> sudah sesuai dan tidak ditemukan bug maupun error	-

Dari hasil pengujian tersebut, pengguna akhir sudah cukup puas dengan prototype yang dibuat dan tidak memberikan masukan. Oleh karena itu penulis melanjutkan ke proses skenario selanjutnya.

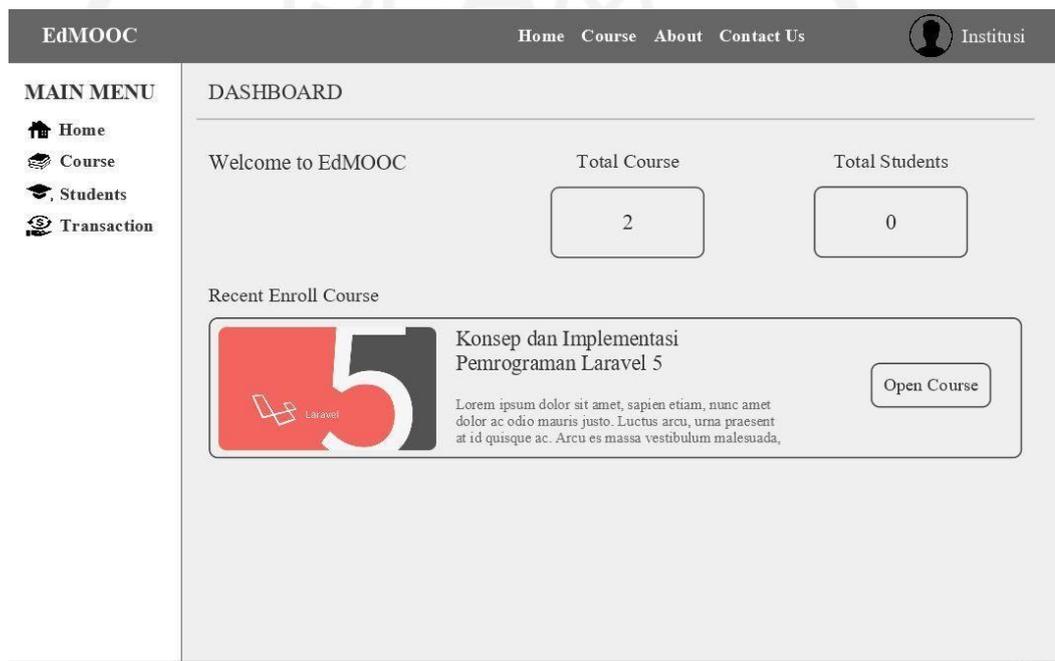
3.6.2 Skenario *User Institusi*

Pada tahapan ini penulis akan melakukan pemodelan desain antarmuka pada *user* institusi.

3.6.2.1 Desain antarmuka *user* institusi

1. Desain antarmuka halaman *dashboard* institusi

Pada halaman ini sistem akan menampilkan *dashboard* untuk *user* institusi. Jadi saat *user* institusi berhasil login akan langsung diarahkan pada halaman ini, jika gagal maka akan kembali ke halaman awal sistem yang sama seperti saat *user student* belum *login*. Adapun beberapa fitur yang ada pada halaman ini yaitu seperti *course*, *student*, *transaction*, info total *course*, dan total *student*. Desain halaman *dashboard* institusi dapat dilihat pada Gambar 3. 29.



Gambar 3. 29 Halaman *dashboard* institusi

2. Desain antarmuka halaman *course*

Pada halaman ini sistem akan menampilkan daftar kursus yang sudah diinputkan dalam sistem. Pada halaman tersebut, sistem juga menyediakan tombol untuk menambah kursus baru, lihat kursus, edit kursus, dan hapus kursus, akan tetapi pada menu hapus *course* hanya muncul konfirmasi kecil saja, sehingga tidak membuka halaman baru seperti menu yang lain. Untuk detail desainnya dapat dilihat pada Gambar 3. 30.

No	Title	Image	Price	Hour/Week	Month	Module	Syllabus	Action
1	Konsep dan Implementasi Pemrograman Laravel 5		-	8	5	-	-	
2	Pengantar Teknologi Informasi		-	6	5	-	-	
3	Proses Mining Data Science in Action		Rp 150.000	7	4	2	6	
4	Machine Learning		-	8	5	-	-	

Gambar 3. 30 Halaman *course* (institusi)

3. Desain antarmuka halaman *create course*

Pada halaman ini sistem akan menampilkan halaman yang berisi form untuk membuat sebuah *course*. Pada halaman tersebut juga berisi menu add module dan add syllabus. Modul sendiri adalah sebuah bab materi dalam suatu kursus, sedangkan syllabus adalah sub bab dari materi pada kursus. Untuk detail desain antarmuka halaman *create course* dapat dilihat pada Gambar 3. 31.

Gambar 3. 31 Halaman *create course*

4. Desain antarmuka halaman edit *course*

Pada halaman ini sistem akan menampilkan halaman edit *course*, dimana pada halaman tersebut mirip seperti pada halaman create *course*, hanya saja pada halaman ini data kursus sudah ada, sehingga akan memudahkan institusi dalam mengedit kursus. Untuk detail desain antarmuka halaman edit *course* dapat dilihat pada Gambar 3. 32.

Gambar 3. 32 Halaman edit *course*

5. Desain antarmuka halaman detail *course*

Pada halaman ini sistem akan menampilkan halaman yang berisi detail *course*, dari mulai judul, review materi singkat, detail harga, estimasi waktu mengikuti kursus hingga selesai dan juga ada jumlah *student* yang mengikuti kursus tersebut. Untuk detail halaman detail *course* dapat dilihat pada Gambar 3. 33.

List Modules	Total Syllabus
Module 1 Pengenalan Materi lanjutan Quis	3
Module 2 Pengenalan Materi lanjutan Quis Tugas Besar	4

Gambar 3. 33 Halaman detail *course*

6. Desain antarmuka halaman *student*

Pada halaman ini, sistem akan menampilkan halaman list *student*, dimana pada halaman tersebut juga terdapat beberapa menu seperti lihat detail *student* dan hapus data *student*. Pada fitur lihat detail *student*, sistem akan menampilkan data lengkap dari *user student* dan untuk fitur hapus akun disini dihadirkan untuk mengantisipasi jika ada *student* yang berbuat curang dengan mengikuti kursus berbayar tanpa melakukan pembayaran. Untuk detail antarmuka halaman *student* dapat dilihat pada Gambar 3. 34 dan Gambar 3. 35.

No	Name	Email	Image	Address	Date Enrollment	Action
1	Hendry Aryadi	hendryaryadi@gmail.com		Jalan kaliurang km 14.5 Ngaglik Yogyakarta	20-11-2021	

Gambar 3. 35 Halaman student

Name	Email	Image	Address	Date Enrollment	Action
Hendry Aryadi	hendryaryadi@gmail.com		Jalan kaliurang km 14.5 Ngaglik Yogyakarta	20-11-2021	

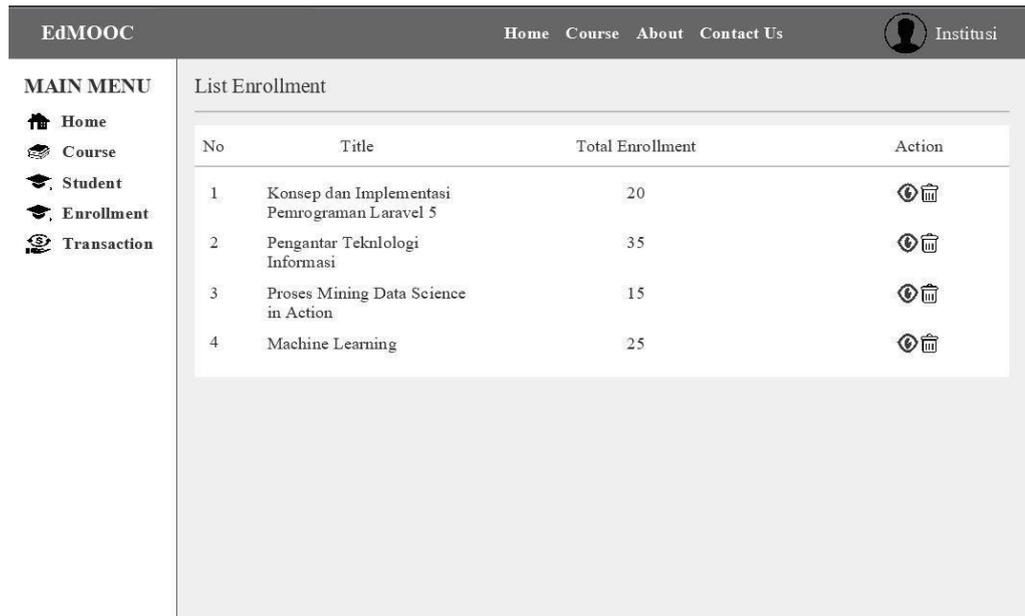
No	Name	Title Course	Date Enroll	Status	Total Course
1	Hendry Aryadi	Pengantar Teknologi Informasi	20-11-21	Not Accomplish	1

Order Id	Total Price	Transaction Status
187729940848274	Rp 100.000	Transaksi Berhasil
183736139472744	Rp 120.000	Transaksi Berhasil

Gambar 3. 34 Halaman detail student

7. Desain antarmuka halaman *enrollment*

Pada halaman ini, sistem akan menampilkan halaman yang berisi list *enrollment*. Dimana pada halaman tersebut terdapat beberapa menu seperti lihat kursus yang saat diklik akan menuju pada halaman detail kursus dan hapus kursus. Untuk detail desain halaman *enrollment* dapat dilihat pada Gambar 3. 36.

