

SKRIPSI
RANCANG BANGUN *E-COURSE* BERBASIS *WEBSITE*
MENGGUNAKAN METODE RAD DENGAN LARAVEL 10



OLEH:

Saeful mu'minin

2026027

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS NAHDLATUL ULAMA INDONESIA
JAKARTA
2024

LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : Saeful Mu'minin
NIM : 2026027
Program Studi : Teknik Informatika
Fakultas : Teknik dan Ilmu Komputer
Jenjang Pendidikan : Strata 1

Menyatakan bahwa skripsi yang saya buat dengan judul:

RANCANG BANGUN E-COURSE BERBASIS WEBSITE MENGGUNAKAN METODE
RAD DENGAN LARAVEL 10

1. Merupakan hasil karya tulis ilmiah sendiri, bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik oleh pihak lain, dan bukan merupakan hasil plagiat.
2. Saya izinkan untuk dikelola oleh UNIVERSITAS NAHDLATUL ULAMA INDONESIA sesuai dengan norma hukum dan etika yang berlaku.

Pernyataan ini saya buat dengan penuh tanggung jawab dan saya bersedia menerima konsekuensi apapun sesuai aturan yang berlaku apabila di Kemudian hari pernyataan ini tidak benar.

Jakarta, 16 April 2024



LEMBAR PENGESAHAN

LEMBAR PENGESAHAN

Nama : Saeful Mu'minin
NIM : 2026027
Program Studi : Teknik Informatika
Fakultas : Teknik dan Ilmu Komputer
Jenjang Pendidikan : Strata 1
Judul Skripsi : RANCANG BANGUN E-COURSE
BERBASIS WEBSITE MENGGUNAKAN
METODE RAD DENGAN LARAVEL 10

Skripsi ini telah dipertahankan di hadapan dewan penguji ujian skripsi Fakultas Teknik dan Ilmu Komputer, Program Studi Teknik Informatika dan dinyatakan LULUS.

Jakarta, 27 April 2024
Menyetujui,

Dosen Penguji I



Ircham Ali, S.Kom., M.Kom.
NIDN. 0320119401

Dosen Penguji II



Handy Fernandy, S.T., M.MSI.
NIDN. 0309029202

Dosen Pembimbing



Mohammad Reza Fahlevi, S.T., M.M.

NIDN. 0307089104

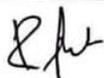
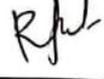
Mengetahui,
Ketua Program Studi
Teknik Informatika Unusia



Ircham Ali, S.Kom., M.Kom.
NIDN. 0320119401

LEMBAR BIMBINGAN

Nama : Saeful Mu'minin
Judul Skripsi : RANCANG BANGUN E-COURSE BERBASIS WEBSITE
MENGGUNAKAN METODE RAD DENGAN LARAVEL 10
Pembimbing : Mohammad Reza Fahlevi, S.T., M.M

No	Hari/Tanggal	Perbaikan	Paraf Pembimbing
1	15/05/2023	Menemukan judul Skripsi dan studi kasus	
2	11/06/2023	Penulisan Bab 1 dan revisi penulisan latar belakang dan tujuan penelitian	
3	28/07/2023	Penulisan Bab 2 dan revisi bagian penelitian terdahulu dan landasan teori	
4	05/08/2023	Penulisan Bab 3 dan revisi kerangka berpikir	
5	1/09/2023	ACC proposal skripsi	
6	20/09/2023	Revisi proposal skripsi	
7	25/09/2023	Pengesahan proposal skripsi	
8	22/03/2024	Penulisan Bab 4 dan Bab 5, serta revisi pengujian blackbox dan website	
9	28/03/2024	Revisi Bab 4, penulisan daftar Pustaka dan Abstrak	
10	1/04/2024	Review Bab 4 dan Bab 5, review penulisan secara keseluruhan	
11	5/04/2024	ACC Sidang Skripsi	

Pembimbing,



Mohammad Reza Fahlevi, S.T., M.M
NIDN. 0307089104

KATA PENGANTAR

Puji syukur Alhamdulillah, teriring ke hadirat Allah SWT yang Maha Pengasih dan Maha Penyayang, yang telah melimpahkan berbagai rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan laporan skripsi ini sebagai salah satu syarat untuk menyelesaikan program studi strata satu (S1) pada program studi Teknik Informatika di Universitas Nahdlatul Ulama Indonesia.

Penulis menyadari bahwa karya ini masih jauh dari kesempurnaan. Oleh karena itu, segala kritik dan saran yang membangun sangat penulis harapkan. Dengan rendah hati, penulis juga menyadari bahwa laporan skripsi ini tidak akan terwujud tanpa bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, dengan ikhlas dan tulus, penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak H. Juri Ardiantoro, M.Si., Ph.D., selaku Rektor di Universitas Nahdlatul Ulama Indonesia.
2. Ibu Adrinoviarini, M.Sc., selaku Dekan Fakultas Teknik dan Ilmu Komputer di Universitas Nahdlatul Ulama Indonesia.
3. Bapak Ircham Ali, S.Kom., M.Kom., selaku ketua program studi Teknik Informatika di Universitas Nahdlatul Ulama Indonesia.
4. Mohammad Reza Fahlevi, S.T., M.M selaku pembimbing skripsi pada program studi Teknik Informatika di Universitas Nahdlatul Ulama Indonesia.
5. Ibu kandung selaku ibu kandung yang telah mendukung penulis baik motivasi,moral maupun materi.

6. Saudara dan sahabat-sahabatku, terutama kawan-kawan angkatan 2020 yang telah memberikan dukungan moral untuk terus menyelesaikan skripsi ini.
7. Saudara dan sahabat-sahabatku, sodara yang di bogor yang telah memberikan dukungan materi dan moral untuk terus menyelesaikan skripsi ini.

Semoga segala amal baik yang telah diberikan oleh semua pihak yang disebutkan di atas mendapatkan balasan yang setimpal dari Allah SWT. Semoga juga laporan skripsi ini dapat memberikan manfaat dan kontribusi positif bagi pengembangan ilmu pengetahuan. Amiin.

Jakarta, 25 Maret 2024



penulis

ABSTRAK

Ditengah kemajuan teknologi saat ini telah mengubah pandangan tentang konsep pembelajaran diseluruh institusi pendidikan secara signifikan tidak hanya di Indonesia, bahkan dunia. Hal ini dialami pula oleh Universitas Nahdlatul Ulama Indonesia (UNUSIA). Kegiatan belajar yang selama ini dilakukan dengan metode hybrid, yang menggabungkan komponen *online* dan *offline* harus dilakukan penambahan konsep pembelajaran yakni dengan menggunakan pembelajaran media elektronik atau *e-learning* Tujuannya adalah untuk membantu mahasiswa dalam memperdalam pemahaman mereka terhadap materi pembelajaran tanpa terikat oleh batasan waktu, kurikulum, atau kendala geografis. *E-learning* atau pembelajaran elektronik dengan sistem *e-course* menjadi solusi penting dalam meningkatkan kualitas dan pengalaman belajar. Sistem *e-course* untuk menyampaikan materi pembelajaran yang lebih spesifik dan terfokus pada suatu bidang atau keterampilan tertentu. Oleh karena itu, dibangunlah sebuah sistem *e-course* menggunakan framework Laravel dengan pendekatan pengembangan sistem menggunakan *Rapid Application Development* (RAD). Metode RAD mempercepat proses perancangan sistem *e-course*. Penelitian ini menghasilkan sistem *e-course* yang memungkinkan akses fleksibel terhadap materi pembelajaran, pengaturan waktu belajar yang sesuai kebutuhan, serta mengatasi hambatan geografis dan keterbatasan waktu melalui sebuah platform *website*.

Kata Kunci: *e-learning*, *e-course*, Laravel, RAD , pembelajaran

ABSTRACT

In the midst of current technological advances, the concept of learning in all educational institutions has significantly changed, not only in Indonesia, but also in the world. This is also experienced by Universitas Nahdlatul Ulama Indonesia (UNUSIA). Learning activities that have been carried out with a hybrid method, which combines online and offline components must be added to the learning concept by using electronic media learning or e-learning. The aim is to assist students in deepening their understanding of learning materials without being bound by time, curriculum, or geographical constraints. E-learning or electronic learning with e-course system becomes an important solution in improving the quality and learning experience. E-course system to deliver learning materials that are more specific and focused on a particular field or skill. Therefore, an e-course system is built using Laravel framework with Rapid Application Development (RAD) approach. RAD method accelerates the e-course system design process. This research resulted in an e-course system that allows flexible access to learning materials, customised learning time, and overcomes geographical and time constraints through a website platform.

Keywords: e-learning, e-course, Laravel, RAD, learning

DAFTAR ISI

LEMBAR PERYATAAN	<i>i</i>
<u>S</u>LEMBAR PENGESAHAN	<i>ii</i>
LEMBAR BIMBINGAN	<i>iii</i>
KATA PENGANTAR	<i>iv</i>
ABSTRAK	<i>vi</i>
ABSTRACT	<i>vii</i>
DAFTAR ISI	<i>viii</i>
DAFTAR GAMBAR	<i>xi</i>
DAFTAR TABEL	<i>xiii</i>
BAB I PENDAHULUAN	<i>1</i>
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian	3
1.5. Manfaat Penelitian	4
1.6. Metodologi Penelitian.....	4
BAB II LANDASAN TEORI	<i>7</i>
2.1. Penelitian Terdahulu	7
2.2. Landasan Teori.....	10

2.2.1. Rancang Bangun Sistem.....	10
2.2.2. <i>E-learning</i>	10
2.2.3. <i>E-course</i>	11
2.2.4. <i>Website</i>	12
2.2.5. RAD (<i>Rapid Application Development</i>).....	12
2.2.6. <i>Unified Modelling Language (UML)</i>	14
2.2.7. <i>Laravel</i>	18
2.2.8. <i>MVC (Model-View-Controller)</i>	19
2.2.9. <i>MySQL</i>	20
2.2.10. <i>Testing</i>	20
2.2.11. <i>System Usability Scale (SUS)</i>	21
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	22
3.1 Analisis Sistem Berjalan	22
3.1.1. Analisis Proses Bisnis.....	22
3.1.2. Kebutuhan Fungsional	23
3.1.3. Kebutuhan <i>Non Fungsional</i>	24
3.2 Desain Sistem Yang Diusulkan.....	24
3.2.1 Pemodelan <i>Use Case</i>	24
3.2.2 Pemodelan <i>Activity Diagram</i>	30
3.2.3 <i>Sequence Diagram</i>	35
3.2.4 <i>Class Diagram</i>	40

3.2.5	<i>Desain Basis Data</i>	40
3.2.6	Perancangan Antar muka.....	42
BAB IV IMPLEMENTASI DAN PENGUJIAN		47
4.1	Hasil penelitian	47
4.1.1	Tampilan <i>Login</i>	47
4.1.2	Tampilan <i>Register</i>	48
4.1.3	Tampilan Utama.....	48
4.1.4	Tampilan Mengelola Kelas.....	49
4.1.5	Tampilan Pembayaran	49
4.1.6	Tampilan Konfirmasi pembayaran.....	50
4.1.7	Tampilan Akses kelas	50
4.1.8	Tampilan Mengelola <i>User</i>	51
4.1.9	Tampilan Ujian kelas	52
4.1.10	Tampilan Sertifikat	52
4.2	Pengujian.....	53
4.3	Hasil <i>System Usability Scale</i>	54
BAB V PENUTUP		57
5.1	Kesimpulan.....	57
5.2	Saran	58
DAFTAR PUSTAKA		59
LAMPIRAN		63

DAFTAR GAMBAR

Gambar 2. 1 Siklus Pengembangan Rad.....	13
Gambar 2. 2 Konsep Mvc	19
Gambar 3. 1 Sistem Yang Berjalan	22
Gambar 3. 2 Sistem Yang Diusulkan	23
Gambar 3. 3 Pemodelan Use Case	24
Gambar 3. 4 Implementasi Activity Diagram Register.....	31
Gambar 3. 5 Implementasi Activity Diagram login	31
Gambar 3. 6 Implementasi Activity Diagram Kelola Kelas	32
Gambar 3. 7 Implementasi Activity Diagram Pembayaran Kelas	32
Gambar 3. 8 Activity Diagram Konfirmasi Pembayaran.....	33
Gambar 3. 9 Implementasi Activity Diagram Akses Kelas.....	33
Gambar 3. 10 Implementasi Activity Diagram Kelola Mahasiswa	34
Gambar 3. 11 Implementasi Activity Diagram Sertifikat	34
Gambar 3. 12 Implementasi Activity Diagram Logout	35
Gambar 3. 13 Implementasi Sequence Diagram Register	35
Gambar 3. 14 Implementasi Sequence Diagram Login	36
Gambar 3. 15 Implementasi Sequence Diagram Kelola Kelas	36
Gambar 3. 16 Implementasi Sequence Diagram Pembayaran Kelas.....	37
Gambar 3. 17 Implementasi Sequence Diagram Konfirmasi Pembayaran	37
Gambar 3. 18 Implementasi Sequence Diagram Akses Kelas	38
Gambar 3. 19 Implementasi Sequence Diagram Kelola Mahasiswa	38
Gambar 3. 20 Implementasi Sequence Diagram Sertifikat.....	39

Gambar 3. 21 Implementasi Sequence Diagram Logout.....	39
Gambar 3. 22 Implementasi Class Diagram Sistem E-Course.....	40
Gambar 3. 23 Perancangan Halaman Register.....	42
Gambar 3. 24 Perancangan Halaman Login.....	43
Gambar 3. 25 Perancangan Halaman Landing Page.....	43
Gambar 3. 26 Perancangan Halaman Dashboard.....	44
Gambar 3. 27 Perancangan Halaman Kelola Kelas.....	44
Gambar 3. 28 Perancangan Halaman Kelola Pembayaran.....	45
Gambar 3. 29 Perancangan Halaman Kelola Mahasiswa.....	45
Gambar 3. 30 Perancangan Halaman Learning.....	46
Gambar 4. 1 Tampilan Login.....	47
Gambar 4. 2 Tampilan Registrasi.....	48
Gambar 4. 3 Tampilan Home.....	48
Gambar 4. 4 Tampilan Kelola Kelas.....	49
Gambar 4. 5 Tampilan Pembayaran.....	49
Gambar 4. 6 Tampilan Konfirmasi Pembayaran.....	50
Gambar 4. 7 Tampilan Akses Kelas.....	51
Gambar 4. 8 Tampilan Kelola User.....	51
Gambar 4. 9 Tampilan Ujian.....	52
Gambar 4. 10 Tampilan Sertifikat.....	52

DAFTAR TABEL

Table 2. 1 Penelitian Sebelumnya	8
Table 2. 2 Use Case Diagram	14
Table 2. 3 Activity Diagram	16
Table 2. 4 Sequence Diagram	17
Table 2. 5 Class Diagram	18
Table 2. 6 Nilai System Usability Scale.....	21
Table 3. 1 Use Case Register	25
Table 3. 2 Use Case Login	25
Table 3. 3 Use Case Kelola Kelas	26
Table 3. 4 Pembayaran Kelas.....	27
Table 3. 5 Konfirmasi Pembayaran	28
Table 3. 6 Akses Kelas	28
Table 3. 7 Kelola Mahasiswa	29
Table 3. 8 Generate Sertifikat	29
Table 3. 9 logout.....	30
Table 3. 10 Basis Data User	41
Table 3. 11 Basis Data Admin	41
Table 3. 12 Basis Data Kelas	41
Table 3. 13 Basis Data Transaksi	42
Table 4. 1 Hasil Pengujian	53
Table 4. 2 Hasil Responden	54
Table 4. 3 Hasil Nilai System Usability Scale.....	55

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Bersamaan dengan perkembangan peradaban manusia, teknologi terus berkembang di berbagai aspek kehidupan salah satu yang mengalami peningkatan signifikan adalah teknologi dalam bidang pendidikan.(Indrawan dan Nugraha, 2020). Teknologi menjadi faktor kunci dalam meningkatkan mutu dan pengalaman belajar di dalam dunia pendidikan. Oleh sebab itu, diperlukan sistem pembelajaran yang dapat memungkinkan mempermudah akses pembelajaran di mana saja dan kapan saja sekaligus memungkinkan pemilihan kelas belajar sesuai dengan kebutuhan. Dalam mengembangkan dan membangun sistem *e-learning*.

E-learning merupakan kependekan dari *electronic learning* dan mengacu pada proses pembelajaran yang menggunakan media elektronik untuk penyampaian materi, diskusi, ujian dan kegiatan perkuliahan lainnya.(Riskiono dan Pasha, 2020). Universitas Nahdlatul Ulama Indonesia dapat memanfaatkan *e-course* untuk memberikan materi yang lebih spesifik dan fokus pada keterampilan yang ingin ditingkatkan oleh mahasiswa, menggunakan *e-course*, mahasiswa dapat memilih materi yang sesuai dengan kebutuhan mereka sendiri, Banyak universitas yang sudah menerapkan beberapa program untuk membantu mahasiswa dalam mendapatkan kemampuan yang mereka inginkan. Salah satunya Universitas Nahdlatul Ulama Indonesia (UNUSIA). Sebagai mahasiswa di UNUSIA, peneliti sangat antusias untuk menggunakan teknologi ini dan menjadikan universitas ini sebagai studi kasus dalam skripsi peneliti. UNUSIA merupakan perguruan tinggi yang terletak di Jakarta, dan saat ini terus meningkatkan pelayanan pendidikan melalui *platform E-learning* yang berbasis *e-campus*.

