

## **BAB IV**

### **HASIL ANALISIS DAN PENELITIAN**

#### **4.1 Analisa Perancangan Sistem**

Perancangan sistem ini didasarkan pada identifikasi masalah yang dihadapi oleh pengguna, yang telah dianalisis sebelumnya pada Bab III. Dalam penelitian ini, peneliti mengusulkan solusi berbasis sistem informasi untuk mengatasi berbagai kendala yang ada. Salah satu masalah utama yang teridentifikasi adalah kurangnya sistem yang terintegrasi dalam proses manajemen kelas dan transaksi, yang menyebabkan ketidakefisienan dalam pengelolaan data.

Sistem yang diusulkan akan mencakup modul-modul utama seperti manajemen kelas, pengelolaan transaksi, serta pelacakan dan pengelolaan booking. Dengan perancangan sistem ini, diharapkan dapat mempermudah pengelolaan jadwal kelas, transaksi pembayaran, dan pengelolaan data pengguna secara lebih efisien. Sistem ini juga akan menyediakan fitur pelaporan dan analisis yang berguna bagi administrator dan instruktur dalam melakukan evaluasi kinerja dan transaksi yang terjadi. Dengan pendekatan ini, sistem informasi yang diusulkan diharapkan dapat memberikan solusi yang efektif dan terintegrasi, tidak hanya menyelesaikan masalah operasional, Namun, juga dapat membantu dalam pengambilan keputusan yang lebih tepat dalam pengelolaan kelas dan transaksi.

#### **4.2 Requirements Planning**

Pada tahap awal perancangan diagram sistem, peneliti menerapkan OOAD (Object-Oriented Analysis and Design) dengan UML (Unified Modeling Language) sebagai metode perancangan. Pendekatan ini dipilih karena OOAD memberikan fleksibilitas dan skalabilitas dalam merancang sistem berbasis objek, sementara UML menawarkan notasi grafis yang jelas dan terstruktur untuk menggambarkan berbagai aspek sistem.

Pada tahap awal ini, diagram yang digunakan meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram, di mana masing-masing diagram memiliki peran penting dalam menggambarkan kebutuhan sistem serta interaksi antar komponen dalam sistem yang diusulkan. Dengan

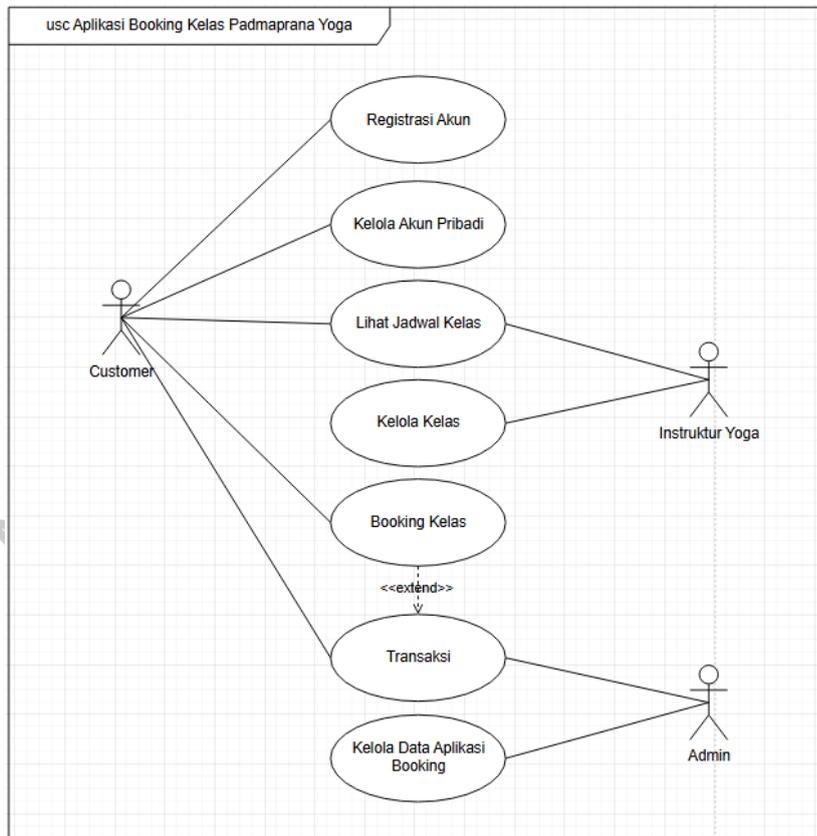
perancangan diagram ini, peneliti dapat merinci cara kerja sistem, interaksi pengguna dengan sistem, serta bagaimana data akan diproses dan disajikan.

#### **4.2.1 Use Case Diagram ( Iterasi 1)**

Use Case Diagram merupakan alat yang sangat efektif dalam menggambarkan interaksi antara pengguna dan sistem pada aplikasi pengelolaan sistem booking kelas yoga. Diagram ini memberikan ilustrasi tentang bagaimana aktor utama, seperti Customer, Instruktur, dan Admin, berinteraksi dengan berbagai fungsi yang ada dalam sistem, seperti pemesanan kelas, pengecekan jadwal, dan pengelolaan data kelas.