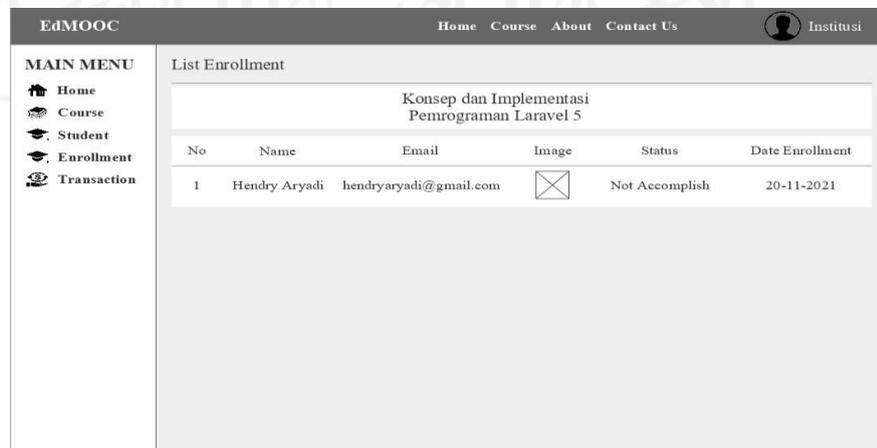


No	Title	Total Enrollment	Action
1	Konsep dan Implementasi Pemrograman Laravel 5	20	 
2	Pengantar Teknologi Informasi	35	 
3	Proses Mining Data Science in Action	15	 
4	Machine Learning	25	 

Gambar 3. 36 Halaman *enrollment* (institusi)

8. Desain antarmuka halaman list *enrollment course*

Pada halaman ini, sistem akan menampilkan halaman yang berisi list dari *student* yang mendaftar pada kursus tersebut. Untuk menuju halaman ini institusi harus masuk terlebih dulu pada halaman enrollment, lalu pilih kursus yang ingin dilihat. untuk detailnya dapat dilihat pada Gambar 3. 37.



Konsep dan Implementasi Pemrograman Laravel 5						
No	Name	Email	Image	Status	Date Enrollment	
1	Hendry Aryadi	hendryaryadi@gmail.com		Not Accomplish	20-11-2021	

Gambar 3. 37 Halaman list *enrollment course*

9. Desain antarmuka halaman *transaction* (institusi)

Pada halaman ini, sistem akan menampilkan halaman yang berisi data transaksi dari semua *student*. Pada halaman ini institusi hanya dapat melihat data transaksi saja. Untuk detail desain halaman *transaction* dapat dilihat pada Gambar 3. 38.

Nama	Email	Order Id	Title Course	Total Price	Date
Hendry Aryadi	hendryaryadi@gmail.com	187729940848274	Pengantar Teknologi Informasi	Rp 100.000	20-11-2021
Hendry septiawan	hendryseptiawan@gmail.com	183736139472744	Pengantar Teknologi Informasi	Rp 100.000	22-11-2021

Gambar 3. 38 Halaman *transaction* (institusi)

3.6.2.2 Pengujian prototype

Setelah selesai membuat prototype, selanjutnya penulis menguji prototype kepada pengguna akhir. Adapun sampel pengujian diambil secara acak yang terdiri dari tenaga pendidik yang sebelumnya telah mengisi kuesioner pada tahap pengumpulan data. Adapun hasilnya dapat dilihat pada tabel Tabel 3. 28 Hasil pengujian *user* (institusi)Tabel 3. 28 Hasil pengujian *user* (institusi)Tabel 3. 28.

Tabel 3. 28 Hasil pengujian *user* (institusi)

Responden	Hasil Evaluasi	Error/bug
R1	Alur proses yang ada pada <i>user</i> institusi ini berjalan normal, akan tetapi pada beberapa halaman ada menu yang hilang, sisanya sudah cukup baik	Menu <i>enrollment</i> hilang pada halaman detail <i>course</i> , <i>edit course</i> , dan <i>create course</i> .
R2	Secara keseluruhan sudah bagus, hanya saja pada beberapa halaman seperti <i>list course</i> dan <i>dashboard</i> institusi, menu <i>enrollment</i> nya hilang.	Menu <i>enrollment</i> hilang pada halaman <i>list course</i> dan <i>dashboard</i> institusi.

Dari hasil pengujian tersebut, penulis mendapat masukan dimana ada bug pada halaman detail *course*, edit *course*, create *course*, list *course*, dan *dashboard* institusi, dimana menu enrollment hilang.

3.6.3 Perbaikan Skenario *User* Institusi

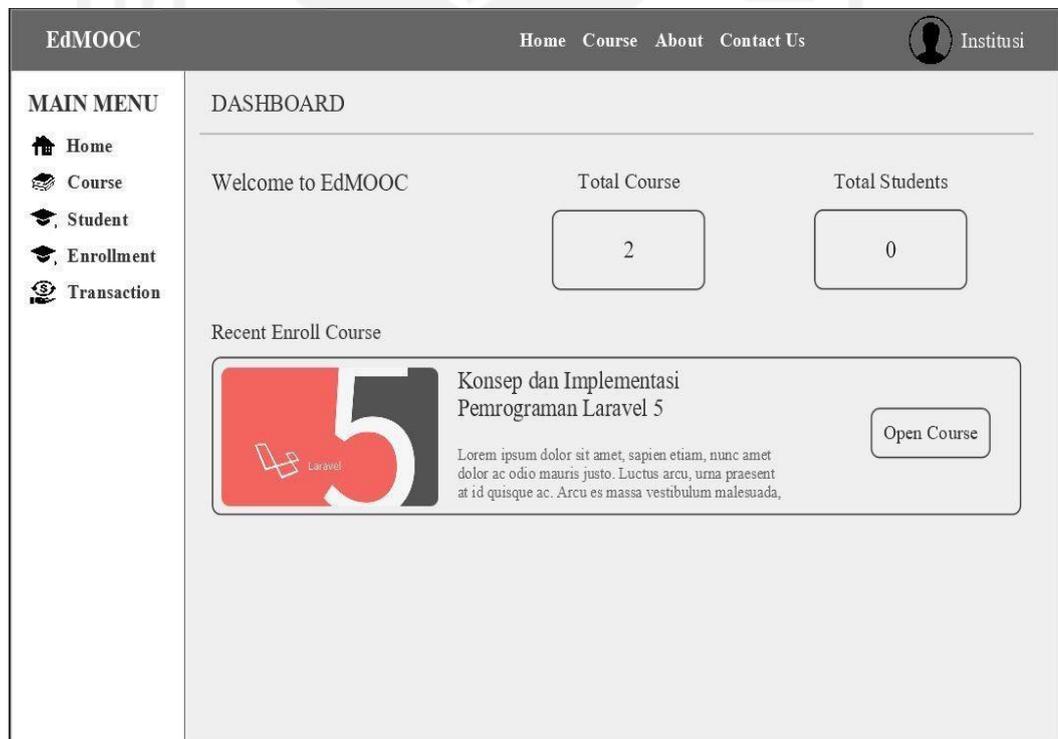
Pada tahap ini berisi perbaikan dari hasil pengujian sebelumnya yang berfokus pada *error* maupun *bug* pada menu yang hilang.

3.6.3.1 Desain antarmuka *user* institusi

Setelah mendapatkan masukan, penulis melakukan perbaikan desain antarmuka pada halaman yang bermasalah. Berikut hasil desain antarmuka pada *user* institusi.