Sistem *e-course* yang dikembangkan menggunakan *framework laravel*, diharapkan dapat menjadi penting dalam dunia pendidikan dalam membantu meningkatkan pengalaman belajar dan membantu mahasiswa mendapatkan

kemampuan yang mereka inginkan, ada banyak sekali mahasiswa yang mendapatkan akses materinya terbatas dengan kurikulum yang ada sehingga mahasiswa tidak mendapatkan kemampuan yang mereka inginkan, maka diperlukan teknologi yang dapat memberikan pengalaman belajar yang lebih baik oleh semua mahasiswa, dan teknologi laravel merupakan salah satu solusinya.

Penggunaan teknologi laravel dalam pengembangan *website e-learning* atau *e-course* dengan pendekatan RAD (*Rapid Application Development*) dapat membantu mempercepat perancangan sebuah sistem *e-course*. Metode RAD adalah model dari *system development life cycle* (SDLC) yang mengedepankan pengembangan perangkat lunak dengan siklus yang singkat. (Hidayat dan Hati, 2021). Metode ini sangat membantu dalam mempersingkat kemungkinan terjadinya kesalahan di awal dalam pembuatan dan pengembangan sebuah sistem *website e-course* yang menggunakan *framework* laravel.

Penelitian sebelumnya dalam pembuatan sistem *e-learning* mengungkapkan beberapa kekurangan. Penelitian pertama oleh (Lesmana dan Silalahi, 2022), kurang fleksibel dalam metodenya dan memakan waktu lama karena minimnya keterlibatan pengguna dan terbatasnya teknologi. Penelitian kedua oleh (Priamnistiko dkk., 2021) menggunakan metode *waterfall* namun mengalami ketidakefektifan karena teknologi yang dipilih memerlukan migrasi dengan pihak ketiga. Penelitian ketiga oleh (Hardianty dkk., 2022), memerlukan waktu yang cukup lama dalam prosesnya dan masih terfokus pada desain yang akan ditampilkan.

Penelitian ini bertujuan merancang sistem *e-course* menggunakan metode RAD dengan Laravel 10. Sistem ini akan memiliki fitur-fitur seperti *login*, *registrasi*, transaksi, akses kelas, pengelolaan kelas, dan pembayaran dengan beragam metode. UNUSIA perlu membangun *e-course* untuk meningkatkan aksesibilitas pendidikan bagi mahasiswa melalui platform *online*. Dengan *e-course*, mahasiswa dapat belajar secara fleksibel, mandiri, dan mengatasi hambatan geografis serta waktu yang sering terjadi dalam pendidikan tradisional. *E-course* juga memungkinkan penyediaan beragam kelas dan program yang menunjang kebutuhan mahasiswa, yang efektif meningkatkan aksesibilitas pendidikan tinggi secara keseluruhan.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, berikut adalah rumusan masalah pada penelitian ini:

1. Bagaimana merancang *course online* dengan laravel 10 di UNUSIA?
2. Bagaimana meningkatkan kualitas pembelajaran dan pengalaman mahasiswa melalui *e-course*, termasuk pengembangan konten yang relevan, interaktif, dan mendukung pembelajaran kolaboratif serta penggunaan teknologi yang tepat?

1.3. Batasan Masalah

Untuk menghindari pembahasan yang panjang, maka perlu dibataskan permasalahan penelitian ini, antara lain:

1. Sistem *e-course* hanya di gunakan di UNUSIA
2. Penelitian ini hanya berfokus pada pengembangan sistem berbasis *web*.
3. *Website e-course* hanya untuk mahasiswa.
4. Materi yang akan ditampilkan adalah materi *non formal*.
5. Sistem ini dibangun menggunakan *framework* laravel 10.
6. Alat yang digunakan adalah VSCODE, XAMPP dan MYSQL sebagai database.
7. Metode RAD digunakan sebagai metode pengembangan pada penelitian ini.

1.4. Tujuan Penelitian

Adapun tujuan yang hendak dicapai dalam penelitian ini adalah sebagai berikut :

1. Merancang *e-course* berbasis web menggunakan laravel 10 di UNUSIA
2. Meningkatkan kualitas pembelajaran dan pengalaman mahasiswa melalui *e-course* dengan konten yang relevan, interaktif, mendukung pembelajaran kolaboratif, dan menggunakan teknologi yang tepat.

1.5. Manfaat Penelitian

Manfaat yang bisa didapat dari penelitian ini diharapkan mampu memberikan pengetahuan bagi semua pihak, antara lain:

1. Dosen
 - a. Sebagai pertimbangan nilai mahasiswa ketika sebuah kelas dapat diselesaikan.
 - b. Dapat juga menjadi pengajar didalam *course online*.
 - c. Memiliki kemampuan untuk mengetahui minat belajar dari mahasiswa.
2. Penulis
 - a. Memperdalam ilmu bahasa pemrograman PHP, JavaScript, dan CSS.
 - b. Dapat menambah wawasan dalam teknologi *web* yang akan dikembangkan.
 - c. Memperdalam ilmu bahasa pemrograman PHP, JavaScript, dan CSS.
3. UNUSIA
 - a. Rencananya akan memiliki 2 platform *E-learning*, yaitu *e-campus* dan *e-course*.
 - b. Sebagai bahan pertimbangan dan referensi karya tulis pada masa yang akan datang.

1.6. Metodologi Penelitian

Metodologi yang digunakan dalam rancang bangun aplikasi rancang bangun aplikasi presensi berdasarkan lokasi GPS berbasis android ini adalah:

1. Metode Pengumpulan Data

- a. Metode observasi

Metode observasi ini dilakukan melalui pengumpulan data, pengamatan dan pencatatan secara langsung maupun tidak langsung dengan pengelola fasilitas. Tujuannya adalah untuk menemukan informasi yang dapat digunakan sebagai bahan penelitian.

b. Kepustakaan

Pengumpulan data dilakukan langsung dari jurnal dan penelitian yang berkaitan dengan masalah tersebut.

2. Metode Rekayasa Perangkat Lunak

Metode pengembangan sistem dalam penelitian ini dibuat dengan menggunakan metode RAD (*Rapid Application Development*). Metode ini mencakup beberapa langkah, seperti pengumpulan permasalahan, desain sistem, proses pengembangan dan pengumpulan umpan balik, implementasi atau finalisasi produk.

a. Pengumpulan Permasalahan

Pengumpulan permasalahan mencari dan mengidentifikasi masalah yang akan peneliti teliti dan merupakan langkah awal dalam pembuatan sistem.

b. Desain Sistem

Desain sistem dilakukan dengan menggunakan notasi UML dan metode pemodelan objek, sehingga yang akan digunakan *use case*, *activity*, *sequence* dan *class diagram* sebagai diagram UML.

c. Pengembangan

Tahap ini merubah desain sistem yang akan menjadi beberapa code dan merubahnya menjadi sistem awal atau beta dan tahap ini peneliti sangat membutuhkan pengguna dalam pengembangannya.

d. Implementasi

Implementasi menggunakan metode *black box* adalah pendekatan di mana pengujian dilakukan tanpa memperhatikan struktur internal atau logika

dari sistem yang diuji. Fokusnya adalah pada pengujian fitur dan fungsi yang terlihat dari luar sistem, sebagaimana dilihat oleh pengguna akhir.

BAB II

LANDASAN TEORI

2.1. Penelitian Terdahulu

Tujuan dari penelitian terdahulu ini adalah untuk mendeskripsikan penelitian-penelitian yang telah dilakukan sebelumnya. Dalam hal ini, penelitian yang diajukan oleh penulis dibandingkan dengan penelitian sebelumnya untuk menunjukkan perbedaan antara ketiganya.

Pada penelitian pertama ini yang diteliti oleh (Lesmana dan Silalahi, 2022), metode *waterfall* dan bahasa pemrograman PHP Procedural atau PHP Native yang di gunakan. Kelemahan dari penelitian ini adalah masih menggunakan metode *waterfall*. Metode ini umumnya kurang fleksibel karena pengguna tidak dilibatkan dalam proses pengembangan sistem. Selain itu, pengembangan sistem menggunakan PHP prosedural atau PHP asli dapat memakan waktu lama karena keterbatasan teknologi yang digunakan.

Penelitian kedua ini yang ditulis (Priamnistiko dkk ., 2021) menggunakan pendekatan *waterfall* dan menggunakan *framework* PHP yaitu CodeIgniter. Namun, pendekatan ini memiliki beberapa kelemahan. Metode *waterfall* pada umumnya kurang efektif dalam memenuhi kebutuhan pengguna karena teknologi yang digunakan membutuhkan pengiriman data yang membutuhkan pihak ketiga. Hal ini dapat mempengaruhi fleksibilitas dan menambah kompleksitas pada proses pengembangan sistem.

Penelitian ketiga ini yang diteliti oleh (Hardianty dkk., 2022) menggunakan metode *prototyping* dengan menggunakan *framework* laravel. Namun, pendekatan ini memiliki kelemahan. Metode *prototyping* masih terfokus pada desain yang muncul dan pengembangannya sering memakan waktu lama dan teknologi laravel yang digunakan masih versi lama. Sebagai contoh ditunjukkan pada tabel di bawah ini.

Table 2. 1 Penelitian Sebelumnya

Judul	Metode	Framework	Kelemahan	Perbedaan
Merancang dan membangun sistem informasi <i>e-learning</i> berbasis <i>web</i> untuk TPQ al-maariji (Lesmana dan Silalahi, 2022)	<i>waterfall</i>	Penelitian ini menggunakan <i>php procedural</i> atau <i>php native</i>	Metode yang digunakan tidak cukup fleksibel karena kurangnya keterlibatan pengguna, dan pengembangan sistem yang memakan waktu lama karena terbatasnya teknologi yang digunakan.	Fungsi yang ditawarkan lebih detail dari sekedar upload materi dan mengisi kehadiran. Metode yang digunakan adalah metode terbaru yaitu RAD (<i>Rapid Application Development</i>) yang mengutamakan kebutuhan pengguna.

<p><i>Membuat Sistem Informasi E-Learning Menggunakan Metode Waterfall (Priamnisti ko dkk., 2021)</i></p>	<p><i>waterfall</i></p>	<p>Penelitian ini menggunakan salah satu <i>Framework</i> php yaitu codeigniter</p>	<p>Metode yang digunakan secara umum kurang efektif dalam memenuhi kebutuhan pengguna, karena teknologi yang digunakan membutuhkan <i>migration</i> dengan pihak ketiga.</p>	<p><i>Website</i> memiliki fitur yang berbeda, di mana pengguna memiliki kontrol penuh dalam mengakses mata pelajaran, sementara <i>admin</i> hanya memberikan konten yang diperlukan.</p>
<p>Rancang Bangun <i>E-learning</i> Berbasis Web Apps Untuk Pembelajaran Online dengan <i>Prototyping</i> (Hardianty dkk., 2022)</p>	<p><i>Prototyping</i></p>	<p>Penelitian ini menggunakan kerangka kerja laravel</p>	<p>Metode yang digunakan masih berfokus pada desain yang akan ditampilkan dan membutuhkan waktu yang cukup lama.</p>	<p>versi laravel yang berbeda, masih terdapat kekurangan dalam pemisahan otorisasi antara guru dan siswa.</p>

Kesimpulannya adalah ketiga penelitian terdahulu tersebut masih menggunakan teknologi dan metode lama, kemudian ada dua hal yang berbeda yaitu metode dan juga teknologi yang digunakan, yang penulis buat yaitu menggunakan metode RAD dan juga laravel 10. Dimana metode dan teknologi ini adalah teknologi dan metode terbaru. Metode ini lebih fokus memahami kebutuhan mahasiswa dan ekosistem *website*, Jadi dapat mengurangi kesalahan fatal pada saat tahap *deploy* nanti dengan bantuan teknologi laravel memungkinkan semua code berjalan dengan baik, karena teknologi ini lebih detail dengan dukungan fitur-fitur yang disediakan oleh laravel sendiri.

2.2. Landasan Teori

2.2.1. Rancang Bangun Sistem

Rancang bangun sistem adalah sebuah langkah setelah menganalisis siklus pengembangan sistem, definisi kebutuhan fungsional, serta gambaran bagaimana sistem dibuat, yang dapat dilakukan dalam bentuk gambar, denah dan sketsa. (Mulyanto dkk., 2020). Rancang bangun merupakan tahap awal perancangan sistem yang dimulai dengan analisis, kebutuhan, visualisasi dan sketsa kerja sistem.

2.2.2. E-learning

E-Learning adalah metode belajar melalui *internet* (Riskiono dan Pasha, 2020) menyatakan bahwa *e-learning* adalah kependekan dari *electronic learning* dan mengacu pada proses pembelajaran yang

menggunakan media elektronik untuk penyampaian materi, diskusi, ujian dan kegiatan konferensi lainnya.

E-learning adalah model pembelajaran yang menggunakan teknologi informasi dan komunikasi untuk kebutuhan mengajar maupun belajar. (Asmara, 2021). Pernyataan diatas bisa disimpulkan bahwa *e-learning* menjadikan metode untuk belajar dari sebuah media *elektronik* dengan memanfaatkan fasilitas teknologi dan membangun komunikasi dalam prosesnya.

2.2.3. E-course

Online course adalah salah satu bentuk dari *e-learning*. Menurut (Rahmadi dan Hayati, 2020) menyatakan bahwa platform pembelajaran online merupakan konsep dari pembelajaran jarak jauh yang memiliki skala lebih luas dan dapat diakses oleh banyak orang.

E-course merupakan sebuah konsep pendidikan yang menempatkan semua materi di dalam sebuah teknologi baik berupa *website* ataupun yang lainnya bukan hanya pelajaran tetapi ujian, semua dan diskusi yang berhubungan dengan mata pelajaran akan disimpan .(Dahri dkk., 2022).

E-Course adalah salah satu produk turunan dari sebuah konsep pembelajaran *e-learning* yang mana *e-course* sendiri hanya mengacu pada kursus ataupun pembelajaran *online* yang terstruktur dengan materi pembelajaran yang terorganisir.

2.2.4. Website

Website menurut (Novitasari dkk., 2021) menyatakan *situs web* adalah kumpulan halaman *web* tentang topik terkait, terkadang disertai dengan gambar, video, atau *file* lainnya. Membuat suatu *web* didalam *internet* membutuhkan banyak sekali hal yang harus digunakan, yang paling penting adalah bahasa pemrograman yang akan digunakan sebagai membuat *website* dinamis, Bahasa yang banyak digunakan untuk membentuk *website* dinamis yaitu php.

PHP adalah bahasa pemrograman sisi *server* yang dirancang untuk pengembangan *web* yang dinamis dan interaktif.(Rina Noviana, 2022). Untuk membangu sebuah *website* dengan php membutuhkan beberapa *tools* tambahan yang dapat mengakses *local server* dan database salah satunya adalah xampp.

2.2.5. RAD (Rapid Application Development)

Rapid application development (RAD), model dari beberapa *model System Development Life Cycle* (SDLC). *Rapid application development* (RAD) adalah model proses perangkat lunak yang cepat dan efisien yang menekankan pada siklus pengembangan yang sangat singkat (Hidayat dan Hati, 2021). Perancangan metode ini melibatkan beberapa tahapan pengembangan sistem, yaitu :



Gambar 2. 1 Siklus Pengembangan Rad

2.2.5.1. Rencana Kebutuhan (Requirements Planning)

Identifikasi dan pemahaman awal terhadap kebutuhan pengguna atau pemangku kepentingan kebutuhan pengguna dan melakukan penyesuaian dan revisi jika diperlukan. Hal ini bertujuan untuk memastikan bahwa kebutuhan pengguna dan persyaratan universitas terpenuhi dalam pengembangan *website e-course* yang efektif dan sesuai dengan konteks universitas tersebut.

2.2.5.2. Desain Sistem

Melakukan analisis kebutuhan lebih lanjut untuk mengidentifikasi fungsi-fungsi penting yang akan ditampilkan dalam suatu sistem, pada tahap ini peneliti akan mengembangkan antarmuka pengguna yang menarik, dan memastikan adanya integrasi yang baik dengan sistem yang ada. Selain itu, penting juga untuk memperhatikan otorisasi yang memisahkan peran *admin* dan mahasiswa,

2.2.5.1. Pengembangan

Pada tahap pengembangan adalah tahap implementasi yang sudah disepakati dan direncanakan sebelumnya dimulai dari rancangan interface dan model bisnis yang akan diterapkan.(Hidayat dan Hati, 2021).

2.2.5.2. Implementasi

Tahap ini adalah pengujian program yang sudah dibuat secara keseluruhan dan sistem yang dibangun semua sistem yang di

bangun akan di uji dengan metode *black box testing* agar mengurangi resiko bug di dalam program (Hidayat dan Hati, 2021)

2.2.6. *Unified Modelling Language (UML)*

Unified modelling language menurut (Mubarak, 2019) UML adalah bahasa yang sering digunakan oleh bidang analisis sistem untuk membuat desain sistem dalam bentuk grafik atau gambar untuk memvisualisasikan, mendefinisikan, membangun dan mendokumentasikan sistem yang akan berfungsi. Secara garis besar, ada 4 UML yang umum digunakan sebagai berikut :

2.2.6.1. *Use case diagram*

Use case Diagram menurut (Aulia, 2022). Adalah sebuah *use case diagram* yang menggambarkan aktor, *use case*, dan hubungannya sebagai aktivitas yang membawa nilai terukur bagi aktor, kasus penggunaan digambarkan dalam diagram UML sebagai *elips horizontal*.