Dalam konteks sistem ini, Use Case Diagram akan memperlihatkan berbagai fungsi utama yang diperlukan, seperti Melihat Daftar Kelas, Mendaftar Kelas Yoga, Membayar Biaya Kelas, dan Melihat Riwayat Booking. Diagram ini juga memberikan gambaran yang jelas mengenai bagaimana sistem merespons permintaan dari masing-masing aktor tersebut. Dengan adanya diagram ini, para pengembang dan pemangku kepentingan dapat dengan mudah memahami bagaimana interaksi antar entitas dalam sistem. Berikut merupakan hasil analisa kebutuhan User yang berasal dari identifikasi dari elisitasi final :

- A. Registrasi Akun
- B. Kelola Akun Pribadi
- C. Lihat Jadwal Kelas
- D. Kelola Kelas
- E. Booking Kelas
- F. Transaksi
- G. Kelola Data Aplikasi Booking



Gambar 4. 1 Use Case Diagram Usulan

#### 4.2.2 Spesifikasi Use Case Diagram ( Iterasi 1)

Spesifikasi Use Case Diagram akan merinci lebih dalam setiap fungsionalitas yang telah digambarkan dalam Use Case Diagram sebelumnya. Misalnya, untuk use case Mendaftar Kelas Yoga, spesifikasi akan mencakup langkah-langkah seperti memilih kelas, memilih jadwal yang sesuai, melakukan pembayaran, dan menerima konfirmasi pendaftaran. Dokumen spesifikasi ini akan digunakan sebagai dasar untuk perancangan diagram lainnya, seperti Activity Diagram dan Sequence Diagram, yang akan membantu memetakan alur proses secara lebih detail dan memastikan bahwa sistem berjalan dengan baik dan sesuai dengan kebutuhan pengguna.

Tabel 4. 1 Spesifikasi Use Case Registrasi Akun

<b>Use Case Name</b>	Use Case Registrasi Akun
<b>Actor</b>	Customer

<b>Deskripsi</b>	Pengguna baru dapat membuat akun untuk mengakses aplikasi. Proses registrasi mencakup pengisian data pribadi seperti nama, email, kata sandi, dan informasi lainnya yang relevan.	
<b>Pre-Condition</b>	Customer ingin mendaftar akun baru	
<b>Post-Condition</b>	Customer berhasil memiliki akun yang terdaftar di Sistem	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Customer membuka halaman registrasi.	2.Menampilkan halaman registrasi
	3.Customer mengisi data seperti nama, email, dan kata sandi.	
	4.Customer menekan tombol "Daftar".	5.Melakukan verifikasi data
		6.Sistem menyimpan data customer baru.
	7.Customer berhasil membuat akun	8.Sistem menampilkan notifikasi
<b>Alternative Flows</b>	—	

Tabel di atas merupakan spesifikasi dari use case Registrasi Akun. Use case ini memungkinkan pengguna baru, yaitu Customer, untuk membuat akun agar dapat mengakses aplikasi. Proses registrasi dimulai dengan Customer membuka halaman registrasi, dilanjutkan dengan mengisi data pribadi seperti nama, email, dan kata sandi. Setelah data diisi, Customer menekan tombol "Daftar," dan sistem akan melakukan verifikasi serta menyimpan data pengguna baru.

Proses registrasi diakhiri dengan sistem menampilkan notifikasi bahwa registrasi berhasil, yang menandakan Customer telah memiliki akun terdaftar.

Tidak terdapat alternative flows dalam skenario ini, sehingga alur registrasi berlangsung secara langsung dan sederhana.

Tabel 4. 2 Spesifikasi Use Case Kelola Akun Pribadi

<b>Use Case Name</b>	Use Case Kelola Akun Pribadi	
<b>Actor</b>	Customer	
<b>Deskripsi</b>	Pengguna dapat melihat dan memperbarui informasi akun mereka, seperti nama, email, kata sandi, atau informasi lainnya.	
<b>Pre-Condition</b>	Customer sudah terdaftar dan login	
<b>Post-Condition</b>	Informasi akun customer berhasil diperbaharui	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Customer membuka halaman kelola Akun Pribadi.	2.Sistem Menampilkan halaman profil
	3.Customer mengubah informasi yang diperlukan.	
	4.Customer menekan tombol "Simpan".	5.Sistem menyimpan data customer
	6.Customer berhasil mengubah profil	7.Sistem menampilkan pesan bahwa perubahan
<b>Alternative Flows</b>	-	

Tabel di atas merupakan spesifikasi dari use case Kelola Akun Pribadi. Use case ini memungkinkan Customer yang telah terdaftar dan login untuk melihat serta memperbarui informasi akun mereka, seperti nama, email, kata sandi, atau informasi lainnya. Proses ini diawali dengan Customer membuka halaman Kelola Akun Pribadi, di mana sistem akan menampilkan halaman profil pengguna.

Selanjutnya, *Customer* dapat mengubah informasi yang diperlukan dan menekan tombol "Simpan." Sistem kemudian akan menyimpan data yang

diperbarui dan menampilkan pesan konfirmasi bahwa perubahan berhasil disimpan. Alur ini tidak memiliki *alternative flows*, sehingga proses mengelola akun berlangsung langsung tanpa adanya variasi skenario lain.

Tabel 4. 3 Spesifikasi Use Case Lihat Jadwal Kelas

<b>Use Case Name</b>	Use case lihat jadwal kelas	
<b>Actor</b>	Customer, Instruktur Yoga	
<b>Deskripsi</b>	Pengguna dapat melihat jadwal kelas yang tersedia, termasuk informasi seperti nama kelas, hari, jam, instruktur, dan deskripsi.	
<b>Pre-Condition</b>	Jadwal kelas telah diatur di dalam sistem	
<b>Post-Condition</b>	Jadwal kelas ditampilkan kepada Instruktur Yoga atau Customer	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1. Aktor membuka halaman jadwal kelas.	2. Menampilkan halaman jadwal
<b>Alternative Flows</b>	-	

Tabel di atas merupakan spesifikasi dari use case Lihat Jadwal Kelas. Use case ini memungkinkan baik Customer maupun Instruktur Yoga untuk melihat jadwal kelas yang tersedia. Jadwal kelas mencakup informasi penting seperti nama kelas, hari, jam, instruktur, dan deskripsi.