1. Desain antarmuka halaman *dashboard* institusi

Halaman *dashboard* institusi setelah dilakukan perbaikan ulang dapat dilihat pada Gambar 3. 39.



Gambar 3. 39 Halaman *dashboard* institusi

2. Desain antarmuka halaman *list course*

Halaman *list course* setelah dilakukan perbaikan ulang dapat dilihat pada Gambar 3. 40.

No	Title	Image	Price	Hour/Week	Month	Module	Syllabus	Action
1	Konsep dan Implementasi Pemrograman Laravel 5		-	8	5	-	-	
2	Pengantar Teknologi Informasi		-	6	5	-	-	
3	Proses Mining Data Science in Action		Rp 150.000	7	4	2	6	
4	Machine Learning		-	8	5	-	-	

Gambar 3. 40 Halaman *list course*

3. Desain antarmuka halaman *create course*

Halaman *create course* setelah dilakukan perbaikan ulang dapat dilihat pada Gambar 3. 41.

Gambar 3. 41 Halaman *create course*

4. Desain antarmuka halaman *edit course*

Halaman *edit course* setelah dilakukan perbaikan ulang dapat dilihat pada Gambar 3. 42.

Gambar 3. 42 Halaman *edit course*

5. Desain antarmuka halaman detail *course*

Halaman detail *course* setelah dilakukan perbaikan ulang dapat dilihat pada Gambar 3. 43.

List Modules	Total Syllabus
Module 1 Pengenalan Materi lanjutan Quis	3
Module 2 Pengenalan Materi lanjutan Quis Tugas Besar	4

Gambar 3. 43 Halaman detail *course*

3.6.3.2 Pengujian prototype

Setelah selesai membuat prototype, selanjutnya penulis menguji prototype kepada pengguna akhir. Adapun sampel penguji diambil secara acak yang terdiri dari tenaga pendidik yang sebelumnya telah mengisi kuesioner pada tahap pengumpulan data. Hasil dari pengujian pada tahap ini dapat dilihat pada Tabel 3. 29.

Tabel 3. 29 Hasil pengujian perbaikan *user* (institusi)

Responden	Hasil Evaluasi	Error/bug
R1	Proses yang ada pada <i>user</i> institusi berjalan lancar, sudah sesuai dengan kebutuhan awal, dan tidak ditemukan <i>error</i> atau <i>bug</i> .	-
R2	Pada alur proses dan fungsi tiap menu yang ada pada <i>user</i> institusi, semuanya berjalan dengan baik dan tidak ditemukan <i>error</i>	-

Dari hasil pengujian tersebut, pengguna akhir sudah cukup puas dengan prototype yang dibuat dan tidak memberikan masukan lagi.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

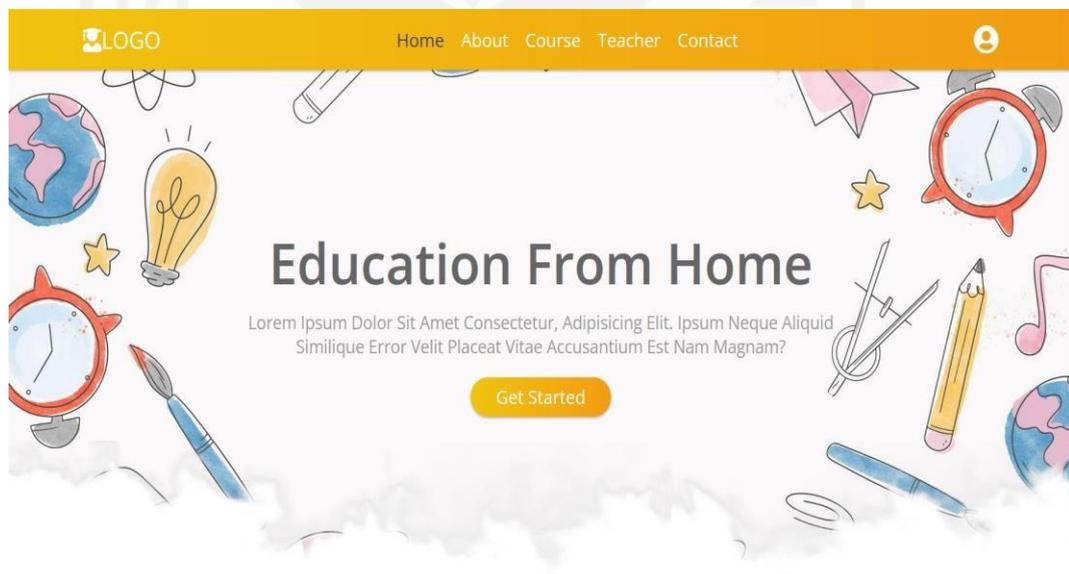
Setelah tidak ada lagi saran dan perbaikan, tahap selanjutnya yaitu implementasi program. Skenario pada sistem ini akan penulis bagi menjadi 2, yaitu dari sisi *student* dan institusi, adapun langkahnya akan dimulai semua dari login hingga cek pembayaran.

4.1.1 Skenario User Student

Skenario *user student* ini terdiri dari beberapa tahapan dimulai dari proses pendaftaran sampai pada pengecekan riwayat pembayaran. Berikut penulis akan menjelaskan secara detail tahapan skenario *user student*.

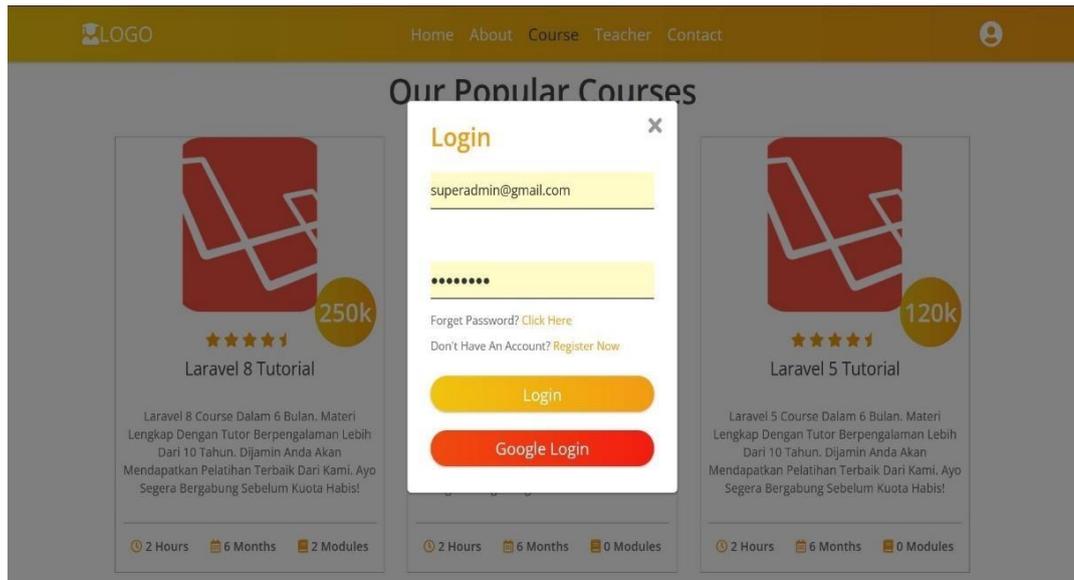
1. Masuk pada halaman awal sistem

Pada saat *student* memasuki sistem MOOC, maka sistem akan menampilkan halaman awal seperti pada Gambar 4. 1.