Use case diagram merupakan representasi visual dari fungsionalitas yang disediakan oleh sistem, yang memungkinkan berbagai unit atau aktor yang terhubung dalam sistem untuk bertukar informasi dan berinteraksi secara efisien. (Reza Fahlevi dkk., 2023).

Table 2. 2 *Use Case Diagram*

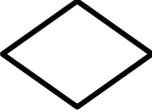
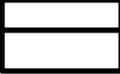
Simbol	Nama	Keterangan
	<i>actor</i>	Entitas eksternal yang berhubungan dengan sistem yang sedang dianalisis atau dirancang

	<i>system</i>	Sistem ini disajikan melalui paket yang memuatnya secara terbatas.
	<i>Use case</i>	Abstraksi dan interaksi antara <i>aktor</i> dan sistem
	<i>association</i>	Keterkaitan antar kelas atau entitas yang menunjukkan keterkaitan atau interaksi antara keduanya.
	<i>generalisasi</i>	<i>Spesialisasi aktor</i> yang bisa berhubungan dengan <i>use case</i>
	<i>Relasi Include</i>	Menunjukkan bahwa <i>use case</i> sepenuhnya mewakili fungsionalitas dari <i>use case</i> lain
	<i>Relasi Extend</i>	Hubungan antara kasus penggunaan tersebut adalah bahwa kasus penggunaan yang ditambahkan dapat berdiri sendiri tanpa memerlukan kasus penggunaan tambahan.

2.2.6.2. Activity diagram

activity diagram menurut Aulia, (2022) menyatakan bahwa Tujuan dari diagram *activity* adalah untuk menggambarkan fungsi, objek, keadaan, transisi keadaan, dan peristiwa. Dengan kata lain, *activity diagram* alur kerja menggambarkan perilaku sistem untuk aktivitas.

Table 2. 3 *Activity Diagram*

Simbol	Nama	Keterangan
	Status keadan pertama	Sebuah simbol yang menggambarkan sebuah status awal
	Status akhir	Sebuah simbol yang menggambarkan keadaan akhir .
	<i>Activity</i>	Tindakan yang dilakukan oleh sistem biasanya dimulai dengan kata kerja
	Perkodisian	Tindakan dengan lebih dari satu opsi pilihan atau keadaan
	Pengabungung	Penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	<i>Swimlane</i>	memisahkan organisasi komersial yang bertanggung jawab atas aktivitas yang berlangsung

2.2.6.3. *Sequence diagram*

Menggambarkan hubungan dinamis antar objek, yang tujuannya adalah untuk menunjukkan urutan pesan yang dikirim antar objek dan interaksi antar objek yang terjadi pada tahap tertentu dari eksekusi sistem. (Tedyyana dkk., 2019).

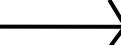
Table 2. 4 *Sequence Diagram*

Simbol	Nama	Keterangan
	peran	Menjelaskan situasi di mana seseorang berinteraksi dengan sistem.
	Kelas entitas	Merujuk pada kelas entitas yang menggambarkan relasi yang akan dilakukan.
	Kelas batas	gambaran umum tentang sebuah formulir atau format.
	Kelas kontrol	Merupakan kelas yang menggambarkan hubungan atau penghubung antara antarmuka (boundary) dengan tabel atau entitas dalam suatu sistem.
	Fokus kontrol & garis langsung	Tempat dimulai dan diakhirinya pesan
	<i>message</i>	Menjelaskan penyampaian pesan

2.2.6.4. *Class Diagram*

Class diagram menurut (Tedyyana dkk., 2019) Diagram kelas dimaksudkan untuk menggambarkan struktur statis kelas dalam suatu sistem. Kelas yang mewakili sesuatu untuk diproses oleh sistem.

Table 2. 5 *Class Diagram*

Simbol	Nama	Keterangan
	<i>generalization</i>	Hubungan objek anak berbagi perilaku dan struktur data dengan objek induk
	<i>Nary association</i>	Usaha untuk Hindari kontak dengan lebih dari dua objek
	kelas	Kumpulan objek yang memiliki atribut dan fungsi yang sama
	<i>colaborasi</i>	Urutan aksi dari deskripsi yang ditampilkan oleh sistem menghasilkan sesuatu yang dapat diukur oleh operator
	realisasi	Tindakan aktual yang dilakukan oleh objek
	<i>ketergantungan</i>	Hubungan di mana perubahan pada elemen independen memengaruhi elemen itu bergantung padanya
	<i>asosiasi</i>	Apa yang menghubungkan satu objek dengan objek lainnya

2.2.7. Laravel

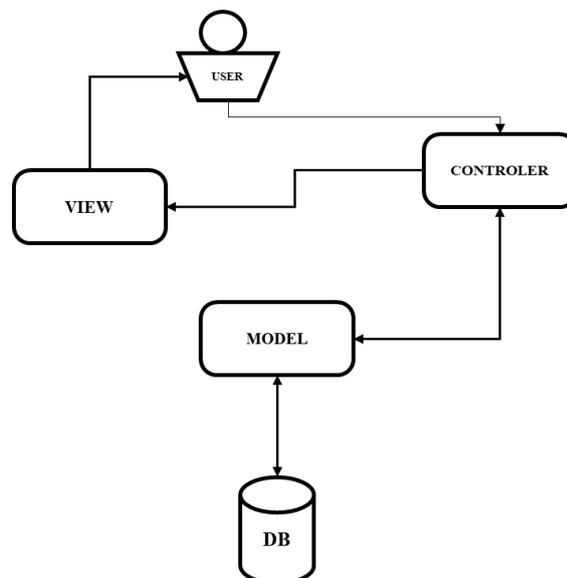
Laravel adalah salah satu dari banyak kerangka kerja PHP berorientasi objek berdasarkan konsep MVC atau *Model View*

Controller. Laravel adalah kerangka kerja untuk mengembangkan situs web berdasarkan model MVP atau *View Presenter*, yang ditulis dalam bahasa pemrograman PHP. (Harahap dkk., 2022).

framework laravel 10 dengan keunggulan pengembangan *aplikasi web* yang mudah. Laravel 10 menawarkan banyak fitur dan alat baru seperti untuk menyederhanakan pengembangan aplikasi *web*, termasuk *migrasi basis data*, perutean fleksibel, pemodelan sistem, dan sistem autentikasi yang aman (Subiksa dkk.,2023).

2.2.8. MVC (*Model-View-Controller*)

MVC adalah teknik desain aplikasi yang memisahkan lapisan data atau *model* lapisan tampilan, dan lapisan pengontrol. Hubungan antara *model layer*, *view layer* dan *controller layer* sangat erat, sehingga setiap layer harus saling berhubungan. (Atmaja dkk., 2021).



Gambar 2. 2 Konsep Mvc

2.2.8.1. Model

Model adalah konsep pada sebuah komponen yang bertanggung jawab untuk mengelola data dan logika bisnis dalam sebuah *website*.

2.2.8.2. View

View merupakan sebuah komponen yang bertanggung jawab untuk menampilkan informasi *website* kepada pengguna dan menerima input dari pengguna.

2.2.8.3. Controller

Controller merujuk pada komponen yang bertanggung jawab untuk mengatur aliran logika, menerima masukan dari pengguna, dan mengirimkan instruksi kepada *model* dan *view*.

2.2.9. MySQL

MySQL adalah sebuah program database *server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi mahasiswa*, serta menggunakan perintah standar SQL (*Structured Query Language*) (Ahmadar dkk., 2021).

2.2.10. Testing

Untuk menguji fungsi-fungsi *black box* sangat membantu (Febriyanti dkk., 2021). Pengujian *black box* didasarkan pada detail aplikasi, seperti tampilan dan tampilan *situs web*, fungsionalitas *situs web*, dan penerapan fungsionalitas tersebut ke proses bisnis yang diinginkan oleh pengguna.

2.2.11. System Usability Scale (SUS)

System Usability Scale (SUS) adalah alat yang digunakan untuk mengukur persepsi pengguna terhadap *usability* (kemudahan penggunaan) pada suatu sistem, metode ini sering digunakan dalam pengembangan perangkat lunak dan desain antarmuka pengguna. (Damayanti dkk., 2022).

Sus terdiri dari serangkaian pertanyaan yang dirancang untuk mengevaluasi berbagai aspek *usability* suatu sistem, termasuk kemudahan penggunaan, kejelasan pesan, dan kepuasan pengguna secara umum dengan cara diberikan 5 jawaban yang mana masing-masing jawaban terdapat nilainya sendiri.

Dengan menggunakan sus, pengembang dapat memperoleh wawasan yang berharga tentang seberapa baik sistem mereka dapat diakses dan dipahami oleh pengguna akhir berikut tabel nilai akhir yang ada pada sus.

Table 2. 6 Nilai *System Usability Scale*

Sus nilai	Nilai huruf	peringkat
>80.3	A	Baik sekali
68-80.3	B	baik
68	C	Sedang
51-68	D	Kurang
<51	F	Kurang sekali

BAB III

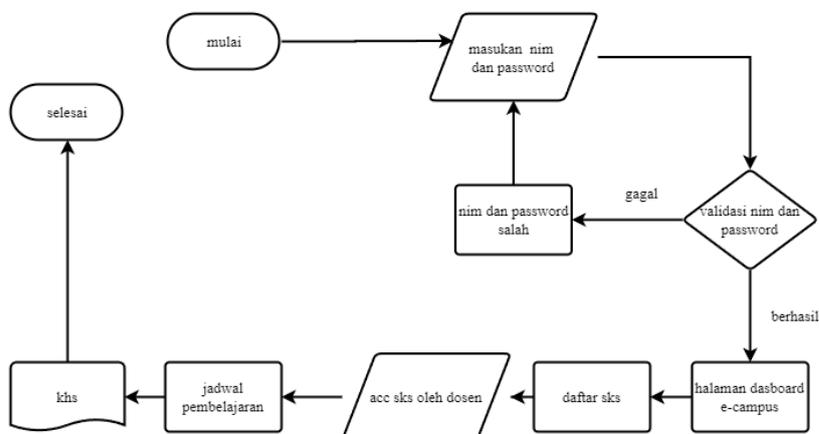
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem Berjalan

3.1.1. Analisis Proses Bisnis

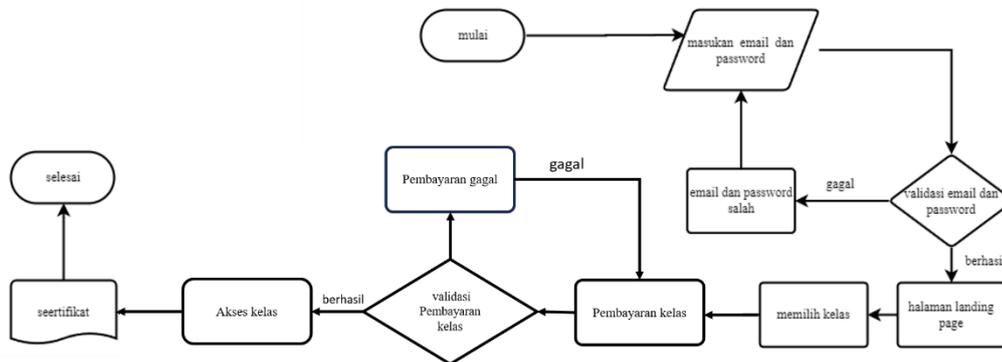
Sistem *e-learning* yang diterapkan di UNUSIA menggunakan pendekatan *e-campus* untuk mendukung kegiatan akademik di kampus. Sistem ini telah dirancang untuk mengatur proses pengambilan mata kuliah berdasarkan kurikulum yang ada, memastikan bahwa pendaftaran dan penjadwalan berjalan dengan lancar. Namun, perlu dicatat bahwa dalam sistem ini, pembelajaran masih dilakukan dengan metode hybrid, yang menggabungkan komponen *online* dan *offline*.

Kekurangan dari sistem ini adalah mahasiswa tidak dapat kebebasan memilih apa yang mau mereka pelajari dan pembelajaran hybrid membutuhkan mahasiswa untuk mengikuti pertemuan tatap muka di kampus, yang dapat membatasi fleksibilitas jadwal mereka. mahasiswa mungkin menghadapi kendala dalam mengatur waktu mereka antara kehadiran di kampus dan keterlibatan dalam kegiatan lain di luar kampus.



Gambar 3. 1 Sistem Yang Berjalan

Sistem yang diusulkan dalam penelitian ini adalah sistem *e-course* yang berbasis *website* sistem ini menambah sistem pembelajaran hybrid yang melalui *meet online* dan masih tradisional. Dengan adanya sistem *e-course* ini dapat memberikan pengalaman belajar yang sesuai dengan mahasiswa dan dapat digunakan dimana saja dan kapan saja selama terkoneksi dengan internet.



Gambar 3. 2 Sistem Yang Diusulkan

3.1.2. Kebutuhan Fungsional

A. Admin

Adapun tugas *admin* sebagai berikut

1. melihat, mengedit dan menghapus data *user* kelas.
2. Menambah, melihat ,mengedit dan menghapus kelas.
3. Mengkonfirmasi pembayaran kelas.
4. Melihat jumlah kelas yang sudah di beli.

B. Author

Adapun tugas *author* sebagai berikut

1. Menambah, melihat ,mengedit dan menghapus kelas.
2. Melihat jumlah kelas yang sudah di beli.

C. user

adapun tugas *user* adalah sebagai berikut

1. Melihat kelas yang sudah dibeli.
2. Mengisi biodata akun.
3. Mendapatkan akses kelas.

4. Generate sertifikat.

3.1.3. Kebutuhan Non Fungsional

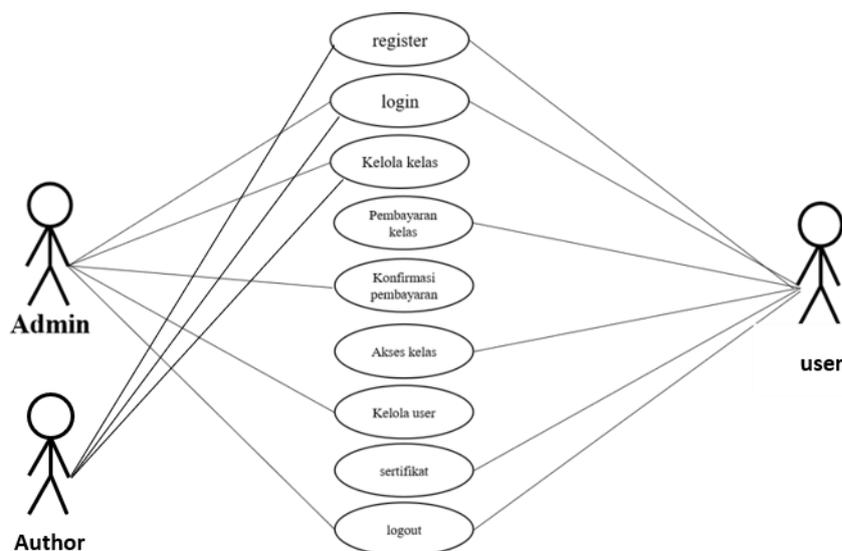
- A. Fungsi generate sertifikat hanya bisa diakses oleh *user*.
- B. *User* tidak bisa Membuka akses kelas sebelum dibuka oleh *admin*.
- C. Tampilan *interface* tidak terlalu rumit dan mudah dipahami oleh pengguna.
- D. Keamanan akan terjamin untuk membantu menunjang pengalaman *user*.
- E. Tampilan sudah responsif di berbagai platform.

3.2 Desain Sistem Yang Diusulkan

Desain sistem pada penelitian ini menggunakan beberapa pemodelan sistem yaitu *Use Case*, *Activity*, *Sequence Diagram* dan *Class Diagram*.

3.2.1 Pemodelan Use Case

Use case diagram terdiri dari *admin author* dan *user* serta hubungannya. *Use Case diagram* digunakan untuk menjelaskan kegiatan apa saja yang dapat dilakukan oleh *admin*, *author* dan *user* pada sistem yang berjalan.