Proses dimulai dengan akses aktor ke halaman jadwal kelas, kemudian sistem menampilkan halaman tersebut. Setelah itu, Customer atau Instruktur Yoga dapat melihat informasi mengenai jadwal kelas yang sudah diatur dalam sistem. Tabel ini tidak menyebutkan adanya alternative flows, sehingga skenario utama untuk melihat jadwal kelas berjalan sesuai alur yang telah ditentukan.

Tabel 4. 4 Spesifikasi Use Case Kelola Kelas

<b>Use Case Name</b>	Use case kelola kelas	
<b>Actor</b>	Instruktur Yoga	
<b>Deskripsi</b>	Admin dapat mengelola data kelas, seperti menambah, mengedit, atau menghapus jadwal kelas.	
<b>Pre-Condition</b>	Instruktur Yoga memiliki hak akses pengelolaan kelas	
<b>Post-Condition</b>	Jadwal dan kuota kelas berhasil di perbarui di sistem	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Instruktur membuka halaman pengelolaan kelas.	2.Menampilkan halaman kelola kelas
	3.Instruktur memilih kelas yang ingin di kelola	4.Menampilkan detail kelas
	5.Instruktur memasukan informasi jadwal atau kuota baru	
	6.Instruktur menekan tombol "Simpan"	7.Sistem Menyimpan data informasi
		8.Sistem menampilkan notifikasi berhasil menyimpan informasi
<b>Alternative Flows</b>	-	

Tabel di atas merupakan spesifikasi dari use case Kelola Kelas. Use case ini memungkinkan Instruktur Yoga yang memiliki hak akses untuk mengelola data kelas, termasuk menambah, mengedit, atau menghapus jadwal kelas. Proses dimulai dengan Instruktur Yoga membuka halaman pengelolaan kelas, di mana sistem akan menampilkan halaman kelola kelas.

Setelah itu, Instruktur Yoga memilih kelas yang akan dikelola, dan

sistem menampilkan detail kelas tersebut. Instruktur Yoga kemudian memasukkan informasi jadwal atau kuota baru, lalu mengklik tombol "Simpan." Sistem akan menyimpan data yang diperbarui dan menampilkan notifikasi yang mengonfirmasi bahwa perubahan telah berhasil disimpan. Tabel ini tidak mencatat adanya alternative flows, sehingga alur pengelolaan kelas berlangsung sesuai dengan langkah yang sudah ditentukan.

Tabel 4. 5 Spesifikasi Use Case Booking Kelas

<b>Use Case Name</b>	Use Case Booking Kelas	
<b>Actor</b>	Customer	
<b>Deskripsi</b>	Pengguna dapat memesan kelas berdasarkan jadwal yang tersedia.	
<b>Pre-Condition</b>	Customer sudah login dan kelas tersedia	
<b>Post-Condition</b>	kelas berhasil di booking oleh customer	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Customer membuka halaman kelas yang ingin diboeking.	2.Menampilkan halaman booking kelas
	3.Customer memilih kelas yang diinginkan.	4.Menampilkan detail kelas yang dipilih
	5.Customer melakukan pemBookingan kelas	6.Menyimpan Data Booking
		6.Menyimpan Data Booking
<b>Alternative Flows</b>	-	

Tabel di atas merupakan spesifikasi dari use case Booking Kelas. Use case ini memungkinkan Customer untuk memesan kelas berdasarkan jadwal yang tersedia. Proses dimulai dengan Customer membuka halaman kelas yang ingin diboeking, dan sistem akan menampilkan halaman booking kelas.

Selanjutnya, Customer memilih kelas yang diinginkan, dan sistem akan

menampilkan rincian kelas tersebut. Setelah itu, Customer melakukan pemesanan kelas, dan sistem akan menyimpan data pemesanan yang dilakukan. Proses berakhir dengan sistem menampilkan notifikasi yang mengonfirmasi bahwa pemesanan berhasil dilakukan. Tabel ini tidak mencantumkan alternatif alur, sehingga proses pemesanan kelas berjalan sesuai dengan alur utama yang telah ditetapkan.

Tabel 4. 6 Spesifikasi Use Case Transaksi

<b>Use Case Name</b>	Use Case Transaksi	
<b>Actor</b>	Customer, Admin	
<b>Deskripsi</b>	Proses pembayaran yang dilakukan pengguna setelah memesan kelas.	
<b>Pre-Condition</b>	Customer telah melakukan pemesanan kelas	
<b>Post-Condition</b>	Pembayaran di konfirmasi dan status transaksi diperbaharui	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Customer membuka halaman transaksi.	2.Menampilkan Halaman Transaksi
	3.Customer memilih kelas yang ingin di bayar	4.Memampilkan detail transaksi
	5.Customer Memilih metode pembayaran	6.Menampilkan detail pembayaran
	7.Customer Membayar	8.Menampilkan notifikasi pembayara
	9.Admin melakukan verifikasi pembayaran	10.Status Pembayaran diperbaharui
<b>Alternative Flows</b>	-	

Tabel di atas menggambarkan spesifikasi dari use case Transaksi. Use case ini menjelaskan proses pembayaran yang dilakukan oleh Customer setelah pemesanan kelas. Proses dimulai ketika Customer membuka halaman

transaksi, diikuti oleh sistem yang menampilkan halaman transaksi tersebut.