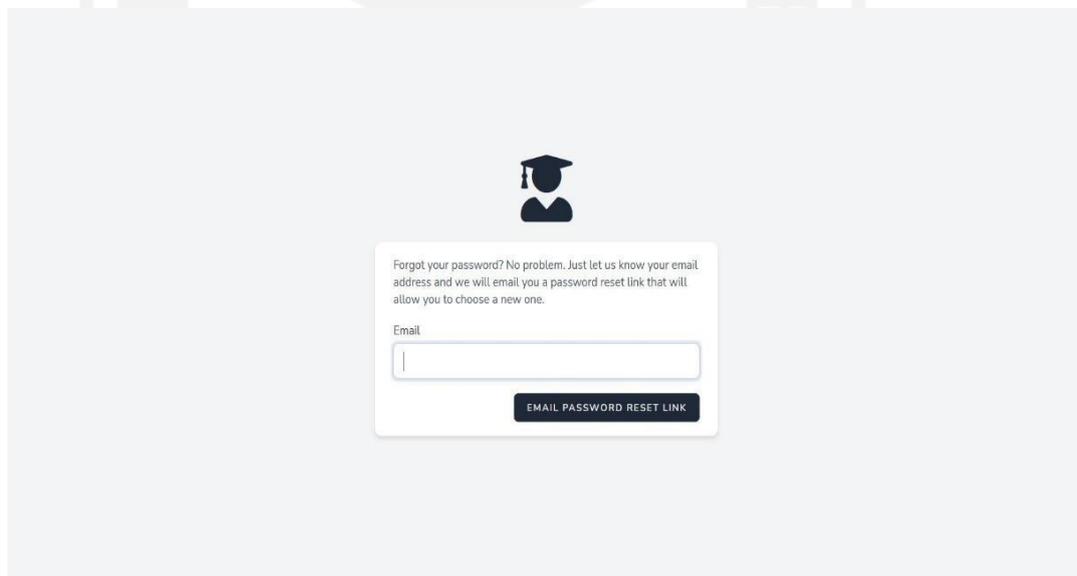
Gambar 4. 1 Halaman awal sistem

2. Selanjutnya *student* bebas untuk memilih, mau mencari kursus terlebih dahulu atau login pada sistem, akan tetapi jika *student* akan mengklik menu *enroll* pada salah satu kursus, maka sistem akan otomatis memberikan halaman pop up agar *student* mendaftar atau login pada sistem.



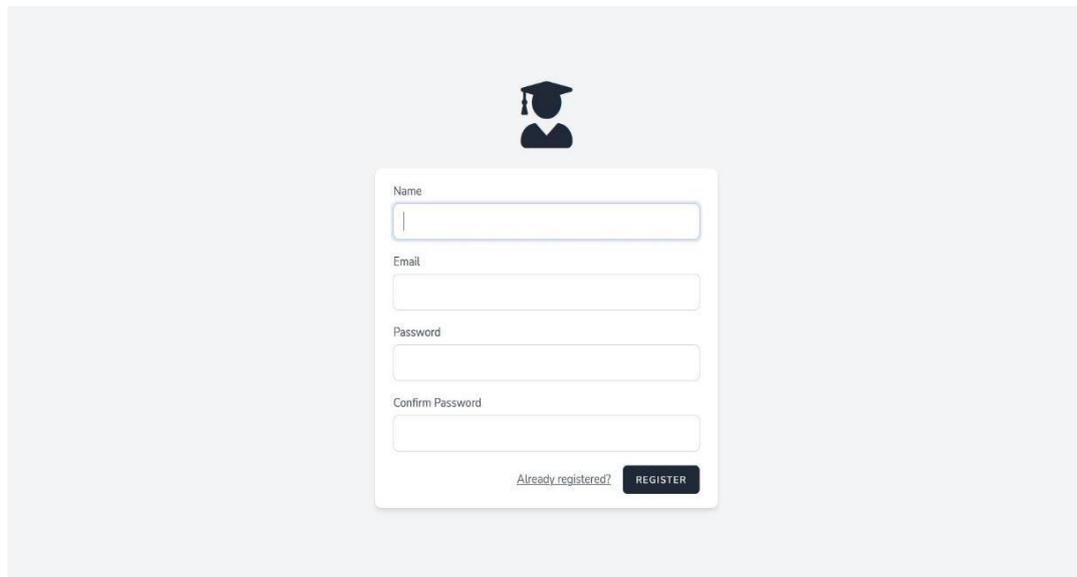
Gambar 4. 2 student dialihkan untuk login

3. Saat sistem menampilkan halaman *pop up login*, *student* dapat memilih beberapa menu. Seperti jika *student* sudah memiliki akun tapi lupa *password*, maka *student* hanya perlu mengklik tulisan “*cick here*” yang berada disamping kata *forget password*.



Gambar 4. 3 Student mengisi form reset *password*

4. *Student* juga dapat memilih login dengan cara mendaftar dengan menggunakan *email*. Adapun caranya yaitu *student* hanya perlu mengklik tulisan “*register now*” yang ada diatas tombol *login*, lalu sistem akan menampilkan halaman login dengan daftar *email*.



Gambar 4. 4 Sistem menampilkan halaman daftar email

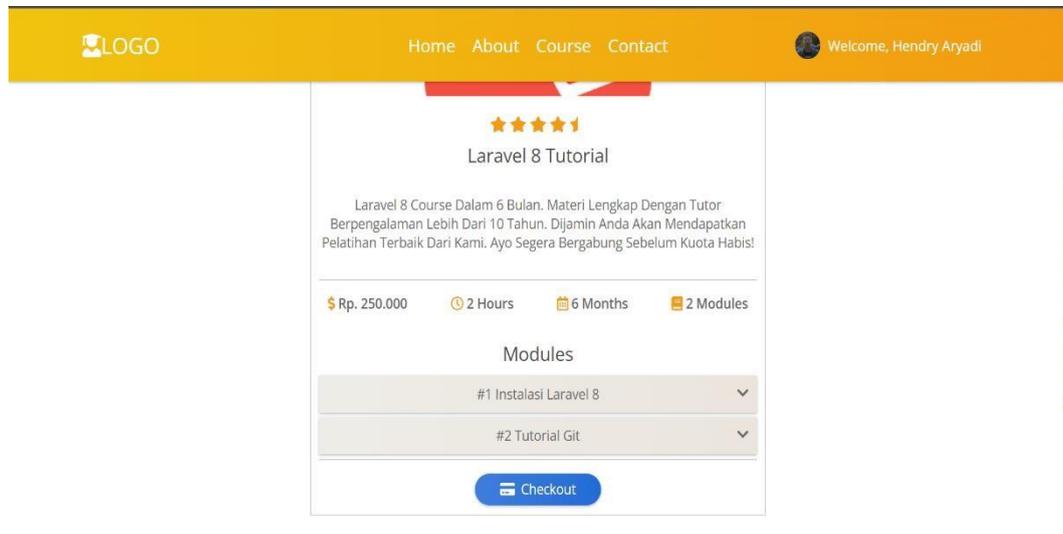
5. Jika *student* memiliki akun google, maka tidak perlu repot untuk mengisi form tersebut, *student* hanya perlu mengklik tombol google login, maka sistem akan menampilkan data *email* google yang sedang atau pernah login pada perangkat yang digunakan *student* tersebut.



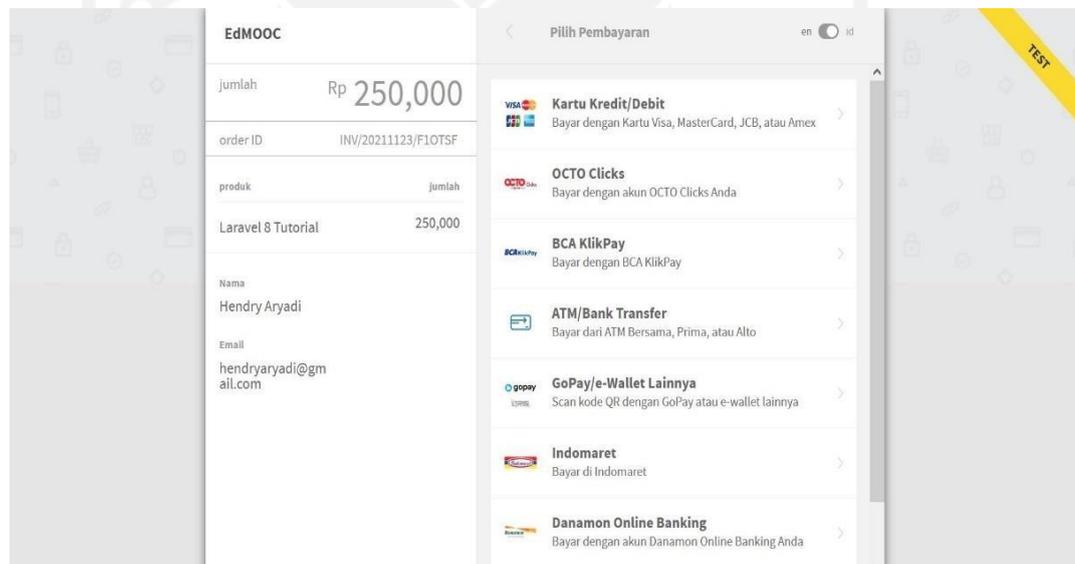
Gambar 4. 5 Halaman pilihan akun google

6. Setelah *student* berhasil *login*, *student* akan bebas mengikuti kursus, akan tetapi ada beberapa kursus yang berbayar, sehingga saat *student* ingin mendaftar pada kursus

tersebut, maka sistem akan menampilkan halaman pembayaran yang berisi detail dan jenis pembayaran yang tersedia pada sistem.