Gambar 3. 3 Pemodelan *Use Case*

3.2.1.1 Use Case Register

- Use case* : *Register*
 Nama aktor : *author,user*
 Deskripsi : Proses mendaftar akun untuk masuk ke sistem
 Kondisi awal : *user* menginputkan email dan *password*
 Kondisi akhir : *user* dapat masuk ke dalam sebuah sistem

Table 3. 1 *Use Case Register*

<i>Use case utama</i>			
no	<i>user</i>	no	sistem
1	<i>User</i> mengunjungi <i>website e-course</i> dan klik <i>icon register</i>	2	Menampilkan halaman <i>register website e-course</i>
<i>Alternatif 1 masukan email dan password</i>			
1	Pada halaman <i>register</i> masukan <i>email</i> dan <i>password</i> dan ulangi <i>password</i>	2	Validasi apakah semua format ditulis dengan benar, validasi kirim notifikasi <i>akun</i> telah terdaftar

3.2.1.2 Use Case Login

- Use case* : *Login*
 Nama aktor : *Admin, author,user*
 Deskripsi : proses masuk ke dalam sebuah sistem
 Kondisi awal : Aktor menginputkan email dan *password*
 Kondisi akhir : Aktor dapat masuk ke dalam sebuah sistem

Table 3. 2 *Use Case Login*

<i>Use case utama</i>			
no	<i>user</i>	no	sistem

1	User mengunjungi halaman <i>login</i> di halaman utama <i>website e-course</i>	2	Menampilkan halaman <i>login website e-course</i>
Alternatif 1 masukan email dan <i>password</i>			
1	Pada halaman <i>login</i> masukan email dan <i>password</i>	2	Validasi apakah email dan <i>password</i> ada di database, validasi apakah email dan <i>password</i> sesuai

3.2.1.3 Use Case Kelola kelas

Use case : Kelola kelas

Nama aktor : *Admin*

Deskripsi : proses menambah mengedit dan menghapus kelas

Kondisi awal : *admin* masuk ke menu tambah kelas

Kondisi akhir : kelas akan masuk ke seksion kelas pada halaman utama

Table 3. 3 *Use Case* Kelola Kelas

<i>Use case</i> utama			
no	<i>user</i>	no	sistem
1	<i>Admin</i> masuk ke dalam session di dalam <i>dashboard</i>	2	Menampilkan halaman tambah kelas
Alternatif 1 Tambah kelas			
1	Pada halaman course tambah kelas <i>admin</i> akan input judul kelas dan beberapa kelas yang ingin di input	2	Validasi apakah format sesuai dengan tambah kelas validasi Ketika sukses aka nada notifikasi
Alternatif 2 edit kelas			

1	Pada halaman course edit kelas <i>admin</i> akan edit course kelas dan beberapa kelas yang ingin di edit	2	Validasi apakah kelas ingin di edit validasi notifikasi bahwa kelas telah di edit dan diperbaharui
Alternatif 3 hapus kelas			
1	Pada halaman course <i>delete</i> kelas <i>admin</i> akan hapus kelas dan beberapa kelas yang ingin di kelas	2	Validasi apakah kelas ingin di hapus validasi notifikasi bahwa kelas telah di hapus dari database

3.2.1.4 Use Case pembayaran kelas

Use case : Pembayaran Kelas
 Nama aktor : *user*
 Deskripsi : proses pembayaran kelas yang ingin didapatkan
 Kondisi awal : *user* masuk ke dalam session kelas
 Kondisi akhir : *user* akan ada notifikasi pembayaran

Table 3. 4 Pembayaran Kelas

<i>Use case</i> utama			
no	<i>user</i>	no	sistem
1	<i>user</i> masuk ke dalam session kelas	2	Menampilkan halaman kelas yang tersedia
Alternatif 1 inputkan nama kelas			
1	Pada halaman kelas <i>users</i> akan memilih kelas yang ingin dibeli aksesnya	2	Akan menampilkan notifikasi pembayaran sesuai dengan kelas yang ingin diberi akses

3.2.1.5 Use Case konfirmasi pembayaran

Use case : Konfirmasi Pembayaran Kelas

Nama aktor : *admin*
 Deskripsi : proses konfirmasi pembayaran kelas mahasiswa
 Kondisi awal : *admin* masuk menu pembayaran di *navbar*
 Kondisi akhir : *admin* validasi pembayaran dengan sukses

Table 3. 5 Konfirmasi Pembayaran

<i>Use case</i> utama			
no	<i>admin</i>	no	sistem
1	<i>Admin</i> masuk menu transaksi di halaman <i>dashboard</i>	2	Menampilkan mahasiswa yang transaksi
Alternatif 1 sukses dan gagal transaksi			
1	Pada halaman transaksi <i>admin</i> akan memilih sukses dan gagal	2	Akan menampilkan transaction Sesuai dengan format, validasi transaksi sukses

3.2.1.6 *Use case* akses kelas

Use case : Akses Kelas
 Nama aktor : *user*
 Deskripsi : proses konfirmasi akses kelas
 Kondisi awal : *user* masuk ke halaman detail kelas
 Kondisi akhir : *user* akan mendapatkan kelas tanpa kunci

Table 3. 6 Akses Kelas

<i>Use case</i> utama			
no	<i>User</i>	no	sistem
1	<i>user</i> masuk ke halaman detail kelas	2	Menampilkan menampilkan kelas yang sudah di beri akses oleh <i>admin</i>

3.2.1.7 Use Case Kelola akun

Use case : Kelola akun
 Nama aktor : *Admin*
 Deskripsi : proses menambah, mengedit dan menghapus kelas
 Kondisi awal : *admin* masuk ke menu edit akun
 Kondisi akhir : *admin* mengubah data akun

Table 3. 7 Kelola Mahasiswa

<i>Use case</i> utama			
no	<i>Admin</i>	no	sistem
1	<i>Admin</i> masuk ke dalam halaman kelola akun	2	Menampilkan halaman Kelola akun
Alternatif 2 edit <i>users</i>			
1	Pada halaman akun <i>admin</i> akan edit akun dan beberapa dan beberapa akun yang ingin di edit	2	Validasi apakah <i>users</i> ingin di edit validasi <i>notifikasi</i> bahwa mahasiswa telah di edit dan diperbaharui

3.2.1.8 Use Case sertifikat

Use case : sertifikat
 Nama aktor : *user*
 Deskripsi : proses mengunduh sertifikat
 Kondisi awal : *user* masuk ke detail kelas
 Kondisi akhir : *user* telah download sertifikat

Table 3. 8 Generate Sertifikat

<i>Use case</i> utama			
no	<i>user</i>	no	sistem

1	<i>User</i> masuk ke halaman kelas klik selesai kelas	2	Menampilkan halaman kelas dan unduh sertifikat
---	---	---	--

3.2.1.9 Use Case logout

Use case : Keluar Sistem
 Nama peran : *Admin, author* dan mahasiswa
 penjelasan : Keluar dari sistem *e-course*
 Kondisi awal : Actor memilih menu atau tombol keluar
 Kondisi akhir : Actor akan keluar dari sistem

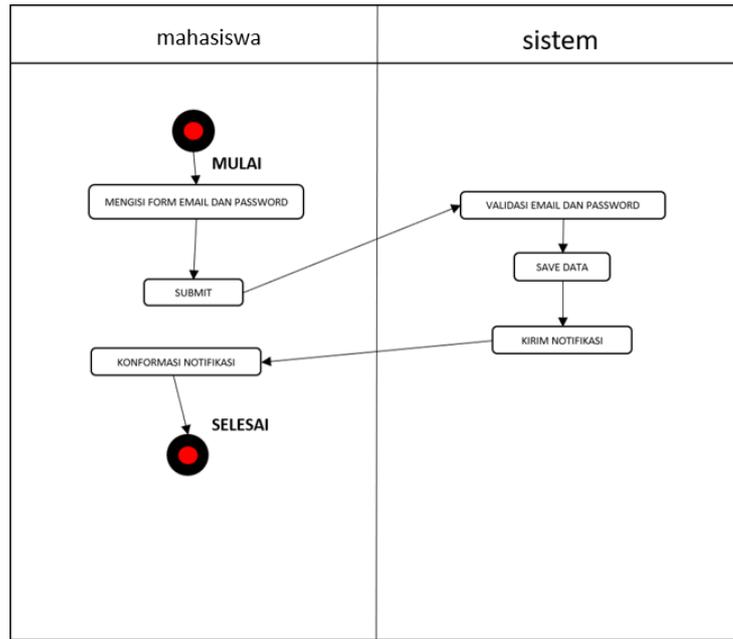
Table 3.9 *logout*

<i>Use case</i> utama			
no	<i>User</i>	no	sistem
1	<i>user</i> memilih tombol <i>logout</i> atau keluar	2	Mahasiswa telah keluar dari sistem

3.2.2 Pemodelan Activity Diagram

3.2.2.1 Activity Diagram Register

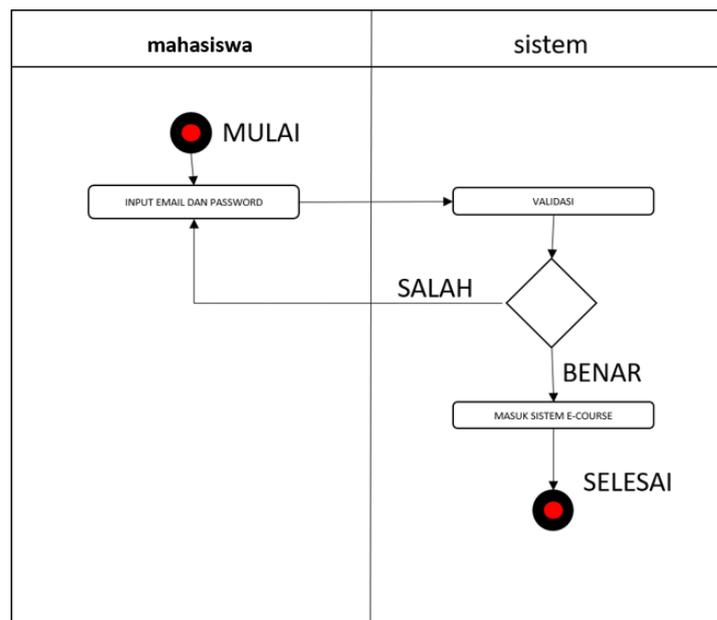
Dalam sistem *e-course*, pengguna akan diarahakan ke halaman *registrasi* dengan nama lengkap, email, *password* dan ulangi *password*. Setelah pengguna memasukkan informasi, sistem akan memvalidasi. Jika informasi benar, pengguna harus verifikasi alamat email yang terdaftar dan baru dapat mengakses fitur-fitur. Namun, jika terjadi kesalahan dalam penulisan email atau *password*, pengguna harus memperbaiki sebelum melanjutkan.



Gambar 3. 4 Implementasi *Activity Diagram Register*

3.2.2.2 *Activity Diagram Login*

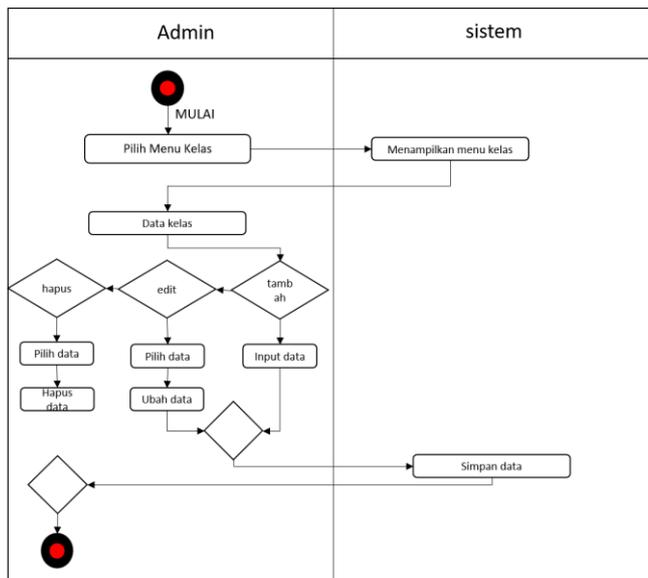
User menginput email dan *password*, sistem akan memvalidasi, jika sesuai maka masuk ke sistem, jika salah maka diarahkan untuk mengisi email dan *password* kembali dan muncul notifikasi.



Gambar 3. 5 Implementasi *Activity Diagram login*

3.2.2.3 Activity Diagram Kelola Kelas

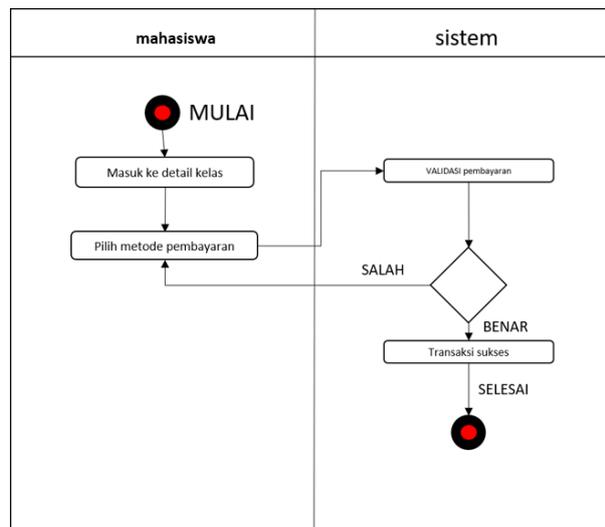
Admin memilih menu kelola kelas, kemudian sistem menampilkan data kelola kelas, kemudian *admin* dapat menambah, Edit kelas dan semua perubahan akan disimpan dalam penyimpanan data.



Gambar 3. 6 Implementasi Activity Diagram Kelola Kelas

3.2.2.4 Activity Diagram Pembayaran Kelas

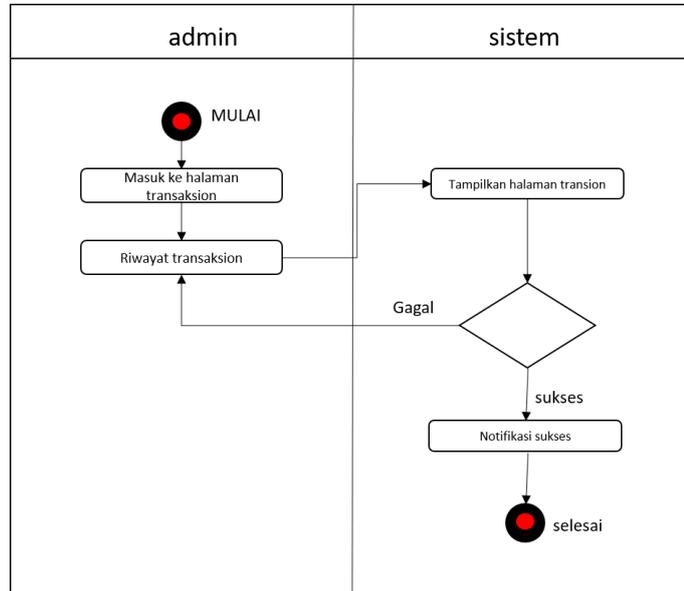
user masuk ke halaman detail kelas dan memilih metode pembayaran , sistem akan memvalidasi pembayaran ketika berhasil akan ada notifikasi transaksi sukses.



Gambar 3. 7 Implementasi Activity Diagram Pembayaran Kelas

3.2.2.5 Activity Diagram Konfirmasi Pembayaran

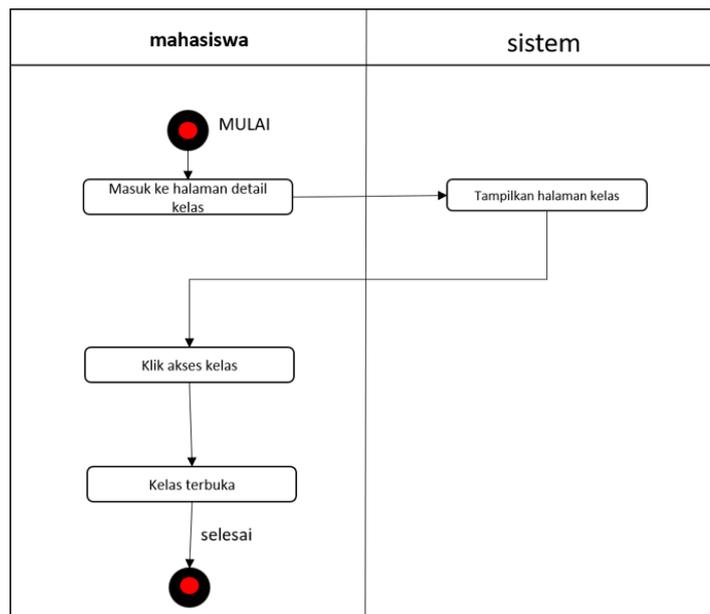
Admin masuk ke halaman menu transaksi dan memilih riwayat transaksi sistem akan memunculkan halaman transaksi, sistem akan memvalidasi pembayaran ketika berhasil akan ada notifikasi transaksi sukses.



Gambar 3. 8 Activity Diagram Konfirmasi Pembayaran

3.2.2.6 Activity Diagram Akses Kelas

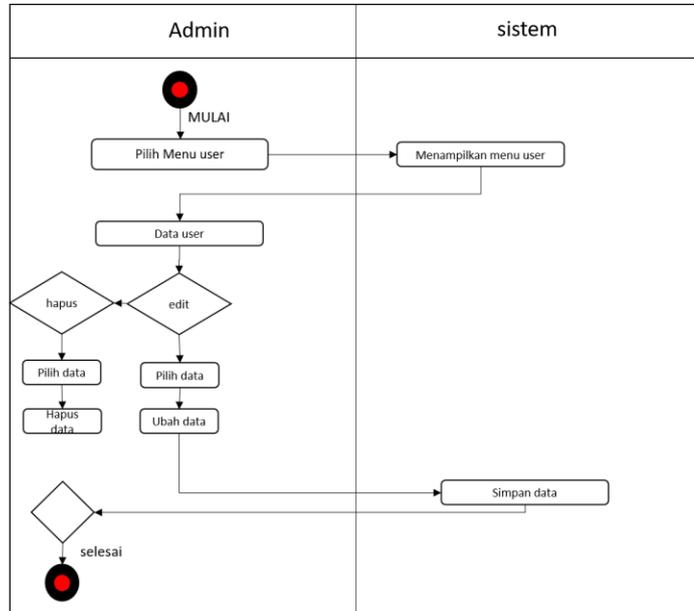
user masuk ke halaman detail kelas, sistem akan menampilkan detail kelas, *user* masuk ke akses kelas yang sudah dibeli dan kelas akan terbuka.