Setelah itu *customer* memilih kelas yang ingin dibayar, dan sistem akan menampilkan detail transaksi tersebut. *Customer* kemudian memilih metode pembayaran yang diinginkan, dan sistem akan memberikan detail pembayaran. Setelah itu, *Customer* melakukan pembayaran, dan sistem menampilkan notifikasi bahwa pembayaran berhasil.

Setelah pembayaran dilakukan, *Admin* akan melakukan verifikasi pembayaran, dan sistem akan memperbarui status pembayaran. Terdapat *alternative flow* yang menjelaskan bahwa jika *Customer* tidak melakukan pembayaran dalam waktu yang ditentukan, maka pembayaran akan dibatalkan.

Tabel 4. 7 Spesifikasi Use Case Kelola Data Aplikasi Booking

<b>Use Case Name</b>	Use Case Kelola Data Aplikasi Booking	
<b>Actor</b>	Admin	
<b>Deskripsi</b>	Admin dapat mengelola data yang berkaitan dengan sistem booking, seperti daftar pengguna, laporan pemesanan, dan status transaksi.	
<b>Pre-Condition</b>	admin memiliki hak akses ke sistem	
<b>Post-Condition</b>	data booking, kelas, atau pengguna diperbarui di sistem	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Admin membuka halaman pengelolaan data aplikasi.	2.Menampilkan halaman kelola data
	3.Admin memilih data yang ingin dikelola (kelas, booking, atau pengguna).	4.Menampilkan detail halaman data
	5.Admin memperbaharui informasi atau data	

	6.Admin menekan tombol "Simpan"	7.Sistem menyimpan informasi atau data
		8.Notifikasi data berhasil di perbaharui
<b>Alternative Flows</b>	-	

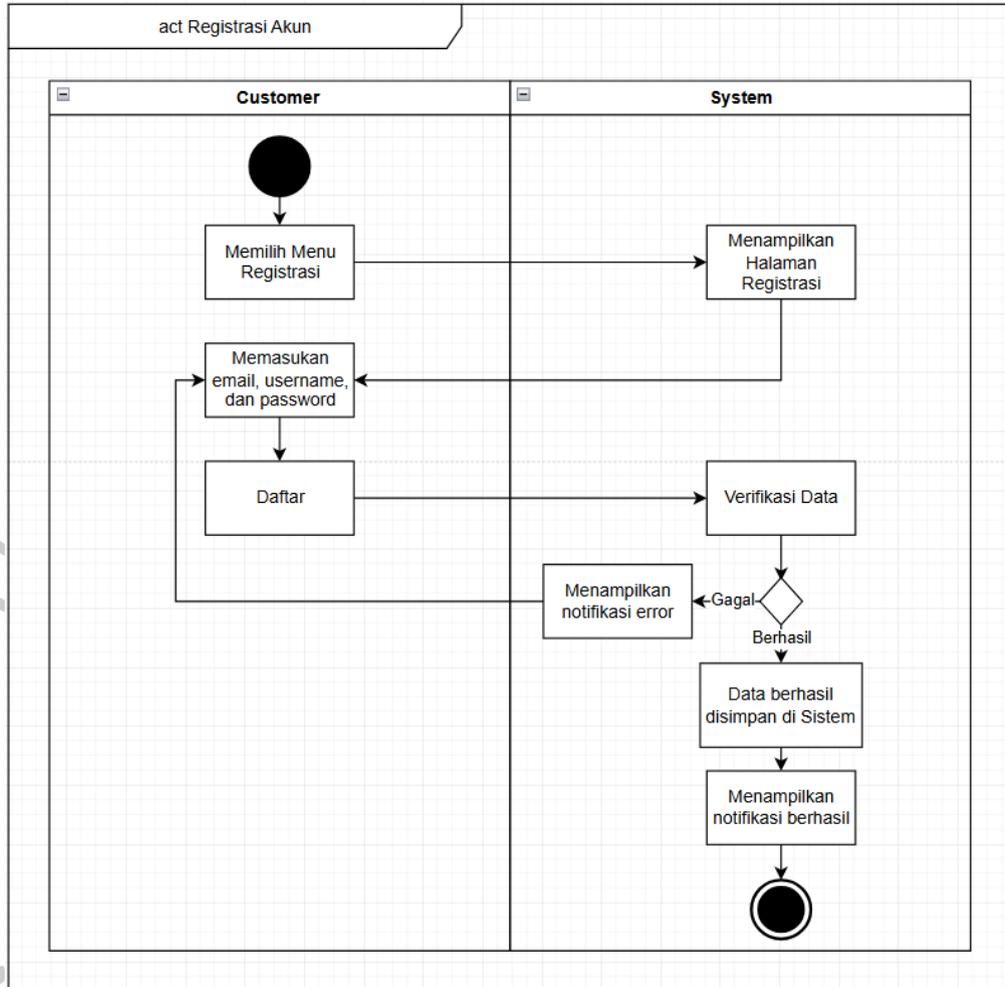
Tabel di atas merupakan spesifikasi dari use case *Kelola Data Aplikasi Booking*. Use case ini memungkinkan *Admin* untuk mengelola data yang berkaitan dengan sistem booking, seperti daftar pengguna, laporan pemesanan, dan status transaksi. Proses dimulai ketika Admin mengakses halaman pengelolaan data aplikasi, dibantu oleh sistem yang menampilkan halaman untuk mengelola data tersebut.