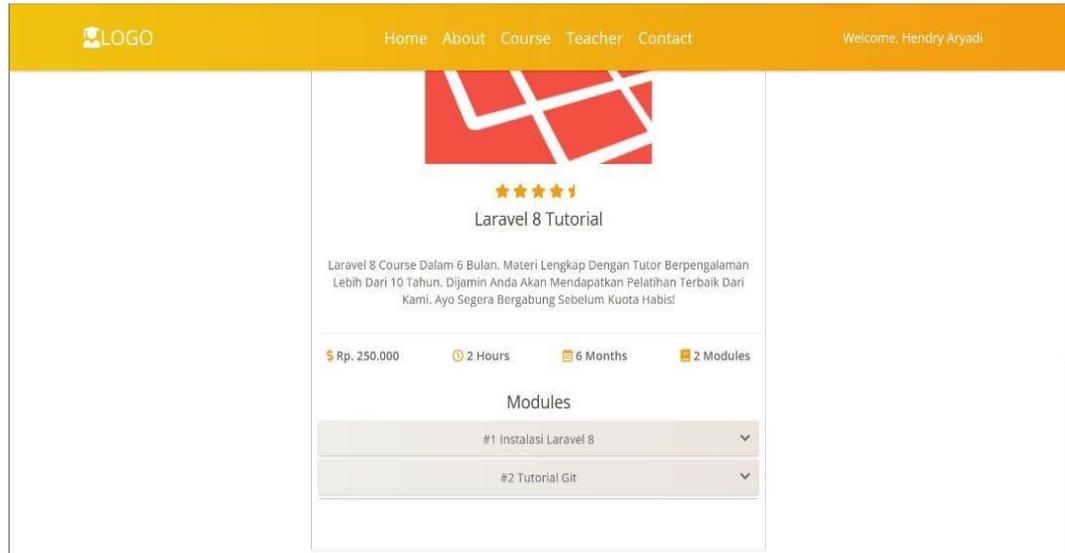


Gambar 4. 6 Halaman detail kursus berbayar



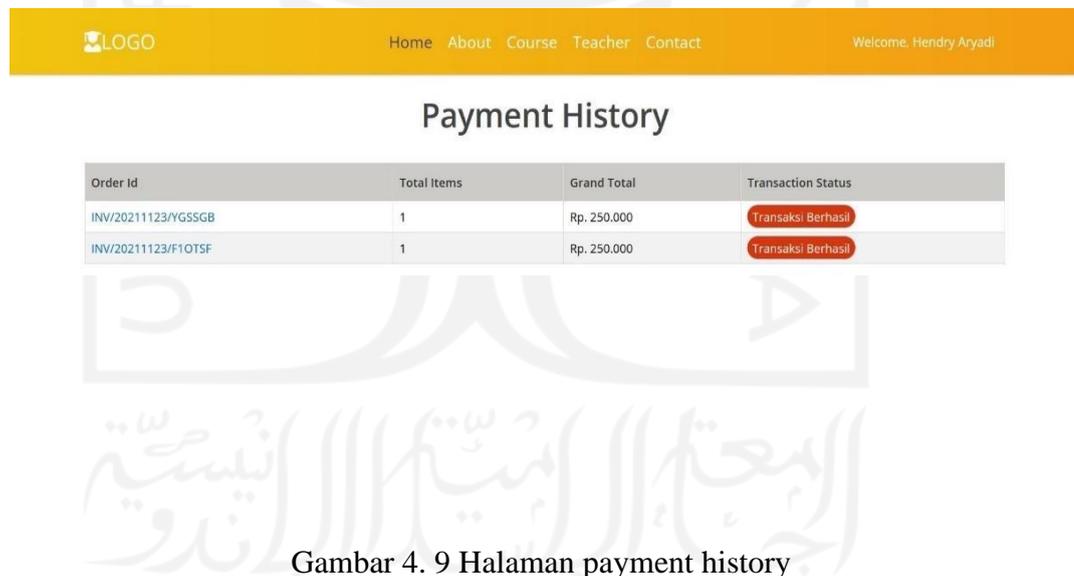
Gambar 4. 7 Halaman pembayaran kursus

7. Setelah *student* melakukan pembayaran pada kursus berbayar, maka modul yang sebelumnya terkunci akan terbuka dan tombol *cekout* akan hilang.



Gambar 4. 8 Halaman *course* berbayar setelah dilakukan pembayaran

8. *Student* juga dapat melihat riwayat pembayaran yang sudah dilakukan, jika sudah melakukan pembayaran maka akan muncul status berhasil seperti terlihat pada Gambar 4. 9



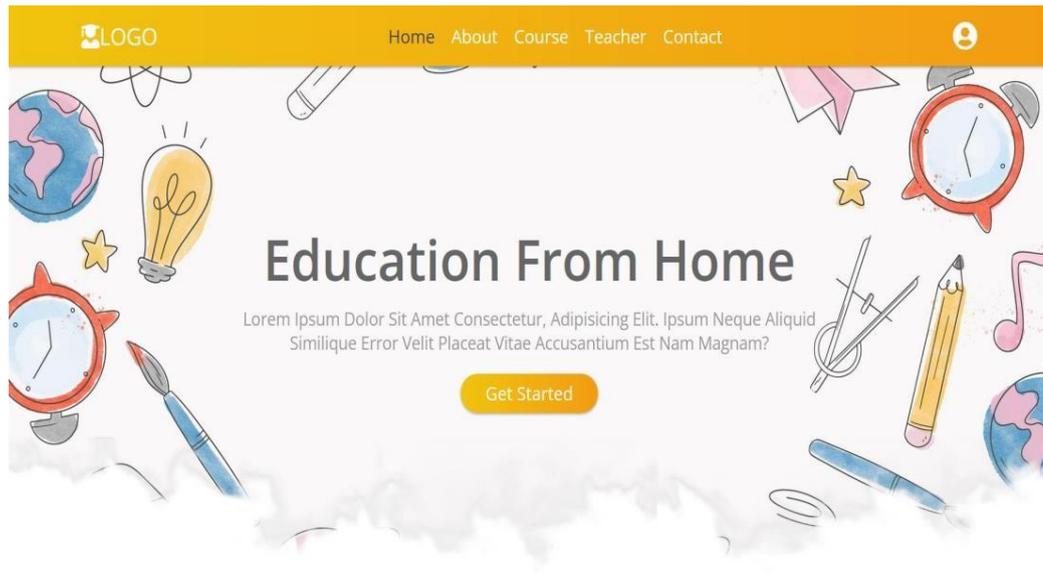
Gambar 4. 9 Halaman payment history

4.1.2 Skenario *user* institusi

Pada skenario *user* institusi digunakan untuk mengelola data yang ada pada sistem MOOC. Sebelum *user* institusi login, maka tampilan akan sama seperti *user* lainnya.

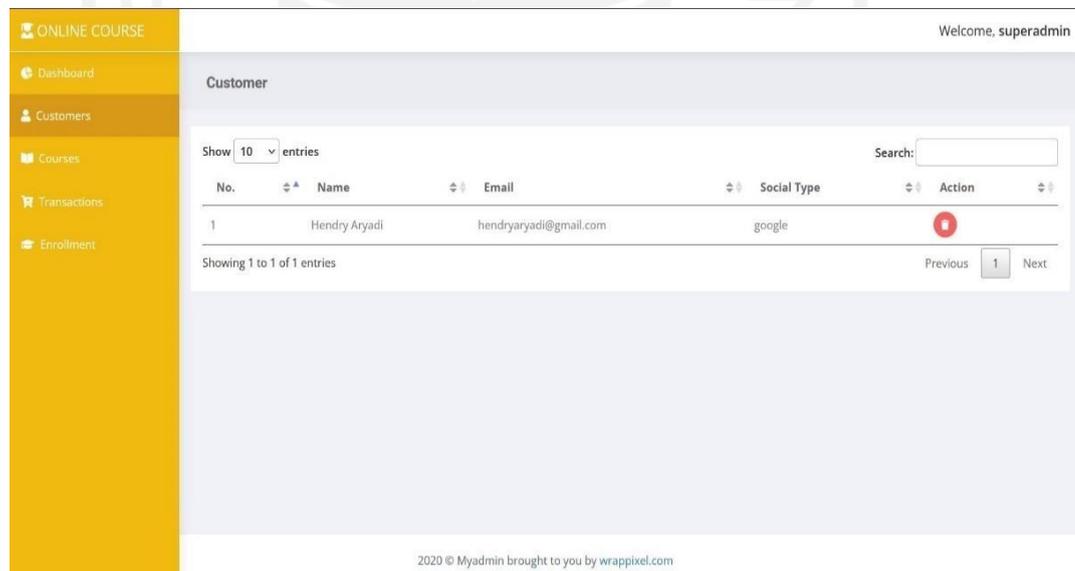
1. Halaman awal sistem

Pada saat *student* memasuki sistem MOOC, maka sistem akan menampilkan halaman awal seperti pada gambar berikut



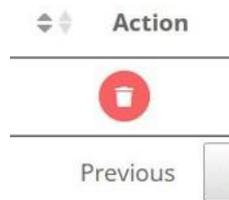
Gambar 4. 10 Halaman awal sistem (institusi)

2. Setelah *user* institusi login pada sistem, *user* dapat melihat data dari *user student* yang sudah mendaftarkan diri pada sistem, baik yang mendaftar menggunakan *email*, maupun langsung menggunakan *google*.