Gambar 3. 9 Implementasi Activity Diagram Akses Kelas

3.2.2.7 Activity Diagram Kelola User

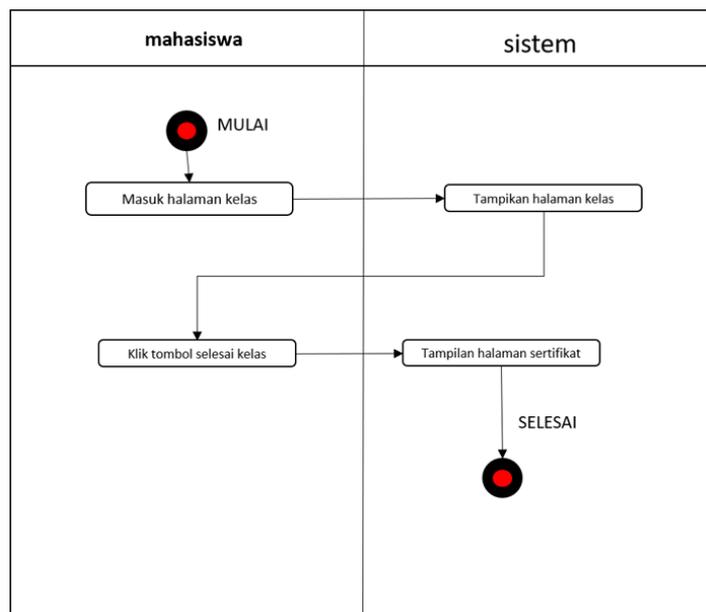
Admin memilih menu *Kelola user*. Sistem kemudian menampilkan informasi *user*, kemudian *admin* mengedit *user* kemudian semua perubahan yang terjadi di simpan ke gudang data.



Gambar 3. 10 Implementasi Activity Diagram Kelola Mahasiswa

3.2.2.8 Activity Diagram Sertifikat

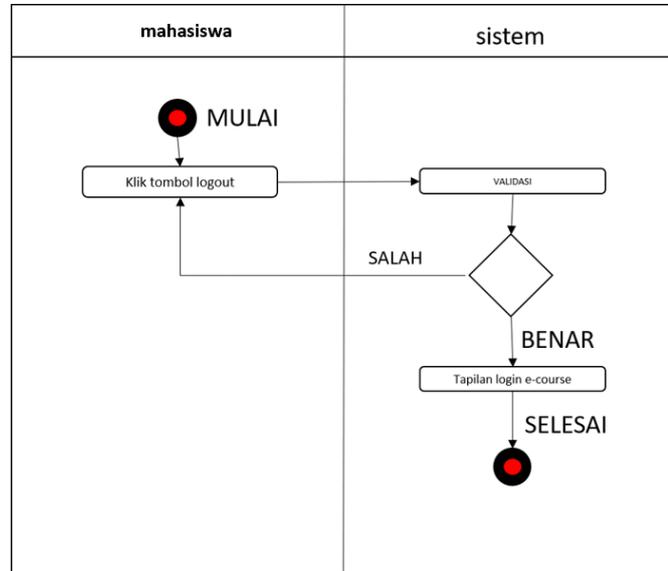
User masuk ke halaman detail kelas sistem akan menampilkan detail kelas *user* menekan tombol selesai kelas sistem akan menampilkan halaman sertifikat.



Gambar 3. 11 Implementasi Activity Diagram Sertifikat

3.2.2.9 Activity Diagram Logout

User memilih menu atau tombol *logout*, kemudian sistem memvalidasi jika benar maka diarahkan ke halaman *login*, Jika dalam kesalahan, tetap *login*.

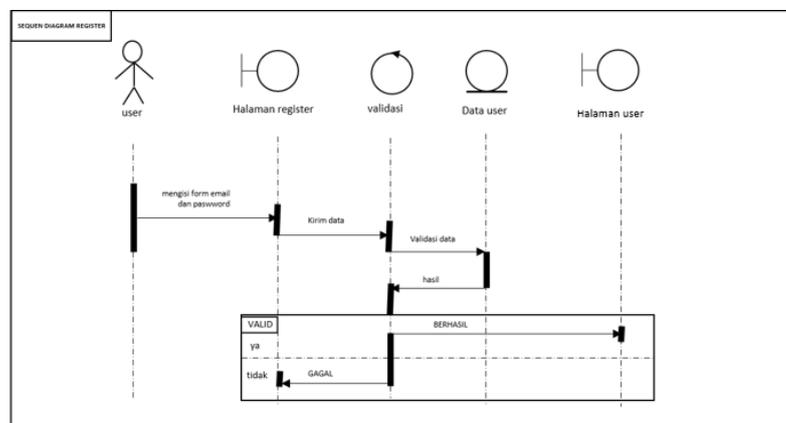


Gambar 3. 12 Implementasi *Activity Diagram Logout*

3.2.3 Sequence Diagram

3.2.3.1 Sequence Diagram Register

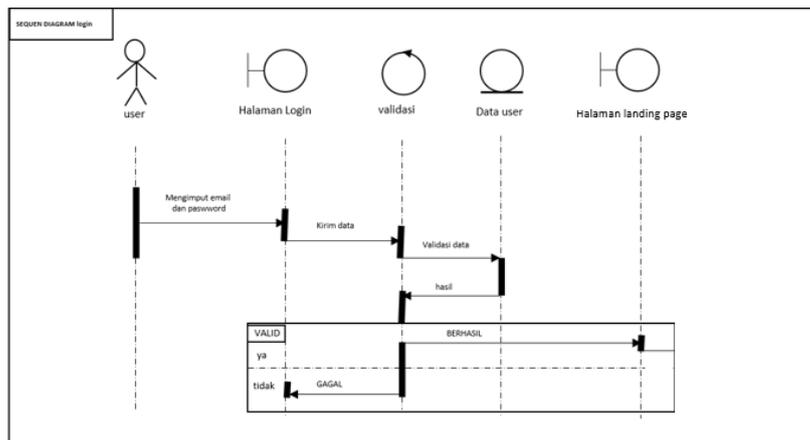
user menginput email dan *password*, kemudian sistem memvalidasi, jika benar maka masuk ke sistem, jika salah maka diarahkan untuk mengisi email dan *password* kembali.



Gambar 3. 13 Implementasi *Sequence Diagram Register*

3.2.3.2 Sequence Diagram login

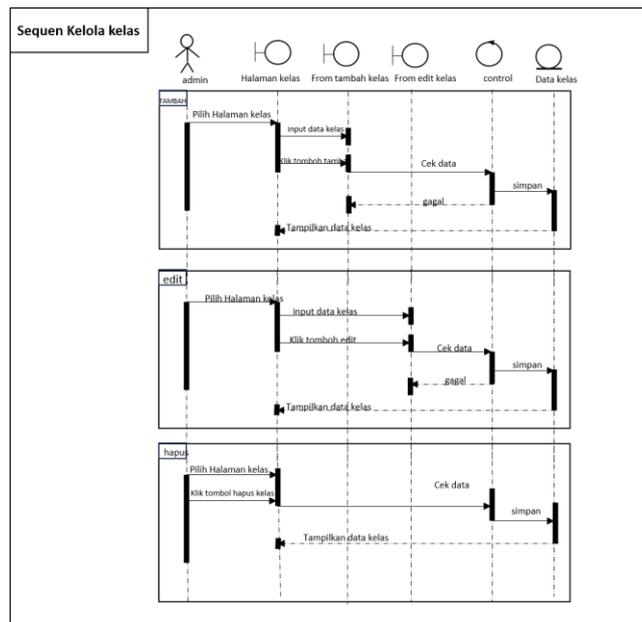
User menginput email dan *password*, sistem akan memvalidasi, jika sesuai maka masuk ke sistem, jika salah maka diarahkan untuk mengisi email dan *password* kembali dan muncul notifikasi.



Gambar 3. 14 Implementasi *Sequence Diagram Login*

3.2.3.3 Sequence Diagram Kelola Kelas

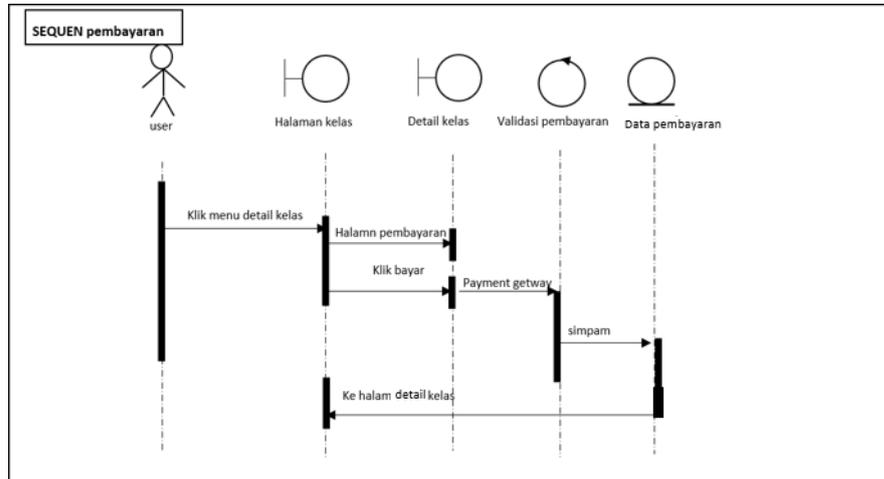
Admin memilih menu kelola kelas, kemudian sistem menampilkan data kelola kelas, kemudian *admin* dapat menambah, mengedit dan menghapus kelas kemudian semua perubahan yang terjadi di simpan ke database.



Gambar 3. 15 Implementasi *Sequence Diagram Kelola Kelas*

3.2.3.4 Sequence Diagram Pembayaran Kelas

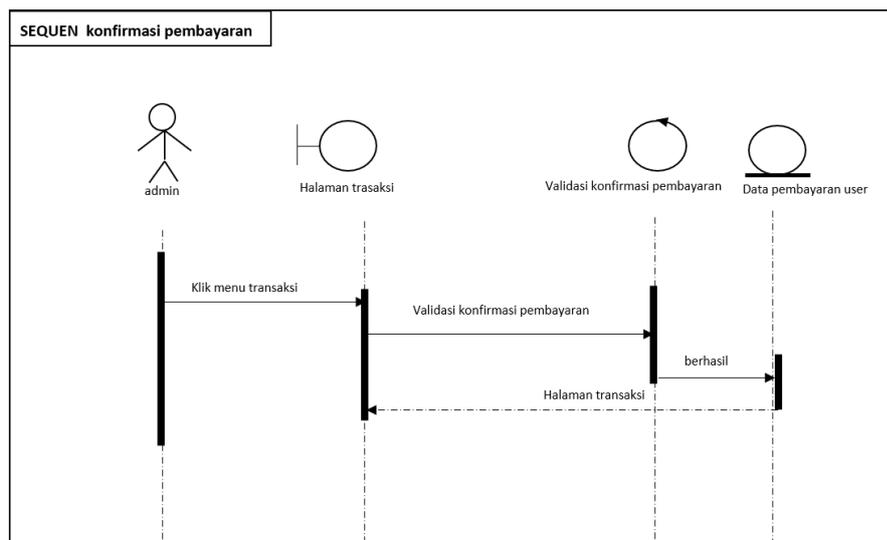
Setelah masuk ke halaman detail kelas, pengguna akan memilih metode pembayaran. sistem akan memvalidasi pembayaran, dan ketika berhasil, pengguna akan menerima notifikasi tentang transaksi yang sukses.



Gambar 3. 16 Implementasi *Sequence Diagram* Pembayaran Kelas

3.2.3.5 Sequence Diagram Konfirmasi Pembayaran

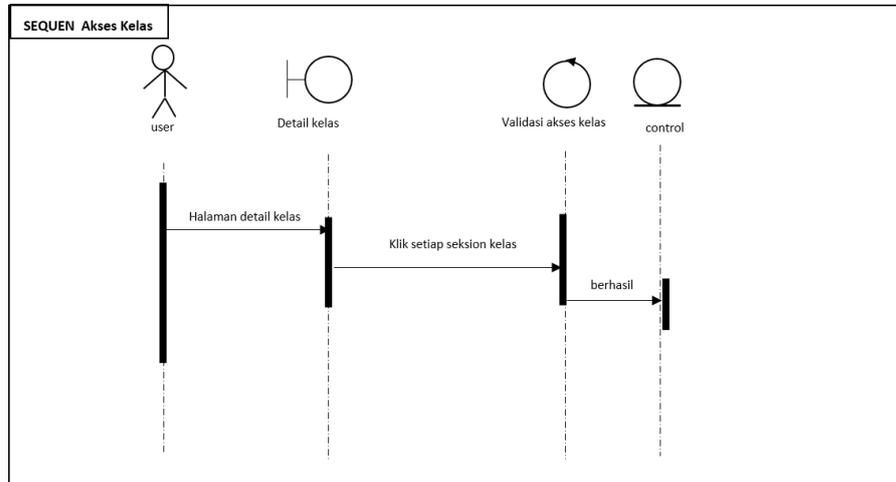
Admin masuk ke halaman menu transaksi dan memilih riwayat transaksi. kemudian, sistem akan menampilkan halaman transaksi terkait. Setelah itu, sistem akan memvalidasi pembayaran, dan jika berhasil, notifikasi tentang transaksi yang sukses akan muncul.



Gambar 3. 17 Implementasi *Sequence Diagram* Konfirmasi Pembayaran

3.2.3.6 Sequence Diagram Akses Kelas

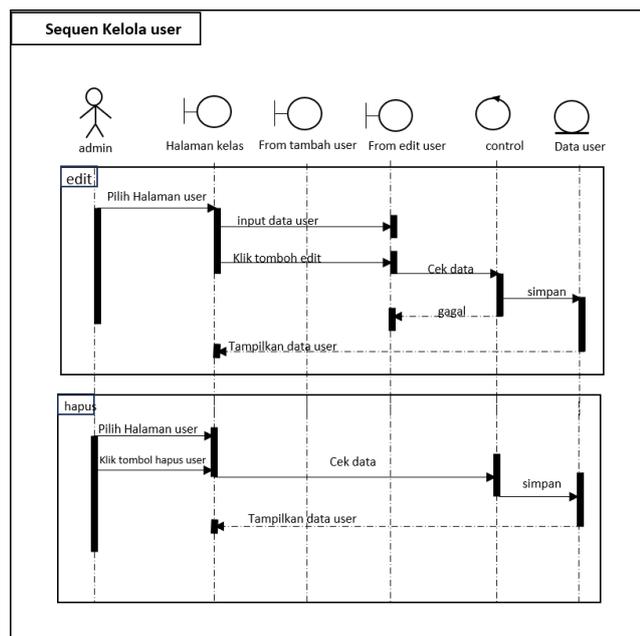
Ketika pengguna masuk ke halaman detail kelas, sistem akan menampilkan informasi detail mengenai kelas tersebut. Setelah itu, pengguna dapat mengakses kelas yang telah dibeli, dan kelas tersebut akan terbuka untuk pengguna.



Gambar 3. 18 Implementasi *Sequence Diagram* Akses Kelas

3.2.3.7 Sequence Diagram Kelola User

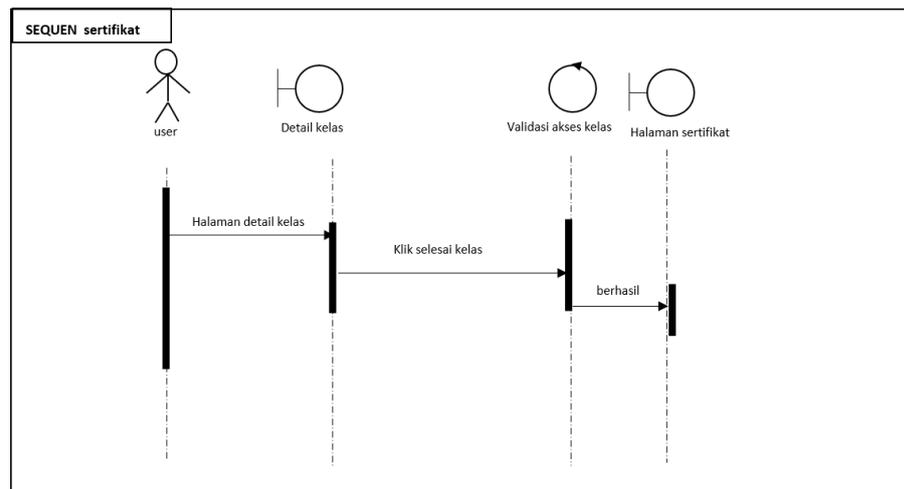
Admin memilih menu kelola *user* kemudian sistem akan menampilkan data pengguna yang dapat dikelola. *admin* dmengedit pengguna yang ada, dan melakukan perubahan lainnya.semua perubahan tersebut akan disimpan ke database.



Gambar 3. 19 Implementasi *Sequence Diagram* Kelola Mahasiswa

3.2.3.8 Sequence Diagram Sertifikat

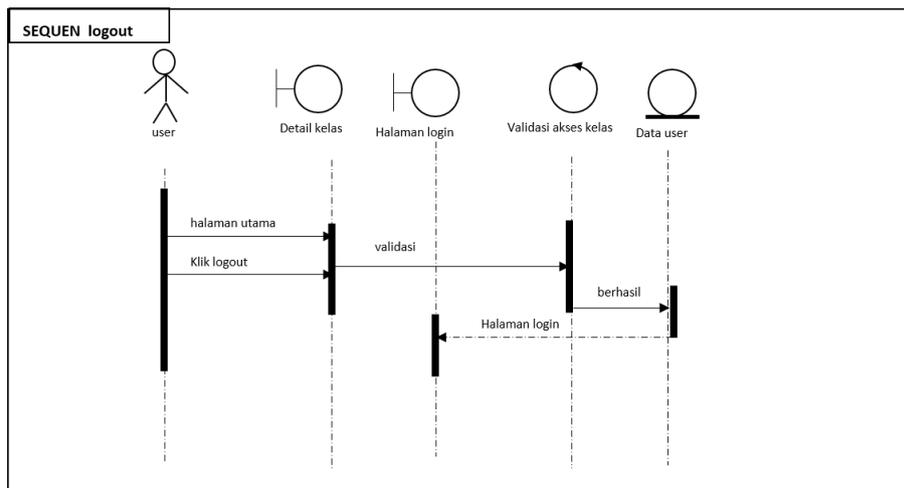
Ketika pengguna masuk ke halaman detail kelas, sistem akan menampilkan informasi detail mengenai kelas tersebut. Setelah itu, jika pengguna menekan tombol selesai kelas yang tersedia, sistem akan menampilkan halaman sertifikat kepada pengguna.