Setelah itu, Admin memilih data yang ingin dikelola, seperti kelas, pemesanan, atau pengguna. Sistem kemudian menampilkan rincian halaman dari data yang dipilih. Admin dapat memperbarui informasi yang ada, lalu menekan tombol "Simpan." Setelah itu, sistem akan menyimpan data yang diperbarui serta menampilkan notifikasi yang mengonfirmasi bahwa perubahan sudah berhasil disimpan. Tabel ini tidak mencantumkan alternatif alur, sehingga proses pengelolaan data aplikasi berjalan sesuai dengan langkah-langkah utama yang telah ditetapkan.

#### 4.2.3 Activity Diagram ( Iterasi 1)

Activity Diagram digunakan untuk menggambarkan alur aktivitas atau proses yang terjadi dalam sistem, baik dari sisi pengguna maupun sistem. Dalam aplikasi pengelolaan sistem booking kelas yoga, Activity Diagram akan digunakan untuk memetakan langkah-langkah yang diambil oleh pengguna saat mereka berinteraksi dengan aplikasi, seperti Mendaftar Kelas, Melakukan Pembayaran, dan Melihat Jadwal Kelas. Diagram ini akan menggambarkan urutan kegiatan yang harus dilakukan, termasuk kondisi keputusan seperti kelas penuh atau pembayaran berhasil/gagal. Dengan Activity Diagram, proses bisnis dalam aplikasi dapat dipahami secara jelas dan mudah, sehingga pengembang bisa memastikan bahwa alur aktivitas berjalan lancar dan tidak

ada langkah yang terlewatkan.

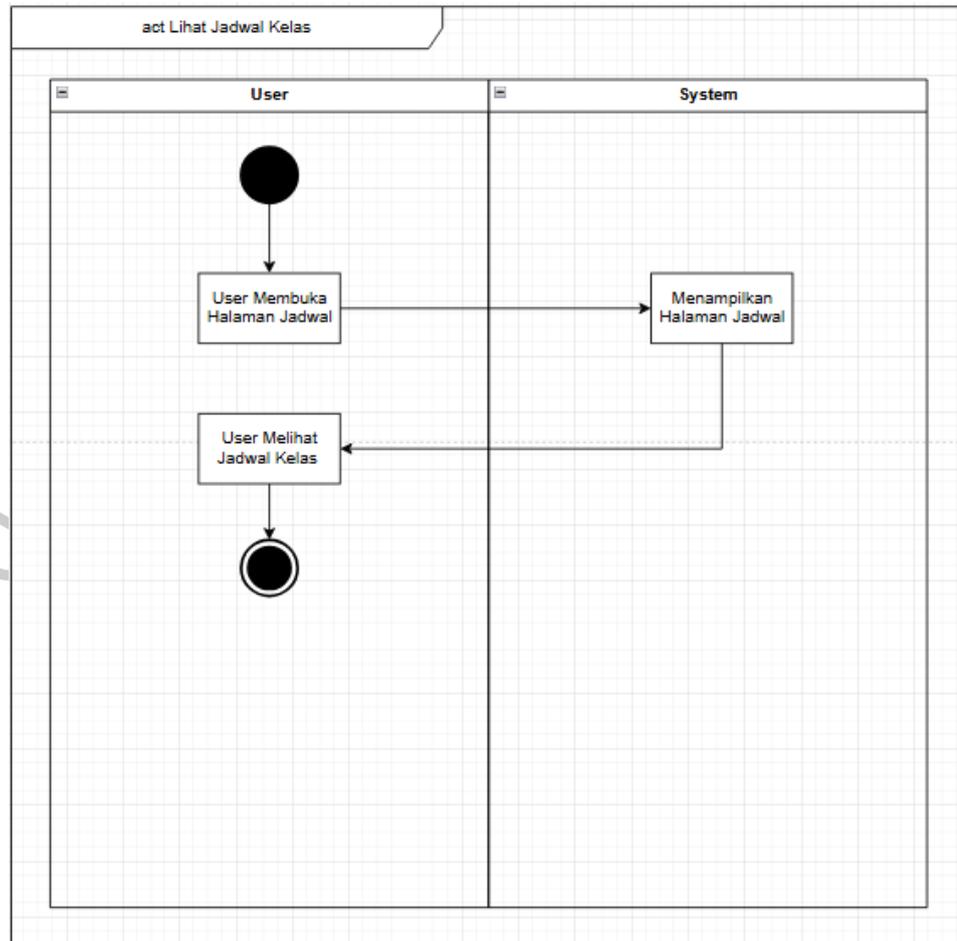


Gambar 4. 2 Activity Diagram Registrasi Akun

Proses registrasi akun dimulai ketika *Customer* memilih menu registrasi pada halaman utama aplikasi. Setelah itu, sistem dengan cepat menampilkan halaman registrasi yang berisi formulir untuk memasukkan data seperti email dan password. *Customer* mengisi formulir tersebut dengan data yang diperlukan, dan setelah itu menekan tombol "Daftar" untuk mengirimkan informasi.