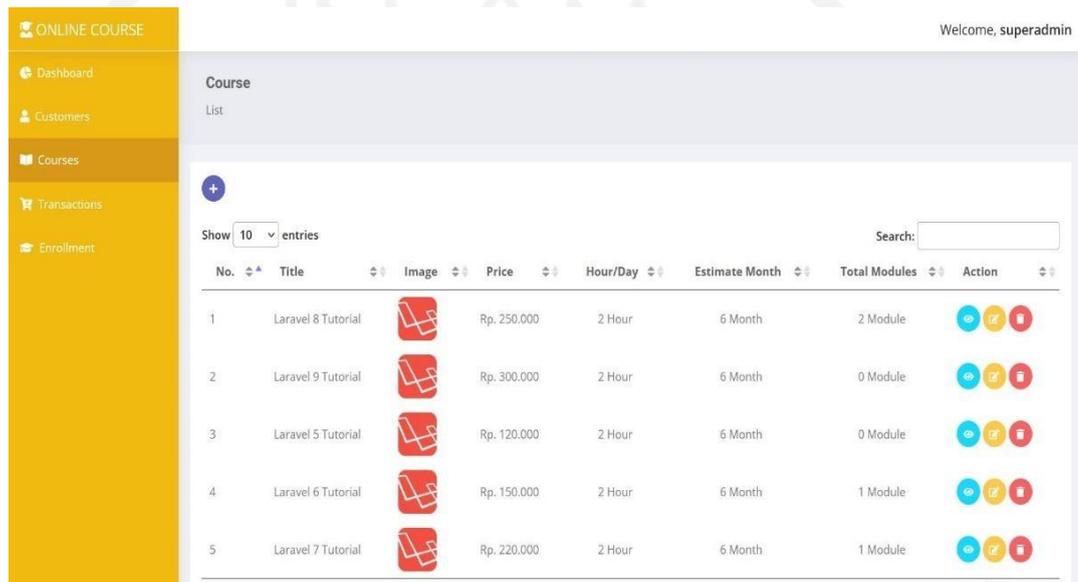
Gambar 4. 11 Institusi melihata data student

3. Pada halaman tersebut terdapat menu hapus akun *student*, akan tetapi fitur itu hanya boleh digunakan jika *user student* melanggar syarat dan ketentuan yang telah *student* setujui saat mendaftar pada sistem ini.



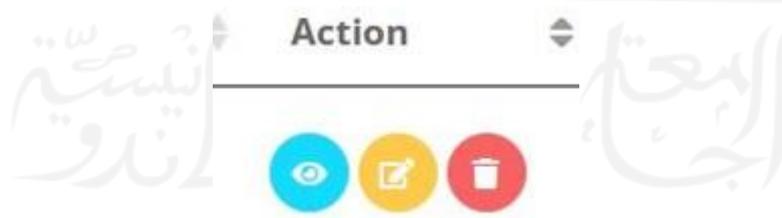
Gambar 4. 12 Tombol hapus akun student

4. Selanjutnya *user* institusi dapat melakukan mengelola data kursus pada menu kursus yang berada pada sebelah kiri



Gambar 4. 13 Insitusi mengelola *course*

5. Pada halaman tersebut juga terdapat beberapa menu seperti lihat detail kursus, edit kursus, dan hapus kursus.



Gambar 4. 14 Tombol lihat, edit, dan hapus kursus

6. Pada menu *transaction* terdapat data dari semua transaksi yang dilakukan oleh *student (customers)* baik yang gagal maupun yang sukses.

No.	Order ID	Customer Info	Grand Total	Status	Checkout Date	Checkout Time
1	INV/20211209/ZAQZZX	Hendry Aryadi hendryaryadi@gmail.com	Rp. 250.000	Transaksi Berhasil	2021-12-09	11:38:34
2	INV/20211209/HPB8T&W	Hendry Aryadi hendryaryadi@gmail.com	Rp. 250.000	Transaksi Batal	2021-12-09	13:20:25

Showing 1 to 2 of 2 entries

Gambar 4. 16 Halaman cek transaksi *student*

7. lalu selanjutnya pada menu *enrollment* berisi data dari *student* yang mendaftar pada suatu kursus, dimana nanti institusi dapat melihat dan menghapus data tersebut.

No.	Image	Title	Total Enrollment	Action
1		Laravel 8 Tutorial	1 students	
2		English Course Titik Nol	0 students	
3		Laravel 5 Tutorial	0 students	
4		Laravel 6 Tutorial	0 students	
5		Laravel 7 Tutorial	0 students	

Showing 1 to 5 of 5 entries

Gambar 4. 15 Menu *enrollment* (institusi)

4.2 Pembahasan

Hal yang akan dibahas disini yaitu pengujian dari implementasi dan komparasi sistem MOOC yang sudah dibuat.

4.2.1 Pengujian

Pada tahap ini penulis melakukan pengujian dengan beberapa tahapan, mulai dari pengujian awal dengan pengujian *alpha*, dilanjutkan dengan pengujian *beta*, dan yang terakhir yaitu pengujian *User acceptance test* (UAT).

4.2.1.1 Pengujian *alpha*

Pada tahap ini dilakukan pengujian oleh internal tim, sebelum diujikan ke pengguna akhir. Adapun hasil dari pengujian ini adalah sebagai berikut.

Tabel 4. 1 pengujian *alpha*

No	Item Pengujian	Keterangan
1	<i>Student</i> dapat membuka halaman login	Berhasil
2	<i>Student</i> dapat memilih login menggunakan email	Berhasil
3	<i>Student</i> dapat memilih login menggunakan akun google	Berhasil
4	<i>Student</i> dapat mereset <i>password</i> dengan email	Berhasil
5	<i>Student</i> dapat merest <i>password</i> di menu profil	Berhasil
6	<i>Student</i> dapat membuka halaman profil	Berhasil
7	<i>Student</i> dapat mengedit profil	Berhasil
8	<i>Student</i> dapat mengecek transaksi	Berhasil
9	<i>Student</i> dapat melihat daftar kursus yang di ikuti	Berhasil
10	<i>Student</i> dapat melihat halaman detail kursus	Berhasil
11	<i>Student</i> dapat mencari kursus pada <i>explore course</i>	Berhasil
12	<i>Student</i> dapat memilih metode pembayaran	Berhasil
13	Institusi dapat membuka halaman login	Berhasil
14	Institusi dapat membuka halaman <i>dashboard</i>	Berhasil
15	Institusi dapat membuka halaman tambah kursus	Berhasil
16	Institusi dapat menambah, mengedit dan menghapus kursus	Berhasil
17	Institusi dapat melihat halaman daftar kursus	Berhasil
18	Institusi dapat melihat detail kursus	Berhasil
19	Institusi dapat melihat halaman daftar <i>enrollment</i>	Berhasil
20	Institusi dapat melihat halaman daftar <i>student</i>	Berhasil
21	Institusi dapat melihat daftar pembayaran	Berhasil
22	Institusi dapat melihat halaman permbayaran tiap <i>student</i>	Berhasil

4.2.1.2 Pengujian *beta*

Setelah selesai membuat prototype, selanjutnya penulis menguji prototype kepada pengguna akhir. Adapun sampel penguji diambil secara acak yang terdiri dari tenaga pendidik yang sebelumnya telah mengisi kuesioner pada tahap pengumpulan data. Adapun hasil dari pengujian ini dapat dilihat pada tabel 4.2 sebagai berikut.