Gambar 3. 20 Implementasi *Sequence Diagram* Sertifikat

3.2.3.9 Sequence Diagram Logout

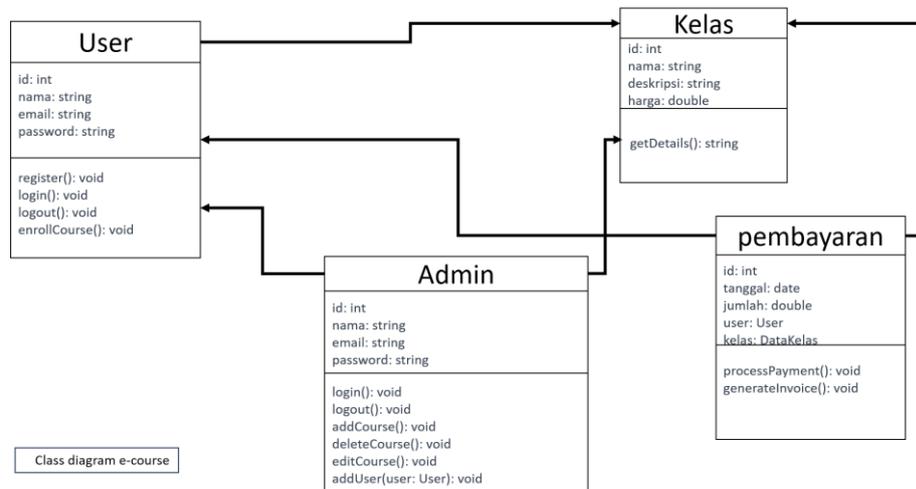
Mahasiswa memilih menu atau tombol *logout*, kemudian sistem memvalidasi jika benar maka diarahkan ke halaman *login*, Jika dalam kesalahan, tetap *login*.



Gambar 3. 21 Implementasi *Sequence Diagram* Logout

3.2.4 Class Diagram

Class diagram tersebut menggambarkan struktur dan hubungan antara entitas dalam sistem *e-course*. Terdapat empat kelas utama, yaitu *user*, *Admin*, Data Kelas, dan Pembayaran. Kelas *user* dan *Admin* memiliki atribut dan metode terkait pengguna dan *administrasi*, sementara Data Kelas menggambarkan informasi tentang kelas yang ditawarkan. Pembayaran menghubungkan pengguna dengan data kelas yang dibeli. Hubungan asosiasi antara kelas-kelas ini menunjukkan interaksi antara pengguna, *admin*, dan kelas dalam sistem *e-course*.



Gambar 3. 22 Implementasi *Class Diagram* Sistem *E-Course*

3.2.5 Desain Basis Data

3.2.5.1 Basis Data User

Tabel *user* pada basis data *e-course* berisi kolom *id*, *nama*, *mahasiswa*, dan *password*. Kolom *id* adalah *primary key* untuk mengidentifikasi pengguna secara unik. Kolom *nama* menyimpan nama pengguna, kolom *mahasiswa* menyimpan alamat mahasiswa pengguna, dan kolom *password* menyimpan kata sandi pengguna. Tabel ini digunakan untuk otentikasi pengguna dan pengelolaan akun dalam sistem *e-course*.

Table 3. 10 Basis Data *User*

Id (primary key)	nama	email	<i>password</i>	<i>role</i>
1	Saeiful Mu'minin	sae@gmail.com	Iful123	<i>Member</i>
2	Saeiful Mu'minin	sae@gmail.com	Iful123	<i>member</i>

3.2.5.2 Basis Data *Admin*

Tabel *Admin* pada basis data *e-course* memiliki kolom id, nama, mahasiswa, dan *password*. Kolom id berfungsi sebagai primary key untuk mengidentifikasi *admin* secara unik, sedangkan kolom nama, mahasiswa, dan *password* menyimpan informasi tentang nama lengkap *admin*, alamat mahasiswa *admin*, dan kata sandi *admin*. Tabel ini digunakan untuk otentikasi *admin* saat *login* dan untuk melakukan tugas *administratif* dalam sistem *e-course*, seperti manajemen data kelas dan pengguna.

Table 3. 11 Basis Data *Admin*

Id (primary key)	nama	email	<i>password</i>	<i>role</i>
1	Saeiful Mu'minin	sae@gmail.com	Iful123	<i>admin</i>
2	Saeiful Mu'minin	sae@gmail.com	Iful123	<i>admin</i>

3.2.5.3 Basis Data Kelas

Tabel data kelas pada basis data *e-course* memiliki kolom id, nama, deskripsi, dan harga. tabel ini digunakan untuk menyimpan informasi tentang kelas yang tersedia, termasuk identifikasi, nama, deskripsi, dan harga kelas.

Table 3. 12 Basis Data Kelas

Id (primary key)	nama	<i>deskripsi</i>	harga	category
1	laravel	Larem adalah	20.000	<i>website</i>
2	laravel	Larem adalah	20.000	<i>website</i>

3.2.5.4 Basis Data Pembayaran

Pembayaran pada basis data *e-course* memiliki kolom id, tanggal, jumlah, *mahasiswa_id*, dan *kelas_id*. Pembayaran yang dilakukan dicatat dalam tabel ini oleh pengguna terhadap kelas yang dibeli. Informasi seperti tanggal pembayaran, jumlah yang dibayarkan, serta kaitannya dengan pengguna dan kelas disimpan dalam tabel ini.

Table 3. 13 Basis Data Transaksi

id	User_id	Kelas_id	tanggal	jumlah	status
1	1	1	20-06-2024	20.000	Sukses
2	1	2	20-08-2024	20.000	Expired
3	2	1	20-07-2024	20.000	pending

3.2.6 Perancangan Antar muka

3.2.6.1 Tampilan Register

Halaman *register* menampilkan *form register* untuk membuat akun dan masuk ke dalam sistem. Formulir pendaftaran yang disediakan memungkinkan pengguna untuk mengisi informasi yang diperlukan, sehingga mereka dapat dengan mudah membuat akun baru.

Gambar 3. 23 Perancangan Halaman *Register*

3.2.6.2 Tampilan Login

Halaman *login* dilengkapi dengan sebuah *form login* yang memungkinkan pengguna untuk masuk ke dalam sistem dengan memberikan informasi kredensial mereka. Formulir ini meminta pengguna untuk memasukkan alamat email beserta kata sandi mereka. Dengan mengisi form

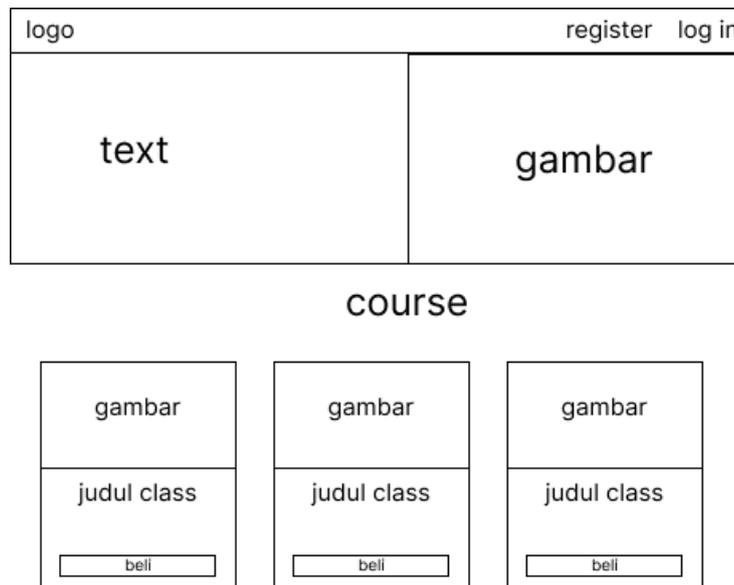
tersebut, pengguna dapat mengakses berbagai fitur dan layanan yang tersedia di dalam sistem dengan mudah dan aman



Gambar 3. 24 Perancangan Halaman *Login*

3.2.6.3 Tampilan *Landing Page*

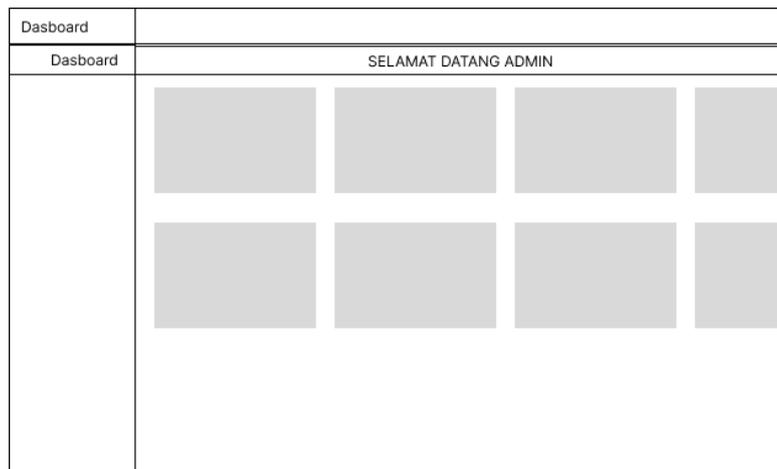
Halaman landing page adalah halaman pertama yang dilihat oleh pengunjung saat mengakses sebuah *website*.



Gambar 3. 25 Perancangan Halaman *Landing Page*

3.2.6.4 Tampilan Halaman *Dashboard*

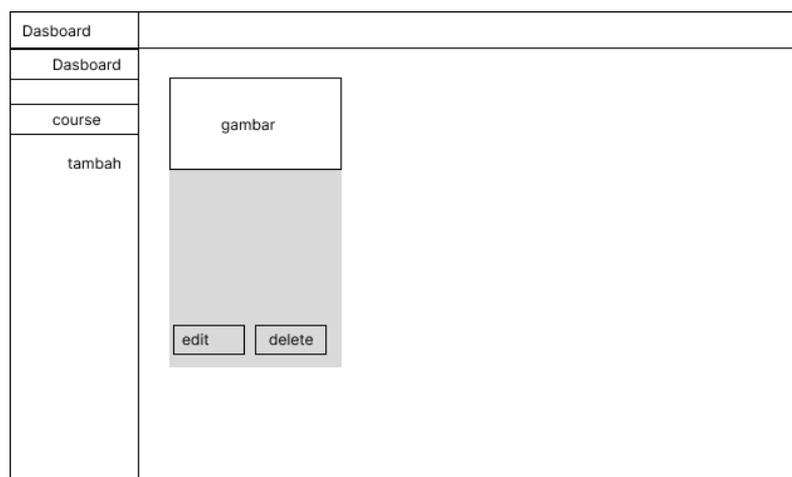
Halaman *dashboard* untuk dengan cepat memuat data tujuan utamanya adalah memberikan informasi yang mudah dipahami dan menggambarkan data dengan cara yang menarik dan interaktif.



Gambar 3. 26 Perancangan Halaman *Dashboard*

3.2.6.5 Tampilan Kelola Kelas

Halaman ini yang bertujuan untuk sebuah tampilan *admin* yang tugasnya adalah menambah, mengedit dan menghapus kelas.



Gambar 3. 27 Perancangan Halaman Kelola Kelas

3.2.6.6 Tampilan Kelola pembayaran

Halaman ini untuk menerima data kirim dari mahasiswa untuk dilakukan validasi pembayaran yang nantinya akan dilakukan proses pembukaan akses kelas.

Dashboard							
Dashboard							
	list transaksi						
course	#	INVOICE	EMAIL	TOTAL	STATUS	DATE	ACTION
transaction							

Gambar 3. 28 Perancangan Halaman Kelola Pembayaran

3.2.6.7 Tampilan Kelola *user*

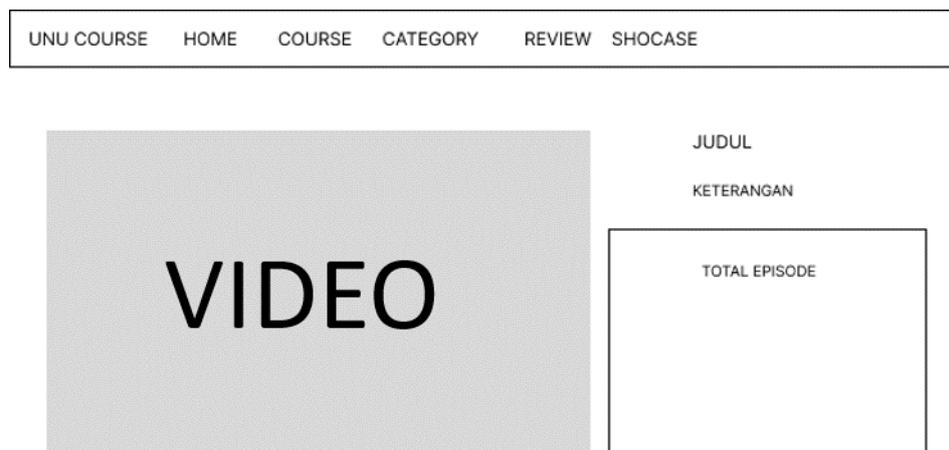
Halaman ini untuk mengelola *user* baik mengedit maupun menghapus data mahasiswa secara akurat dan cepat dalam sebuah tampilan.

Dashboard							
Dashboard							
	list USER						
course	#	INVOICE	EMAIL		ROLE		ACTION
						EDIT	HAPUS
transaction							

Gambar 3. 29 Perancangan Halaman Kelola Mahasiswa

3.2.6.8 Tampilan Halaman *Learning*

Halaman ini didesain untuk memandu proses belajar dari awal hingga akhir, menyediakan langkah-langkah dan informasi yang akan ditampilkan.



Gambar 3. 30 Perancangan Halaman *Learning*

BAB IV

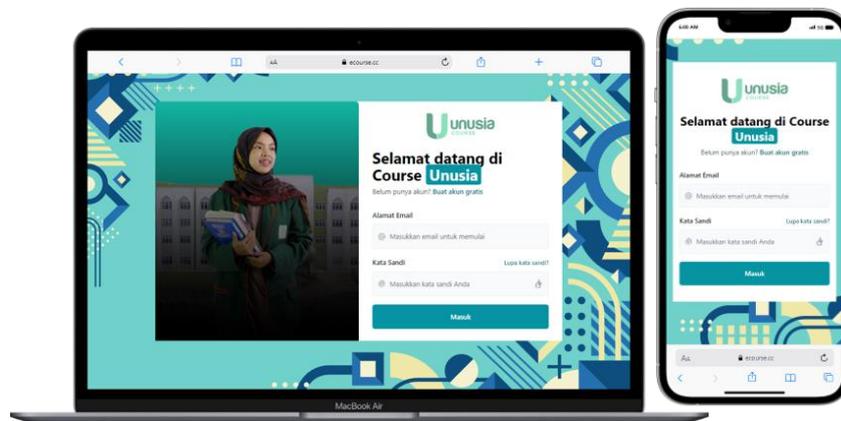
IMPLEMENTASI DAN PENGUJIAN

4.1 Hasil penelitian

Penelitian ini menghasilkan sebuah platform *web* yang dirancang untuk beroperasi melalui halaman browser, dengan menyediakan sejumlah fitur yang bertujuan untuk memenuhi kebutuhan pengguna. Platform ini menawarkan beragam fitur, mulai dari halaman *login* dan pendaftaran, hingga halaman utama, mengelola kelas, pembayaran, konfirmasi pembayaran, akses ke kelas, mengelola mahasiswa, ujian, cetak sertifikat, testing menggunakan *blackbox* dan hasil survai menggunakan *System Usability Scale*. Dalam penjelasan rinci berikut, akan dijabarkan spesifikasi masing-masing fitur yang tersedia:

4.1.1 Tampilan *Login*

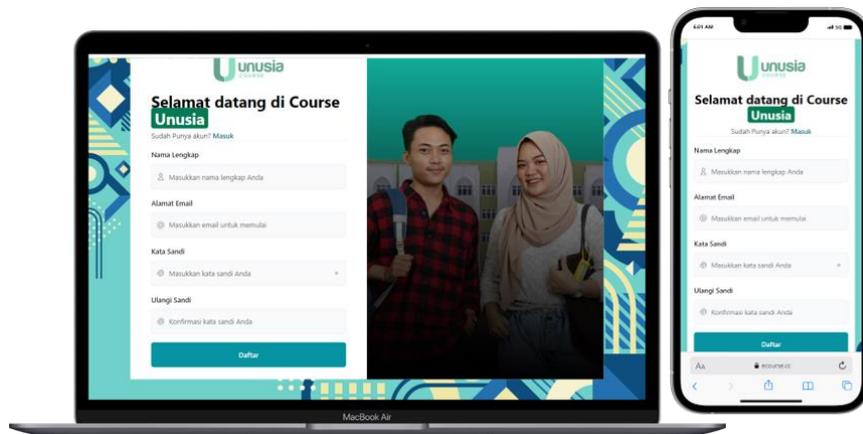
Tampilan *login* adalah antarmuka untuk memasukkan informasi keamanan seperti email dan kata sandi untuk mengakses sistem. Berikut adalah contoh tampilan yang biasanya ditemui saat mengakses sistem:.