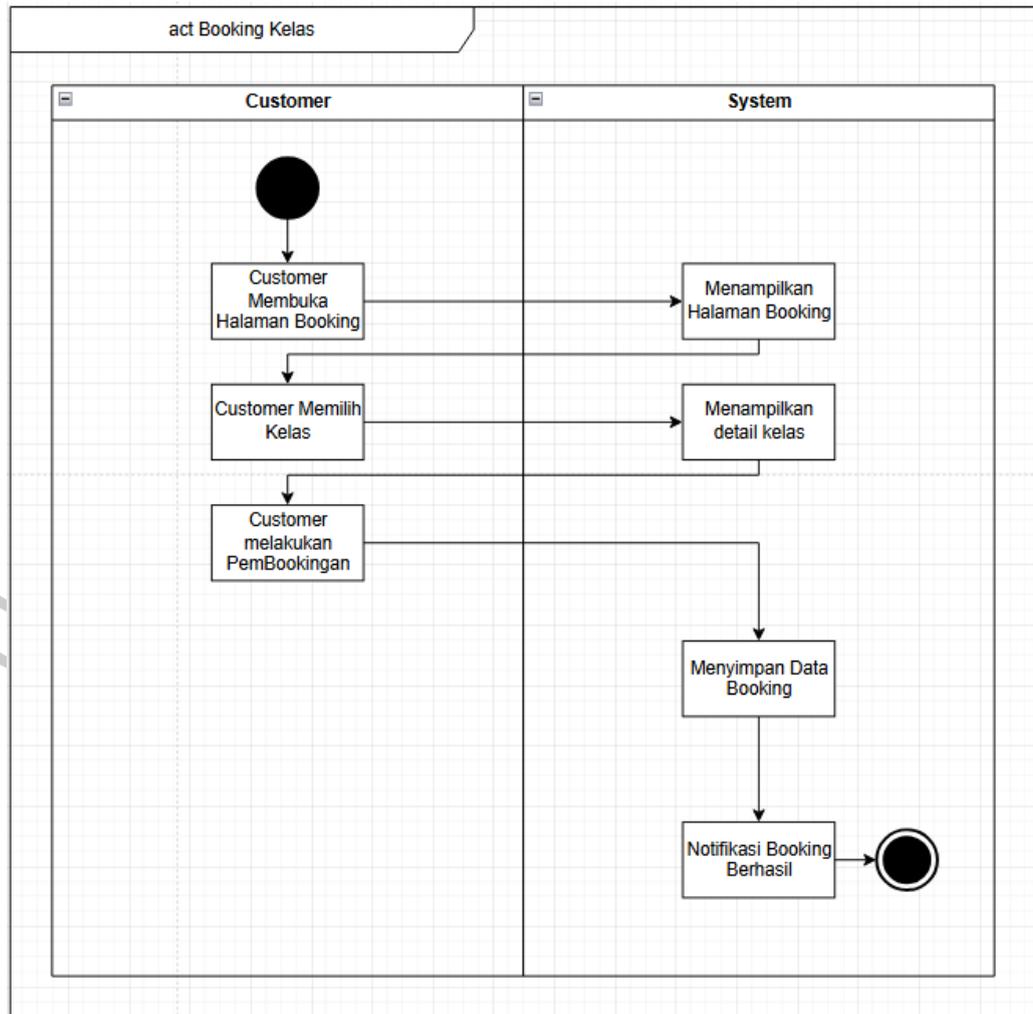
Sistem kemudian memverifikasi data yang dimasukkan oleh *Customer*, khususnya email yang digunakan untuk memastikan apakah email yang dimasukkan sudah terdaftar. Jika data email yang dimasukkan tersebut sudah terdaftar, sistem akan menampilkan notifikasi kesalahan yang memberi tahu *Customer* bahwa email tersebut telah digunakan. Sebaliknya, jika email belum terdaftar dan data valid, sistem akan menyimpan informasi akun baru

dan menampilkan notifikasi bahwa registrasi berhasil dilakukan.



Gambar 4. 3 Activity Diagram Lihat Jadwal Kelas

Proses dimulai ketika *User* membuka halaman jadwal di aplikasi. Setelah itu, Sistem akan merespons dengan cepat dan menampilkan halaman jadwal yang berisi informasi mengenai kelas-kelas yang tersedia. *User* kemudian melihat halaman jadwal tersebut, memeriksa detail informasi yang disediakan, seperti nama kelas, instruktur, dan waktu kelas. Setelah itu, proses berakhir, karena *User* telah berhasil melihat jadwal kelas yang ada di aplikasi.