Tabel 4. 2 Pengujian *beta*

Responden	Hasil Evaluasi	Saran
<i>Proses enrollment dan login</i>		
R1	Proses <i>enrollment</i> dan <i>login</i> berjalan dengan baik. alur, proses dan fitur yang digunakan juga sudah sesuai dengan kebutuhan awal.	-

R2	Penerapan fitur dan alur proses saat <i>login</i> sudah sesuai dan tidak ditemukan bug maupun <i>error</i> .	-
<i>Proses payment</i>		
R1	Proses <i>payment</i> berjalan dengan baik. Alur, proses dan fitur yang digunakan juga sudah sesuai dengan kebutuhan awal.	-
R2	Penerapan fitur dan alur proses <i>payment</i> sudah sesuai dan tidak ditemukan bug maupun <i>error</i> .	-
<i>Manajemen data student</i>		
R1	Pengelolaan data <i>student</i> berjalan dengan baik. walaupun data yang disimpan sedikit, tapi sudah cukup untuk memenuhi kebutuhan minimal yang dibutuhkan.	-
R2	Dalam mengelola data <i>student</i> cukup simpel dan mudah dipahami. Pada saat menjalankan proses manajemen juga tidak ditemukan bug maupun <i>error</i> .	-
<i>Manajemen data enrollment dan payment</i>		
R1	Pengelolaan data <i>enrollment</i> dan <i>payment</i> cukup sederhana dan berjalan dengan baik.	-
R2	Dalam mengelola data <i>enrollment</i> dan <i>payment</i> cukup mudah dipahami dan tidak ditemukan bug maupun <i>error</i> .	-

Dari hasil pengujian tersebut penulis tidak mendapat masukan dan laporan adanya error maupun bug pada sistem MOOC, sehingga penulis melanjutkan pada proses pengujian selanjutnya.

4.2.1.3 Pengujian *User Acceptance Test* (UAT)

Setelah melakukan pengujian alpha pada internal tim, lalu penulis melakukan pengujian beta kepada pengguna akhir. Setelah melewati pengujian tersebut selanjutnya penulis melakukan pengujian *user acceptance test* (UAT). UAT sendiri merupakan salah satu jenis pengujian yang berada pada sudut pandang pengguna akhir untuk mengetahui apakah sistem yang dibuat sudah sesuai dengan kebutuhan awal atau belum. Pada pengujian metode UAT ini, pengguna akhir akan diuji melalui instrumen – instrumen yang sudah ditentukan sebelumnya.

Penentuan pengguna akhir yang akan dijadikan responden dipilih seperti pada pengujian sebelumnya, yaitu pada responden awal saat pengumpulan data. Adapun dalam menentukan jumlah responden, penulis menggunakan metode sampel incidental. Dari metode tersebut, penulis mendapatkan jumlah responden sebanyak 25 orang yang terdiri dari 20 *student* dan 5 tenaga pendidik.

1. Instrumen pengukuran

Pada instrumen pengukuran, penulis ingin mengetahui apakah sistem yang dibuat sudah menjawab rumusan masalah atau belum. Adapun proses penilaian yang digunakan yaitu menggunakan skala *likert* untuk mengukur sikap, pendapat dan persepsi dari pengguna akhir

1 = Sangat Tidak Setuju

2 = Tidak Setuju

3 = Ragu – ragu

4 = Setuju

5 = Sangat Setuju

a. User Student

pada tahap ini berisi data pertanyaan yang diujikan kepada responden, dalam kasus ini penulis memilih 20 responden, adapun datanya adalah sebagai berikut.

Tabel 4. 3 Pengujian UAT *Student*

Aspek	Pertanyaan	Rentang Penilaian				
		STS	TS	RR	S	SS
Usability	Apakah proses login pada sistem mudah digunakan dan dipahami?			5	13	2
	Apakah pilihan login menggunakan google memudahkan anda dalam login?			4	14	2
	Apakah dengan membuka halaman baru membuat anda merasa terganggu saat login?		4	4	4	8
	Apakah tampilan pop up login tidak mengganggu anda saat akan login?			5	12	3
	Apakah tampilan pop up login yang muncul saat anda akan melakukan <i>enrollment</i> kursus menarik anda untuk login?			13	6	1
	Apakah proses mendaftar kursus mudah digunakan dan dipahami?			5	12	3
	Apakah proses pembayaran mudah digunakan dan dipahami?			5	12	3

	Apakah metode pembayaran yang tersedia memudahkan anda dalam melakukan pembayaran?		6	10	4
	Apakah metode pembayaran yang tersedia tergolong mudah ditemukan disekitar anda?		3	13	4
	Apakah proses cek transaksi memudahkan anda dalam melakukan pengecekan transaksi?		6	13	1
Total		4	56	109	31

b. *User* institusi

pada tahap ini berisi data pertanyaan yang diujikan kepada responden, dalam kasus ini penulis memilih 5 responden, adapun datanya adalah sebagai berikut.

Tabel 4. 4 Pengujian UAT Institusi

Aspek	Pertanyaan	Rentang Penilaian				
		STS	TS	RR	S	SS
Usability	Apakah proses pengelolaan <i>course</i> mudah digunakan dan dipahami?			1	2	2
	Apakah proses pada menu <i>enrollment</i> mudah digunakan dan dipahami?			2	3	
	Apakah proses pada menu <i>student</i> mudah digunakan dan dipahami?			2	2	1
	Apakah proses pada menu <i>transaction</i> mudah digunakan dan dipahami?				4	1
	Apakah tampilan menu pada halaman <i>dashboard</i> dan keseluruhan sistem menarik?			3	2	
Total				8	13	4

4.2.1.4 Hasil Pengujian UAT

Dari pengujian UAT yang sudah dilakukan, penulis menjabarkannya dalam Tabel

4. 5.

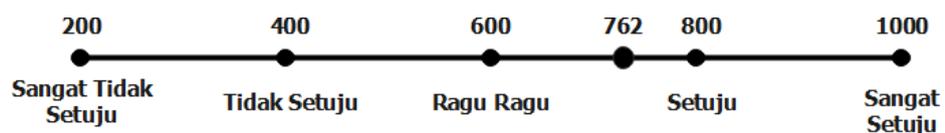
1. Hasil pengujian UAT *student*

Tabel 4. 5 Hasil pengujian UAT *student*

N	Pertanyaan										Jumlah
	1	2	3	4	5	6	7	8	9	10	
R_1	4	4	5	4	3	4	4	5	4	5	37
R_2	4	4	4	4	3	4	4	3	4	4	38
R_3	3	4	5	3	3	4	4	3	3	4	36
R_4	4	4	3	4	4	5	5	4	4	4	41
R_5	5	5	3	4	4	5	5	4	4	4	43
R_6	4	4	2	5	3	4	4	4	3	3	36
R_7	3	3	2	5	3	3	4	3	3	3	32
R_8	4	4	2	5	3	3	3	3	4	4	35
R_9	4	4	3	4	4	4	4	4	4	4	39
R_10	3	3	2	3	3	4	3	3	4	3	31
R_11	4	4	5	4	3	3	3	4	4	4	38
R_12	4	4	4	3	3	4	4	4	4	4	38
R_13	4	4	5	4	4	4	4	4	4	4	41
R_14	3	3	3	3	3	3	3	4	4	3	32
R_15	4	4	5	4	3	4	4	4	4	3	39
R_16	3	3	4	3	3	3	3	3	4	3	32
R_17	4	4	5	4	4	4	4	5	5	4	43
R_18	4	4	4	4	3	4	4	4	5	4	40
R_19	5	5	5	4	5	5	5	5	5	4	48
R_20	4	4	5	4	4	4	4	5	5	4	43
Total	73	78	76	78	68	78	78	78	81	75	762

Hasil dari pengujian yang dilakukan pada *user student*, didapatkan jumlah total sebanyak 762, jika nilai sempurna maka totalnya adalah $20 \times 50 = 1000$, karena total maksimum nilai pada tiap pertanyaan yaitu 5 dan jumlah responden 20 orang. Dari skor yang didapatkan hasil pengujiannya masuk dalam kategori antara ragu – ragu dan setuju. Dengan demikian usabilitas MOOC modul *enrollment* dan *payment* yaitu $772 : 1000 = 77.2\%$. untuk lebih jelasnya dapat dilihat pada skala berikut.

Nilai 762 termasuk dalam kategori interval ragu – ragu dan setuju, akan tetapi lebih condong mendekati setuju.