Gambar 4. 1 Tampilan *Login*

4.1.2 Tampilan Register

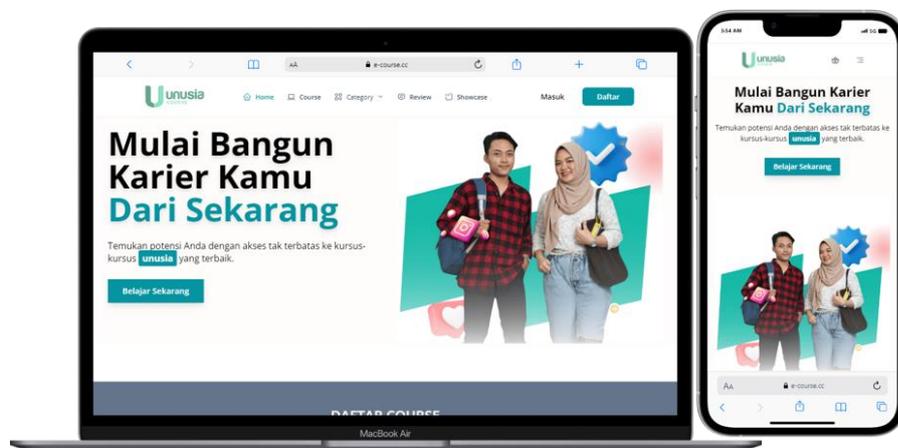
Halaman *registrasi* adalah antarmuka yang memungkinkan pengguna untuk membuat akun baru di dalam sistem. Pengguna diharuskan untuk mengisi formulir dengan informasi yang diperlukan, seperti nama lengkap, alamat email, kata sandi, dan informasi lain yang mungkin diperlukan oleh sistem. Di bawah ini adalah tampilan yang dihasilkan.



Gambar 4. 2 Tampilan *Registrasi*

4.1.3 Tampilan Utama

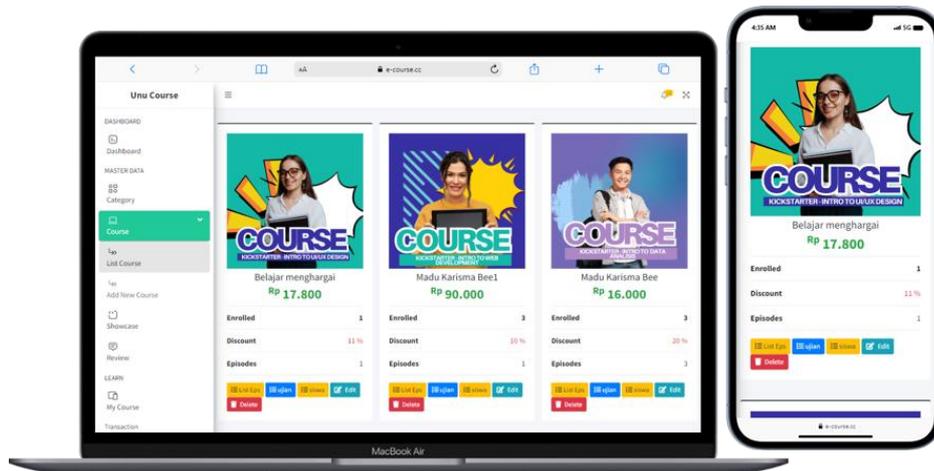
Halaman utama, atau landing page, adalah titik awal yang dilihat oleh pengunjung saat mereka mengakses sistem, memberikan kesan pertama tentang konten dan nilai situs. Berikut adalah tampilannya.



Gambar 4. 3 Tampilan *Home*

4.1.4 Tampilan Mengelola Kelas

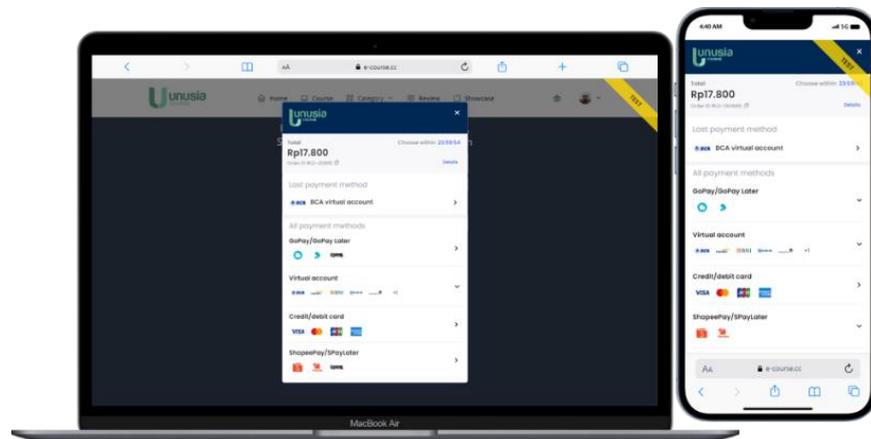
Halaman kelola kelas adalah tempat di situs *web* anda di mana *author* atau *admin* dapat membuat, melihat, memperbarui, dan menghapus kelas-kelas yang ada. Berikut tampilan dari halaman kelola kelas.



Gambar 4. 4 Tampilan Kelola Kelas

4.1.5 Tampilan Pembayaran

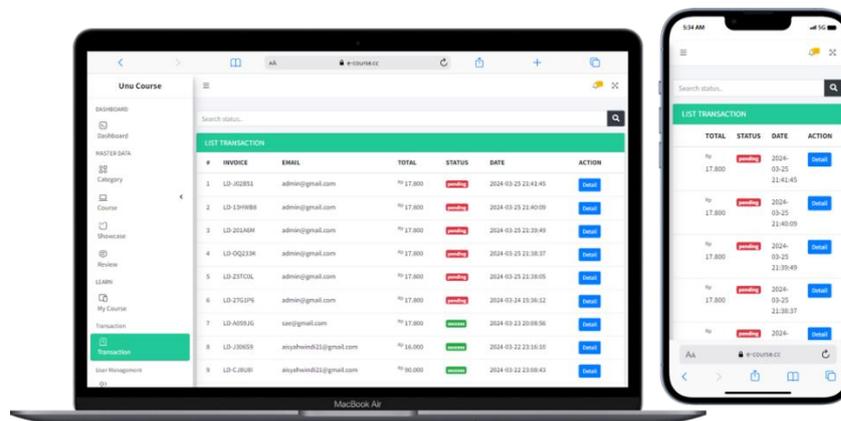
Halaman pembayaran dalam sistem ini memudahkan pengguna untuk membeli kelas dengan aman. Dengan midtrans pengguna dapat melakukan pembelian tanpa kekhawatiran. Setelah pembayaran berhasil, pengguna langsung mendapatkan akses ke kelas yang dibeli. Berikut tampilan pada halaman pembayaran.



Gambar 4. 5 Tampilan Pembayaran

4.1.6 Tampilan Konfirmasi pembayaran

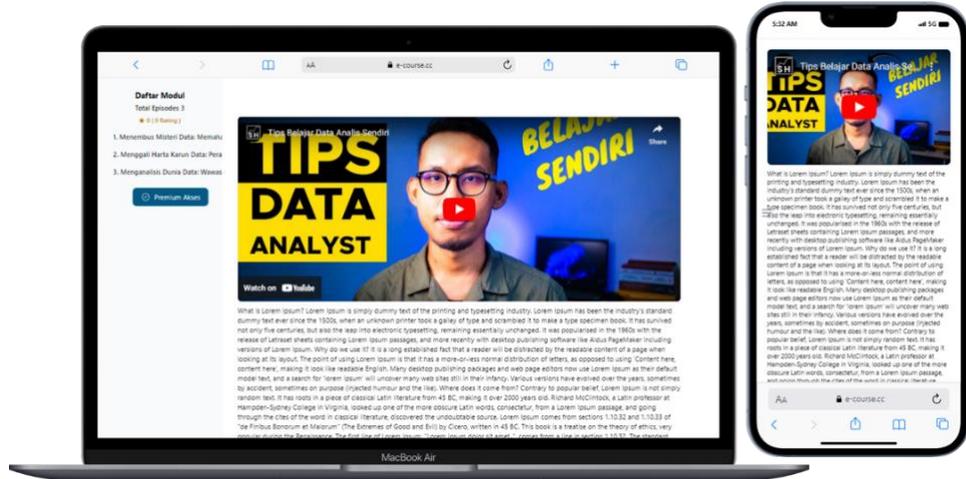
Halaman konfirmasi pembayaran merupakan antarmuka penting yang memuat data transaksi dari pengguna yang tercatat dalam halaman *admin*. Halaman ini memberikan gambaran detail mengenai transaksi yang dilakukan oleh pengguna, termasuk informasi seperti nama pengguna, kelas yang dibeli, jumlah pembayaran, metode pembayaran yang digunakan, serta status pembayaran. Berikut tampilan yang ada pada halaman konfirmasi pembayaran.



Gambar 4. 6 Tampilan Konfirmasi Pembayaran

4.1.7 Tampilan Akses kelas

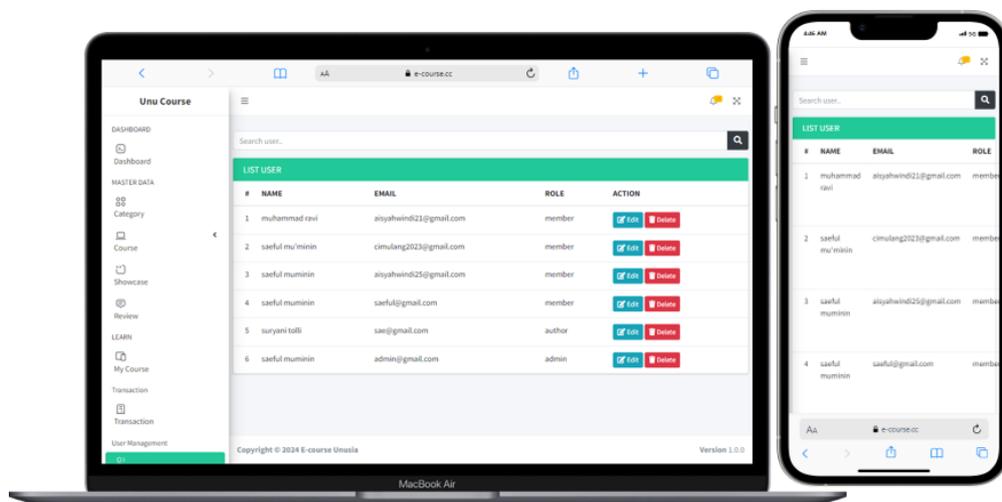
Halaman akses kelas adalah halaman yang memungkinkan pengguna untuk mengakses semua materi pembelajaran yang terkait dengan kelas yang telah dibelinya. Di halaman ini, pengguna akan menemukan semua konten kursus yang tersedia, seperti video pembelajaran, modul bacaan, tugas. Halaman ini biasanya dirancang untuk memberikan navigasi yang intuitif, sehingga pengguna dapat dengan mudah menemukan materi yang mereka butuhkan. Berikut tampilan pada halaman akses kelas.



Gambar 4. 7 Tampilan Akses Kelas

4.1.8 Tampilan Mengelola User

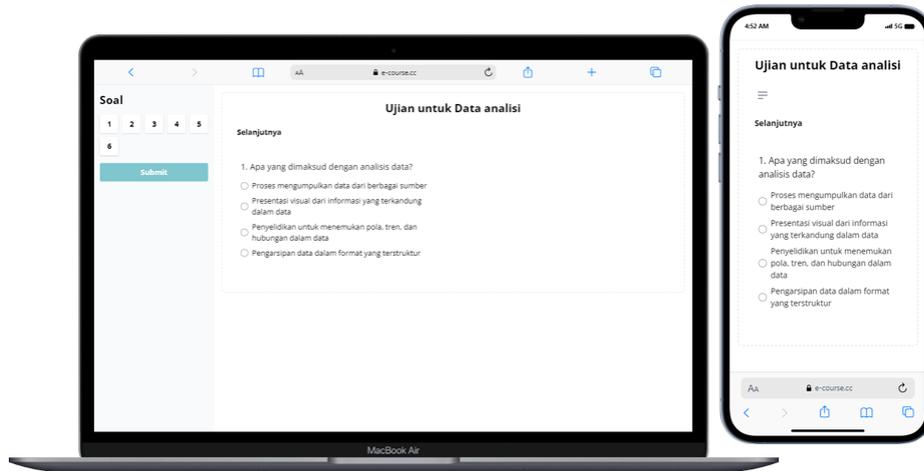
Halaman *admin* dashboard untuk mengelola pengguna adalah area penting di platform. *Admin* dapat mengatur peran pengguna dengan fitur edit role, termasuk mengangkat menjadi *admin* atau *author*. Fitur *Delete* memungkinkan penghapusan pengguna tidak aktif. *Admin* juga dapat melihat detail pribadi dan riwayat aktivitas pengguna. Ini merupakan kontrol pusat bagi *admin* dalam manajemen pengguna di platform. Berikut tampilannya.



Gambar 4. 8 Tampilan Kelola User

4.1.9 Tampilan Ujian kelas

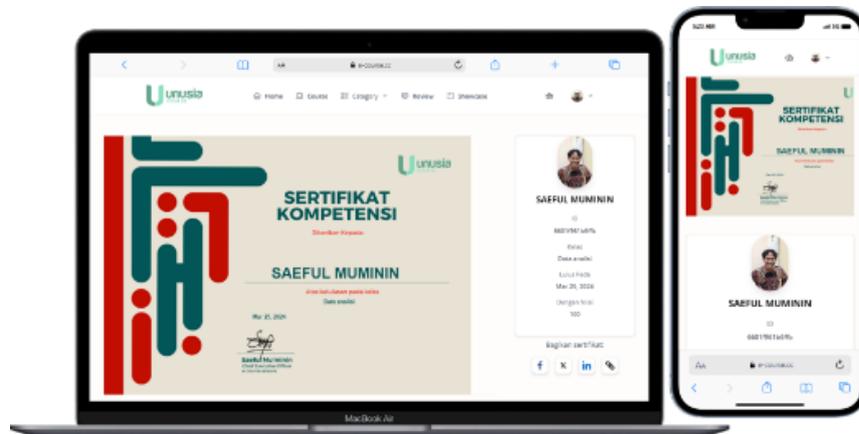
Halaman ujian kelas memungkinkan pengguna mengakses dan menjawab ujian yang telah disiapkan oleh pengajar atau *admin*. Mereka dapat mengirimkan jawaban untuk dinilai sesuai instruksi yang diberikan. Berikut tampilannya.



Gambar 4. 9 Tampilan Ujian

4.1.10 Tampilan Sertifikat

Halaman sertifikat adalah halaman di sistem di mana pengguna, terutama siswa atau peserta kursus, dapat mengakses sertifikat yang diberikan sebagai pengakuan atas keberhasilan menyelesaikan suatu kursus atau program pembelajaran. Berikut tampilan pada halaman sertifikat.



Gambar 4. 10 Tampilan Sertifikat

4.2 Pengujian

Dalam penelitian ini, pengujian fungsionalitas dilakukan dengan tujuan untuk mengidentifikasi kesalahan yang ada dalam sistem, sehingga dapat diperbaiki. Metode yang digunakan dalam pengujian adalah metode *black box*, di mana pengujian dilakukan tanpa memperhatikan struktur internal dari sistem. Hasil pengujian ini dijelaskan secara rinci sebagai berikut.