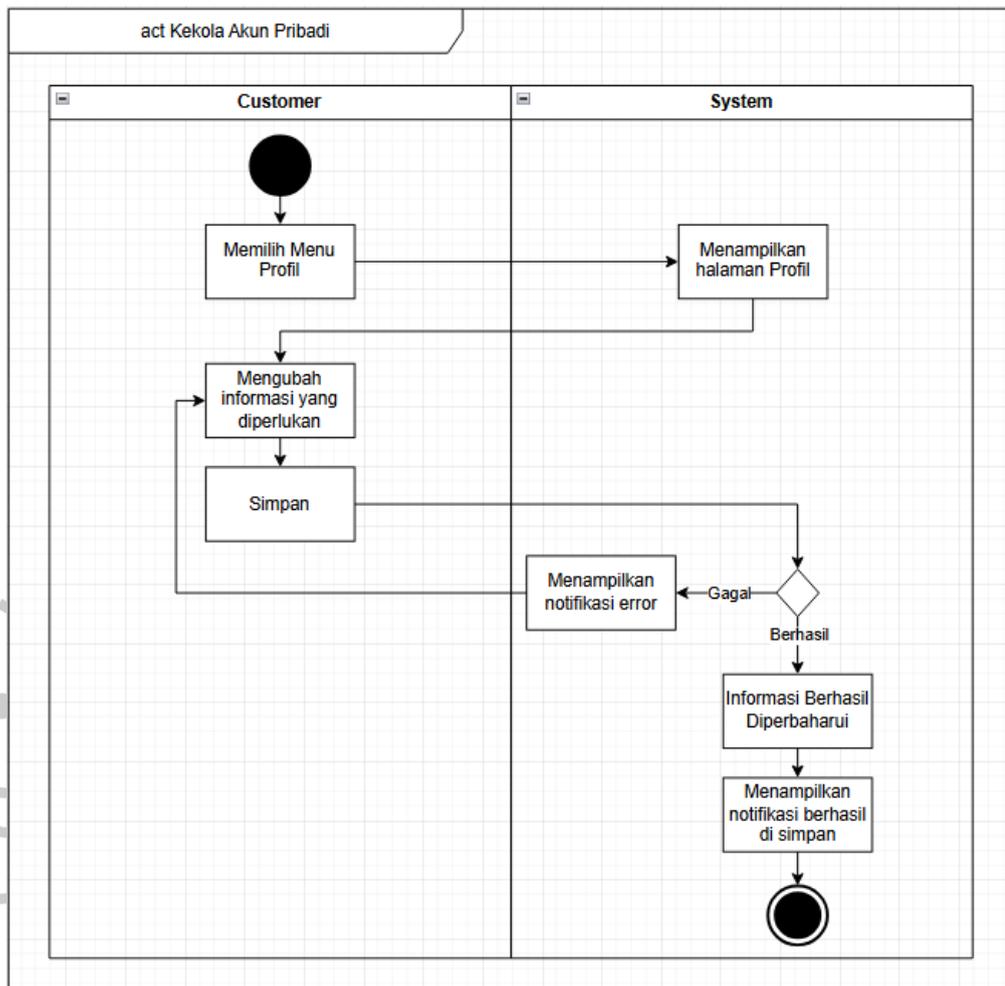


Gambar 4. 4 Activity Diagram Booking Kelas

Proses dimulai ketika *Customer* membuka halaman booking di aplikasi. Sistem kemudian menampilkan halaman booking yang berisi daftar kelas yang tersedia untuk dipilih oleh *Customer*. Setelah melihat daftar kelas, *Customer* memilih kelas yang diinginkan untuk diboeking.

Setelah *Customer* memilih kelas, sistem akan menampilkan detail kelas yang dipilih, seperti jadwal, instruktur, dan kapasitas kelas. *Customer* kemudian melakukan pemesanan kelas dengan menekan tombol untuk melanjutkan proses booking. Sistem akan menyimpan data booking yang dilakukan oleh *Customer* dan kemudian menampilkan notifikasi bahwa booking telah berhasil dilakukan.

Proses berakhir setelah *Customer* menerima notifikasi booking berhasil dan pemesanan kelas tercatat di sistem.



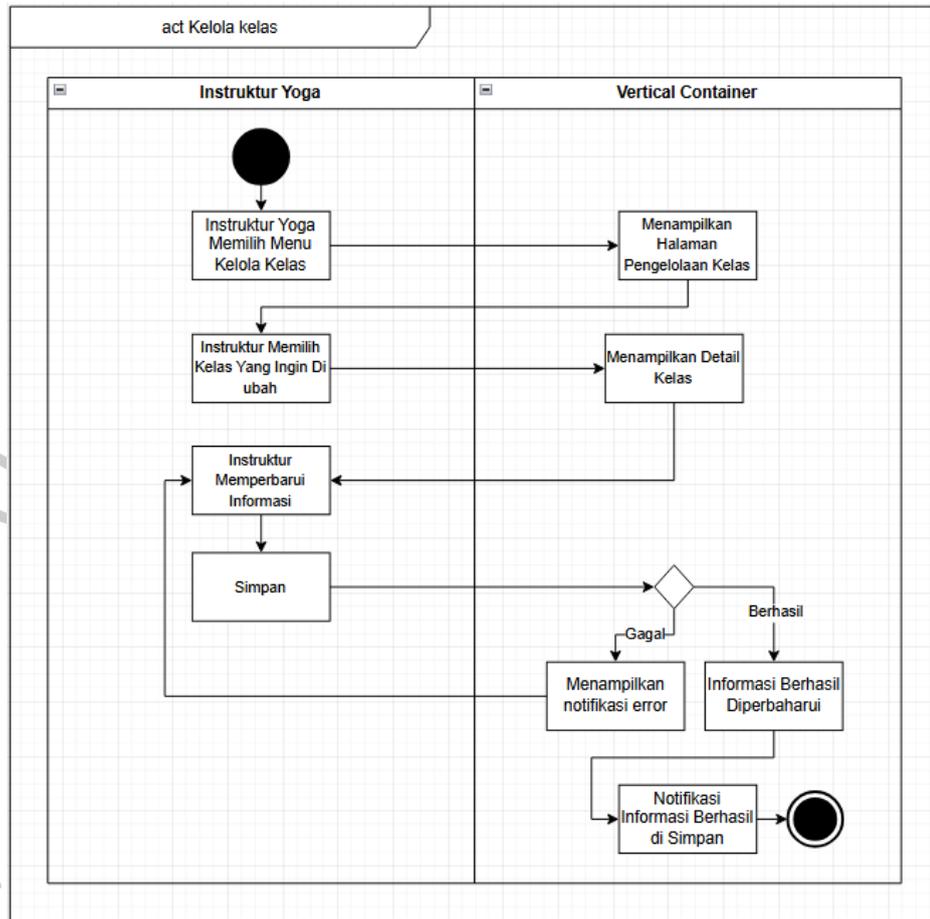
Gambar 4. 5 Activity Diagram Kelola Akun Pribadi

Proses dimulai ketika *User* memilih menu profil pada aplikasi. Sistem kemudian menampilkan halaman profil yang berisi informasi akun pribadi yang telah terdaftar, seperti nama, email, dan informasi lainnya. *User* kemudian mengubah informasi yang diperlukan, seperti memperbarui alamat email atau kata sandi.