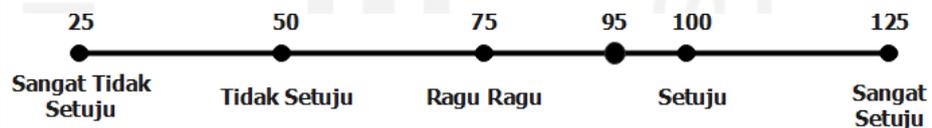


2. Hasil pengujian UAT institusi

Tabel 4. 6 Hasil pengujian UAT institusi

N	Pertanyaan					Jumlah
	1	2	3	4	5	
R_1	4	4	5	4	3	20
R_2	4	3	4	4	3	18
R_3	5	3	4	4	3	19
R_4	5	4	4	4	4	21
R_5	3	3	4	4	3	17
Total	21	17	21	20	16	95

Seperti pada tabel 4.5 diatas, jumlah skor yang didapatkan pada pengujian UAT role Institusi yaitu 95, sedangkan jumlah skor maksimal yaitu $5 \times 5 \times 5 = 125$. Data tersebut berasal dari maksimal nilai pertanyaan yaitu 5, total pertanyaan 5 dan responden berjumlah 5 orang. Dari skor tersebut didapat hasil pengujian masuk dalam kategori antara ragu – ragu dan setuju. Dengan demikian usabilitas MOOC modul *enrollment* dan *payment* yaitu $95 : 125 = 76\%$. Untuk lebih jelasnya dapat dilihat pada skala berikut.



Nilai 95 termasuk dalam kategori interval ragu – ragu dan setuju. Tetapi lebih condong mendekati setuju.

4.2.2 Komparasi sistem

Komparasi dilakukan dengan membandingkan MOOC yang dibuat berdasar *platform* referensi. Komparasi ini dilakukan untuk mengetahui perbedaan yang terdapat pada MOOC yang dibuat dengan *platform* MOOC referensi.

Tabel 4. 7 Tabel Komparasi

Nama Proses	EdMOOC	Platform Refrensi
Mengelola Proses Enrollment (<i>Student</i>)	Proses Enrollment mirip dengan referensi MOOC yang sudah ada, adapun perbedaannya yaitu pada saat akan login, dimana pada EdMOOC, akan menampilkan halaman pop up login maupun register.	Pada MOOC yang menjadi referensi, beberapa diantaranya seperti edx, futurelearn, udemy, udacity dan indonesiaiax, akan menampilkan halaman login biasa, tidak menggunakan halaman pop up. Adapun yang menggunakan pop up yaitu <i>coursera</i> .
Mengelola Proses Pembayaran (Institusi)	Proses payment yang ada pada EdMOOC yaitu menggunakan payment gateway midtrans.	Pada MOOC referensi menggunakan payment gateway yang berbeda beda, seperti pada <i>coursera</i> menggunakan payment gateway authorize.net dan razorpay. Edx menggunakan payment gateway cybersourcevisa, yang merupakan bagian dari authorize.net. Futurelearn menggunakan payment gateway stripe. Indonesiaiax menggunakan payment gateway midtrans

Mengelola Proses Pembayaran (<i>Student</i>)	Proses payment yang ada pada EdMOOC yaitu transfer bank, kartu kredit, alfamart dan indomart	Pada MOOC referensi lebih berfokus pada kartu kredit dan paypal
Mengelola Course (Institusi)	Pada mengelola kursus, sama seperti MOOC referensi, hanya sedikit berbeda pada tampilan dan data yang ditampilkan	Pada beberapa MOOC ada beberapa kursus yang mengharuskan <i>student</i> untuk mengisi form agar dapat masuk ke program kelasnya dan ada juga beberapa kursus yang saat mendaftar akan dialihkan ke halaman untuk memilih ingin mendaftar dengan gratis atau berbayar seperti pada Futurelearn.
Melihat data enrollment (Institusi)	Pada EdMOOC data yang ditampung oleh <i>student</i> hanya sedikit sehingga saat <i>student</i> mendaftar tidak perlu memasukan data terlalu banyak. Saat enroll juga sistem menyediakan login dengan 2 cara yaitu daftar email dan google	Pada MOOC referensi data yang ditampung agak banyak, akan tetapi <i>platform</i> MOOC referensi ini banyak yang menyediakan login dengan google, facebook, linkedin dan juga microsoft.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil yang di dapat dalam mengembangkan *massive open online course* (MOOC) modul *enrolmmment* dan *payment* dalam menambah minat peserta dan mempermudah akses pembayaran, adalah sebagai berikut:

1. Fitur login dengan menggunakan bantuan pihak ketiga seperti google dapat menambah minat agar orang mau mendaftar, hal tersebut dibuktikan dengan hasil survei dan pengujian sistem yang menghasilkan kriteria hampir mencapai memuaskan.
2. Menggunakan metode pembayaran berupa kartu kredit, transfer atm, dan indomart dapat memudahkan pengguna dalam melakukan pembayaran, hal ini dibuktikan dengan hasil survei pengujian sistem yang menghasilkan kriteria hampir mencapai memuaskan.

5.2 Saran

Berdasarkan hasil yang sudah didapatkan, penulis menyadari bahwa sistem yang dikembangkan masih jauh dari kata sempurna, adapun saran untuk pengembangan sistem MOOC modul *enrollment* dan *payment* adalah sebagai berikut.

1. Memperbanyak metode login yang sedang banyak digunakan pengguna sehingga *student* akan lebih tertarik untuk mendaftar.
2. Memperbanyak metode pembayaran yang mudah ditemukan di sekitar dan untuk memperbesar lingkup MOOC dapat dengan menambahkan pembayaran universal seperti paypal.

DAFTAR PUSTAKA

- Gené, O. B., Núñez, M. M., & Blanco, Á. F. (2014). Gamification in MOOC: Challenges, Opportunities and Proposals for Advancing MOOC Model. *Proceedings of the 2nd International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM)*. <https://doi.org/10.1145/2669711.2669902>
- Hatane Semuel. (2006). Dampak Respon Emosi Terhadap Kecenderungan Perilaku Pembelian Impulsif Konsumen Online Dengan Sumberdaya Yang Dikeluarkan Dan Orientasi Belanja Sebagai Variabel Mediasi. *Jurnal Manajemen Dan Kewirausahaan*. <https://doi.org/10.1007/s11071-009-9550-8>
- Jordan, K. (2014). Initial trends in enrolment and completion of massive open online courses. *International Review of Research in Open and Distance Learning*. <https://doi.org/10.19173/irrodl.v15i1.1651>
- Ng, A., Stavens, D., Thrun, S., Sokolsky, M., Intelligence, A., Agarwal, A., Mckeown, N., & Levis, P. (2012). *The Year of the MOOC NY Times*. 1–7. <https://doi.org/10.1038/nmeth1154>
- Pressman, R. S. (2009). Software engineering A practitioner's approach. In *McGraw-Hill Higher Education*. <https://doi.org/10.1017/CBO9781107415324.004>
- Sitorus, S. R. (2006). *ANALISIS PENGARUH PENGGUNAAN KARTU PEMBAYARAN ELEKTRONIK DAN DAYA SUBSTITUSI TRANSAKSI NON TUNAI ELEKTRONIK TERHADAP TRANSAKSI TUNAI INDONESIA*.
- Suliyanto. (2017). *Pelatihan Metode Penelitian*. 1–39.
- Talbert, P. Y. (2012). Strategies to increase enrollment, retention, and graduation rates. *Journal of Developmental Education*. <https://doi.org/10.1007/s10526-008-9169-6>
- Pressman, R. S. (2012). *Rekayasa Perangkat Lunak: Pendekatan Praktisi Edisi 7*.
- Chan, Kah Sing. (2011). *Electronic Commerce and Supply Chain Management*. 2nd edition. Thomas Learning Asia, Singapore
- Kop, R. & Carroll, F. (2011). Cloud computing and creativity: Learning on a massive open online course. Retrieved from <http://www.eurodl.org/?p=special&sp=articles&article=457>

- Anderson, E.W., Potter, K.C., Matzen, L.E., Shepherd, J.F., Preston, G.A., & Silva, C.T. (2011). A *user* study of visualization effectiveness using EEG and cognitive load. *Computer Graphics Forum*, 30(3), 791-800.
- Weller, M., Siemens, G., & Cormier, D. (2012). MOOCS: An interview with Dave Cormier and George Siemens [Video file]. Retrieved from http://www.youtube.com/watch?list=UUyxGjHJz2S-b5DHDVpkMu-w&v=11G4SUblnbo&feature=player_embedded
- Mackness, J., Mak, S. & Williams, R. (2010). The Ideals and Reality of Participating in a MOOC. *Proceedings of the 7th International Conference on Networked Learning*, 266-274, Aalborg, Denmark, May 2010.



LAMPIRAN