Table 4. 1 Hasil Pengujian

No	Pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Halaman login	<i>Input email dan password</i>	Masuk ke Dalam sistem dan mempunyai akun.	Sesuai	<i>Valid</i>
2	Halaman register	<i>Input nama lengkap Input email dan password</i>	Masuk ke Dalam sistem dan mempunyai akun.	Sesuai	<i>Valid</i>
3	Tampilan utama	<i>Masuk ke halaman</i>	Menampilkan text dan list kelas	Sesuai	<i>Valid</i>
4	Tampilan mengelola kelas	<i>Menambah mengedit dan menghapus kelas,eps,ujian</i>	Menampilkan data yang diinput dan memasukan ke dalam database	Sesuai	<i>Valid</i>
5	Tampilan pembayaran	<i>membeli kelas yang mau di bayar</i>	Menampilkan box metode pembayaran	Sesuai	<i>Valid</i>

6	Tampilan konfirmasi pembayaran	<i>Menampilkan list pembayaran pengguna</i>	Menampilk semua data transaksi yang sudah sukses dan gagal atau pending	Sesuai	<i>Valid</i>
7	Tampilan akses kelas	<i>Melihat kelas yang sudah di beli dan mendapatkan akses</i>	Menampilkan kelas yang sudah dibeli dan sudah sukses pembayarannya	Sesuai	<i>Valid</i>
8	Tampilan mengelola user	<i>Menampilkan data user yang sudah terdaftar</i>	Menampilkan data dan dapat edit dan hapus user	Sesuai	<i>Valid</i>
9	Tampilan ujian kelas	<i>Menampilkan ujian kelas</i>	Menampilkan ujian yang sudah di input admin atau author	Sesuai	<i>Valid</i>
10	Tampilan sertifikat	<i>Menampilkan sertifikat sesuai dengan kelas</i>	Tampil sertifikat sesuai kelas dan nilai nya	Sesuai	<i>Valid</i>

4.3 Hasil System Usability Scale

Dalam penelitian ini, untuk survai hasil kepuasan pengguna megunakan System Usability Scale ,Tujuan sus adalah untuk memberikan pengembang dan peneliti alat yang efektif untuk mengukur persepsi pengguna terhadap *usability* suatu sistem. Berikut hasil sus ini dijelaskan secara rinci sebagai berikut:

Table 4. 2 Hasil Responden

NO	Responden	Skor Asli									
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R1	5	1	5	1	5	1	5	1	5	1

2	R2	5	1	5	1	5	1	3	2	5	1
3	R3	5	1	5	1	5	1	5	1	5	1
4	R4	5	1	5	1	5	1	5	1	5	1
5	R5	4	2	4	2	4	2	4	2	4	2
6	R6	5	2	4	2	4	1	3	2	5	2
7	R7	5	2	4	2	4	2	4	2	4	2
8	R8	5	1	5	1	5	1	5	1	5	1
9	R9	3	2	3	2	3	3	3	3	3	3
10	R10	4	2	4	2	3	2	4	3	4	3
11	R11	4	2	4	2	4	2	4	2	4	2
12	R12	4	2	4	2	4	2	4	2	4	2
13	R13	4	2	3	2	5	1	4	3	4	2
14	R14	4	2	4	2	4	2	4	2	4	2
15	R15	4	2	4	2	4	2	4	2	4	2
16	R16	4	2	4	2	4	2	4	2	4	2
17	R17	5	3	5	1	5	1	5	1	5	1
18	R18	5	1	5	1	5	1	5	1	5	1
19	R19	5	1	5	1	5	1	5	1	5	1
20	R20	4	2	3	2	3	3	4	2	4	3

Setelah memasukkan nilai dari setiap responden, nilai-nilai tersebut akan diubah menjadi skor sesuai dengan rumus dari *System Usability Scale*. Pada pertanyaan bernomor ganjil, skor dari pengguna akan dikurangi. Sedangkan pada pertanyaan bernomor genap, hasil skor akhir didapatkan dari nilai 5 dikurangi skor yang diberikan oleh pengguna, kemudian jumlahkan dan dikali 2,5. lalu dicari nilai rata-ratanya dengan cara menjumlahkan nilai dan membaginya dengan jumlah responden. Dengan demikian, hasilnya akan disajikan sebagai berikut:

Table 4. 3 Hasil Nilai *System Usability Scale*

Skor Hasil Hitung										JUMLAH	NILAI (JUMLAH X 2,5)
Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	2	3	4	4	37	92,5
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
3	3	3	3	3	3	3	3	3	3	30	75
4	3	3	3	3	4	2	3	4	3	32	80
4	3	3	3	3	3	3	3	3	3	31	77,5

4	4	4	4	4	4	4	4	4	4	40	100
2	3	2	3	2	2	2	2	2	2	22	55
3	3	3	3	2	3	3	2	3	2	27	67,5
3	3	3	3	3	3	3	3	3	3	30	75
3	3	3	3	3	3	3	3	3	3	30	75
3	3	2	2	4	4	3	2	3	3	29	72,5
3	3	3	3	3	3	3	3	3	3	30	75
3	3	3	3	3	3	3	3	3	3	30	75
3	3	3	3	3	3	3	3	3	3	30	75
4	2	4	4	4	4	4	4	4	4	38	95
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
3	3	2	3	2	2	3	3	3	2	26	65
JUMLAH SCORE											1655
JUMLAH SCORE RATA-RATA											82,75

Setelah hasil evaluasi menunjukkan skor 82.875, peneliti dapat dengan yakin menyimpulkan bahwa tingkat usability sistem ini sangat baik, sesuai dengan kriteria yang tercantum dalam tabel System Usability Scale. Skor yang tinggi menandakan bahwa pengguna merasa sistem ini mudah digunakan dan efisien dalam memenuhi kebutuhan mereka.

BAB V

PENUTUP

5.1 Kesimpulan

Hasil penelitian ini menunjukkan bahwa pengembangan sistem *e-course* menggunakan metode RAD dengan framework Laravel 10 memiliki potensi besar dalam meningkatkan aksesibilitas dan kualitas pendidikan tinggi. Implementasi teknologi ini dapat berdampak besar terhadap mahasiswa untuk belajar secara mandiri, fleksibel, dan sesuai dengan kebutuhan mereka, serta mengatasi hambatan geografis dan waktu yang sering terjadi dalam pendidikan tradisional.

Sistem ini menegaskan bahwa penggunaan teknologi dalam pendidikan, khususnya melalui *e-learning* menggunakan sistem *e-course*, memberikan dampak positif yang signifikan. Hasil survei responden dari beberapa mahasiswa menunjukkan kepuasan yang tinggi, mencapai 82,875 dari total 20 responden berdasarkan perhitungan menggunakan metode *System Usability Scale*, yang menyoroti peningkatan aksesibilitas dan kualitas pendidikan tinggi.

Inovasi teknologi pendidikan seperti sistem *e-course* juga membuka peluang baru untuk eksplorasi metode pembelajaran yang lebih interaktif dan terarah. Selain itu, dengan adopsi teknologi yang tepat, lembaga pendidikan dapat menyesuaikan kurikulum mereka dengan lebih baik terhadap perkembangan kebutuhan industri dan pasar kerja. teknologi dalam pendidikan bukan hanya tentang meningkatkan efisiensi belajar, tetapi juga mempersiapkan mahasiswa untuk menghadapi perubahan-perubahan yang dinamis dalam dunia kerja yang semakin kompleks di masa depan.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan, maka penulis memberikan beberapa saran yaitu:

1. Perlu ditambahkan fitur timer di ujian serta validasi nama pada sertifikat dengan *input* nama sesuai KTP sebelum sertifikat di generate. Hal ini akan meningkatkan validitas dan keamanan penggunaan platform.
2. *Website* dapat dikembangkan menjadi aplikasi *mobile* untuk platform Android dan iOS. Ini akan meningkatkan aksesibilitas dan fleksibilitas pengguna dalam mengakses konten pembelajaran.
3. Penelitian selanjutnya baiknya platform video dari YouTube beralih ke platform lain yang lebih eksklusif, dapat meningkatkan pengalaman pengguna dan mengurangi risiko terkait kebijakan privasi dan pembatasan akses. Hal ini juga dapat memberikan kontrol yang lebih besar terhadap konten yang disajikan kepada pengguna.

DAFTAR PUSTAKA

- Ahmadar, M., Perwito, P., & Taufik, C. (2021). PERANCANGAN SISTEM INFORMASI PENJUALAN BERBASIS WEB PADA RAHAYU PHOTO COPY DENGAN DATABASE MySQL. *Dharmakarya*, 10(4), 284. <https://doi.org/10.24198/dharmakarya.v10i4.35873>
- Asmara, J. (2021). Rancang Bangun Aplikasi E-learning Pada SMP N 2 Busalangga. *JSAI (Journal Scientific and Applied Informatics)*, 4(1), 85–94. <https://doi.org/10.36085/jsai.v4i1.1331>
- Atmaja, R. K., Komarudin, I., & Hariyanto. (2021). Konsep MVC Pada Rancang Bangun Aplikasi Sistem Informasi Rekrutmen Karyawan Berbasis Web. *IMTechno: Journal of Industrial ...*, 2(1), 30–36. <https://jurnal.bsi.ac.id/index.php/imtechno/article/view/163>
- Aulia, S. C. I. (2022). Pemanfaatan Uml (Unified Modeling Language) Dalam Perancangan Sistem Informasi Rekam Medis Sederhana Pada Kegiatan Posbindu Ptm. *Jurnal Ilmiah Sains Dan Teknologi*, 6(1), 38–44. <https://doi.org/10.47080/saintek.v6i1.1665>
- Dahri, F., Hanafi, A. M. El, Handoko, D., & Wulan, N. (2022). Implementation of Microservices Architecture in Learning Management System E-Course Using Web Service Method. *Sinkron*, 7(1), 76–82. <https://doi.org/10.33395/sinkron.v7i1.11229>
- Damayanti, C., Triayudi, A., & Sholihati, I. D. (2022). Analisis UI/UX Untuk Perancangan Website Apotek dengan Metode Human Centered Design dan System Usability Scale. *Jurnal Media Informatika Budidarma*, 6(1), 551.

<https://doi.org/10.30865/mib.v6i1.3526>

Febriyanti, N. M. D., Sudana, A. A. K. O., & Piarsa, I. N. (2021). Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen. *Jurnal Ilmiah Teknologi Dan Komputer*, 2(3), 1–10.

Harahap, M. M. I., Septama, H. D., & Komarudin, M. (2022). Pengembangan Sistem Agenda Pimpinan Universitas Lampung Menggunakan Framework Laravel. *Jurnal Informatika Dan Teknik Elektro Terapan*, 10(3), 193–200. <https://doi.org/10.23960/jitet.v10i3.2650>

Hardianty, D. A., Yustiana, I., & Somantri. (2022). Rancang Bangun Aplikasi E-Learning Berbasis Progressive Web Apps Untuk Menunjang Pembelajaran Online dengan Metode Prototyping. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 6(2), 754–765.

Hidayat, N., & Hati, K. (2021). Penerapan Metode Rapid Application Development (RAD) dalam Rancang Bangun Sistem Informasi Rapor Online (SIRALINE). *Jurnal Sistem Informasi*, 10(1), 8–17. <https://doi.org/10.51998/jsi.v10i1.352>

Indrawan, I. P. Y., & Nugraha, putu gede surya. (2020). *Rancangan dan Implementasi Sistem E-Learning Berbasis Web*. 3(3), 367–374.

Lesmana, T., & Silalahi, M. (2022). Rancang Bangun Sistem Informasi E-Learning Berbasis Web Pada TPQ AL-Maarij. *Jurnal Comasie*, 6(4), 11–19.

Mubarak, A. (2019). Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek. *JIKO (Jurnal Informatika Dan Komputer)*, 2(1), 19–25. <https://doi.org/10.33387/jiko.v2i1.1052>

Mulyanto, Y., Handani, F., & Hasmawati. (2020). Rancang Bangun Sistem

- Informasi Penjualan Pada Toko Omg Berbasis. *Jinteks*, 2(1), 69–77.
- Novitasari, Y. S., Adrian, Q. J., & Kurnia, W. (2021). Rancang Bangun Sistem Informasi Media Pembelajaran Berbasis Website (Studi Kasus: Bimbingan Belajar De Potlood). *Jurnal Teknologi Dan Sistem Informasi (JTISI)*, 2(3), 136–147. <http://jim.teknokrat.ac.id/index.php/JTISI>
- Priamnistiko, A., Handrianto, Y., & Sukmana, H. S. (2021). Rancang Bangun Sistem Informasi E-Learning Menggunakan Model Waterfall. *Jurnal Inovasi Informatika*, 6(1), 50–57. <https://doi.org/10.51170/jii.v6i1.143>
- Rahmadi, I. F., & Hayati, E. (2020). Literasi Digital, Massive Open Online Courses, dan Kecakapan Belajar Abad 21 Mahasiswa Generasi Milenial. *Jurnal Studi Komunikasi Dan Media*, 24(1), 91. <https://doi.org/10.31445/jskm.2020.2486>
- Reza Fahlevi, M., Muminin, S., Sutiono, Mega Fitria, A., Anisa, Fildzah, Indah Ulfiah, C., & Fathurrohman, A. (2023). *Sistem Informasi Berbasis Web Publisitas Kelompok Wanita Tani dan Desa Wisara Situ Lebak Wangi di Desa Pamegarsari*. 2, 197–205.
- Rina Noviana. (2022). Pembuatan Aplikasi Penjualan Berbasis Web Monja Store Menggunakan Php Dan Mysql. *Jurnal Teknik Dan Science*, 1(2), 112–124. <https://doi.org/10.56127/jts.v1i2.128>
- Riskiono, S. D., & Pasha, D. (2020). *Analisis Perbandingan Server Load Balancing dengan Haproxy & Nginx dalam Mendukung Kinerja Server E- Learning*. 10(2), 135–144.
- Subiksa, B. G., Peling, I. B. A., & Ariawan, M. P. agus. (2023). *Pengembangan CMS (Content Management System) dalam Pembuatan Website Jurusan Menggunakan Framework Laravel*. 11(4), 1019–1026.

Tedyyana, A., Ratnawati, F., & Kurniati, R. (2019). Rancangan Sistem Informasi Penelitian Dan Pengabdian Masyarakat Politeknik Negeri Bengkalis Menggunakan Metode Uml (Unified Modeling Language). *Sistemasi*, 8(3), 413. <https://doi.org/10.32520/stmsi.v8i3.535>

LAMPIRAN

Berikut terlampir source code dan data pertanyaan kuesioner yang digunakan dalam pengembangan *website e-course* untuk Universitas Nahdlatul Ulama Indonesia. Source code ini meliputi langkah-langkah dari perancangan hingga implementasi dalam pengkodean. Semoga lampiran ini memberikan gambaran yang lebih jelas tentang implementasi *website e-course* yang telah dibahas dalam penelitian ini. Jika Anda memiliki pertanyaan atau memerlukan bantuan dalam memahami atau memperbaiki source code tersebut, jangan ragu untuk bertanya. Peneliti siap membantu Anda untuk membuatnya menjadi lebih baik. Ini link *websitenya* <https://ecourse.syahravi.my.id/>

Dan ini link source codenya

<https://github.com/saefulmuminin/e-course.git>

Deskripsi:

1. *website* ini berjalan disemua platform browsers dan untuk aktif di local harus install beberapa persyaratan seperti php , composer , npm dan mysql untuk databasenya. <https://www.phpmyadmin.net/> , <https://www.php.net/downloads> untuk instal php, <https://getcomposer.org/download/> untuk install composer. Dan ini untuk install npm (node package menager) <https://nodejs.org/en/download> .
- 2 kalo sudah di visual code install beberapa extension wajib.
 - a. *Laravel Extra Intellisense*
 - b. *Laravel Blade formatter*

- c. *PHP Intelephense*
 - d. *PostCSS Language Support*
4. setelah sudah terinstall maka lakukan ini di terminal .
- a. Buat database di mysql
 - b. Actikan server database.
 - c. copy file env.example lalu tempel dan ubabh menjadi env .
 - d. masuk ke file env dan sesuaikan nama databasenya.
 - e. Npm install.
 - f. *Composer install.*
 - g. *Php artisan migrate –seed.*
 - h. *php artisan key:generate.*
 - i. *php artisan storage:link.*
 - j. *php artisan serve.*
5. Setelah masuk ke websitenya silahkan *login* dengan email dan *password* untuk menentukan role.
6. Cara mengoperasikan *admin*, *author* dan *user* baik di local maupun di hosting.
- a. *Admin*
 - a) Setelah masuk ke dalam website bisa langsung *login* dengan email dan *password* yang sudah ditentukan.
 - b) Setelah masuk bagian pojok atas bisa liat dashboard maka disitu sudah banyak menu yang terkhusus buat *admin*.
 - b. *Author*

- c) Setelah masuk ke dalam website bisa langsung bisa *registrasi* dulu dan melakukan penginputan *registrasi*
- d) Setelah *registrasi* ketika liat di menunya masih menggunakan menu *user* untuk mengganti ke menu *author* caranya minta persetujuan dari *admin* untuk mengubah rolenya dari member ke *author* di halaman dashboard kelola *user*
- e) Setelah diubah maka akan ada tampilan baru yaitu tambah kelas

c. *User*

- f) Setelah masuk ke dalam website bisa langsung bisa *registrasi* dulu dan melakukan penginputan *registrasi*
- g) Setelah *registrasi* ketika liat di menunya disitu banyak sekali menu untuk *user* salah satunya menu transaksi untuk transaksi kita yang sudah dilakukan.
- h) Untuk membeli kelasnya bisa dilihat di menu awal di bagian atas ada course pilih salah satu course dan click maka akan menuju halaman detail dari kelas disitu ada tombol beli.
- i) Ketika sudah di beli maka total pembayaran masuk ke keranjang di keranjang bisa menghapus kelas , selanjutnya klik bayar dan kemudian diarahakan di halaman pembayaran dengan nominal tertentu click bayar sekarang dan kemudian muncul metode pembayaran yang ingin digunakan ketika sudah bisa melakukan pembayaran hanya diberi waktu 24 jam lebih dari itu hangus pembayarannya.

Berikut ini adalah data pertanyaan 1 sampai 10 yang digunakan dalam survei menggunakan SUS (*System Usability Scale*) untuk *website e-course* Universitas Nahdlatul Ulama Indonesia:

No	Pertanyaan
Q1	Saya merasa <i>website e-course</i> ini mudah untuk digunakan.
Q2	Saya merasa sistem interaksi di <i>website e-course</i> ini cukup sederhana.
Q3	Saya merasa ada banyak fitur di <i>website e-course</i> ini yang mudah diingat.
Q4	Saya merasa <i>website e-course</i> ini membutuhkan waktu yang sesuai untuk menyelesaikan tugas yang diberikan.
Q5	Saya merasa mudah memahami bagaimana cara menggunakan <i>website e-course</i> ini.
Q6	Saya merasa ada kesalahan yang sering terjadi saat menggunakan <i>website e-course</i> ini.
Q7	Saya merasa puas dengan kemampuan <i>website e-course</i> ini untuk memenuhi kebutuhan saya.
Q8	Saya merasa mudah untuk belajar menggunakan <i>website e-course</i> ini.
Q9	Saya merasa kesulitan saat menggunakan <i>website e-course</i> ini.
Q10	Saya merasa percaya bahwa saya dapat menggunakan <i>website e-course</i> ini dengan baik.

Hasilnya dari perhitungannya dapat diakses di link ini <https://docs.google.com/spreadsheets/d/1DN7ZspWWM31VJ9jaH08e7JQrPTSSzZx1Ihj17Cnd7tc/edit?usp=sharing>