Setelah selesai mengubah informasi, *User* menekan tombol "Simpan" untuk menyimpan pembaruan. Sistem kemudian memverifikasi apakah data yang dimasukkan valid. Jika terjadi kesalahan atau verifikasi gagal, sistem akan menampilkan notifikasi error yang menginformasikan *User* mengenai kesalahan tersebut.

Namun, jika pembaruan berhasil, sistem akan menyimpan perubahan yang dilakukan dan menampilkan notifikasi bahwa pembaruan akun berhasil

disimpan. Proses ini berakhir dengan pembaruan data akun yang sukses di sistem.

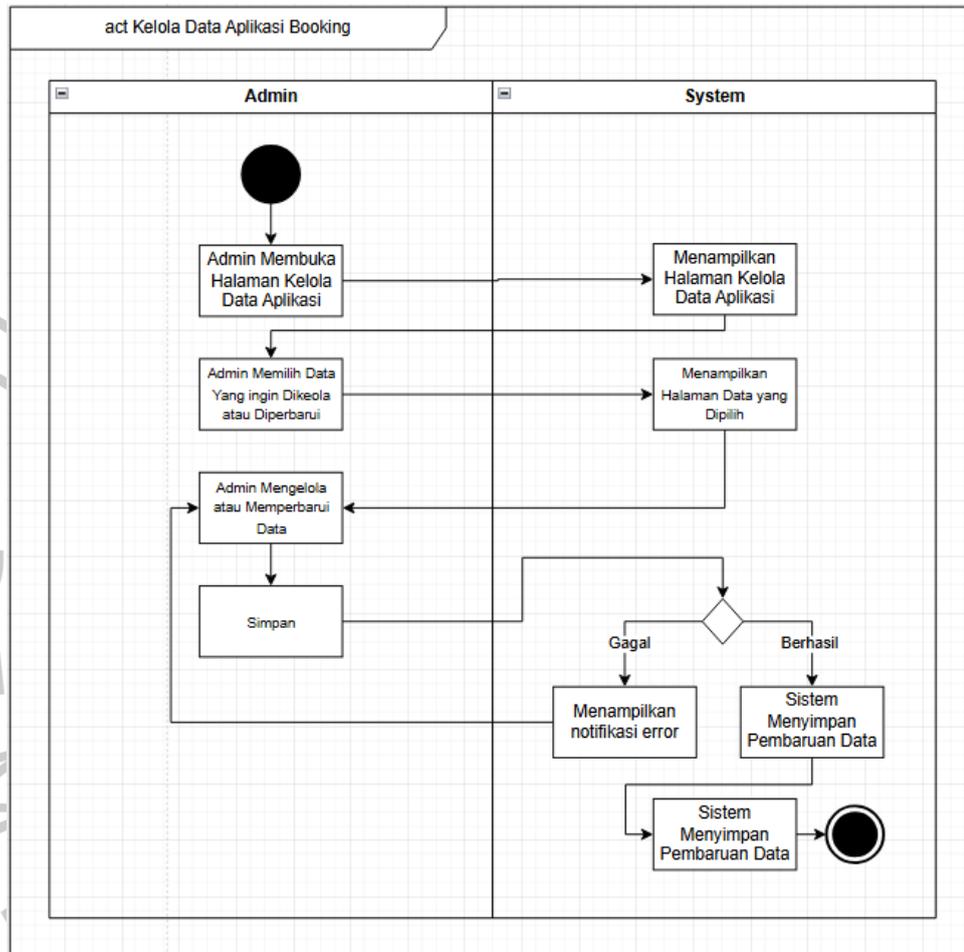


Gambar 4. 6 Activity Diagram Kelola Kelas

Proses dimulai ketika *Instruktur Yoga* memilih menu kelola kelas pada aplikasi. Sistem kemudian menampilkan halaman pengelolaan kelas yang berisi daftar kelas yang dapat dikelola. *Instruktur Yoga* kemudian memilih kelas yang ingin dikelola, misalnya untuk memperbarui jadwal atau kuota kelas.

Setelah memilih kelas yang ingin dikelola, sistem akan menampilkan detail kelas tersebut, seperti jadwal, instruktur, dan kapasitas kelas. *Instruktur Yoga* dapat memperbarui informasi kelas yang diperlukan, seperti mengubah waktu kelas atau menambah kapasitas peserta. Setelah perubahan selesai dilakukan, *Instruktur Yoga* mengklik tombol "Simpan" untuk menyimpan perubahan yang telah dilakukan.

Sistem kemudian akan memverifikasi data yang diperbarui. Jika ada kesalahan atau verifikasi gagal, sistem akan menampilkan notifikasi error. Namun, jika informasi berhasil diperbarui, sistem akan menyimpan perubahan dan menampilkan notifikasi bahwa informasi kelas berhasil diperbarui.



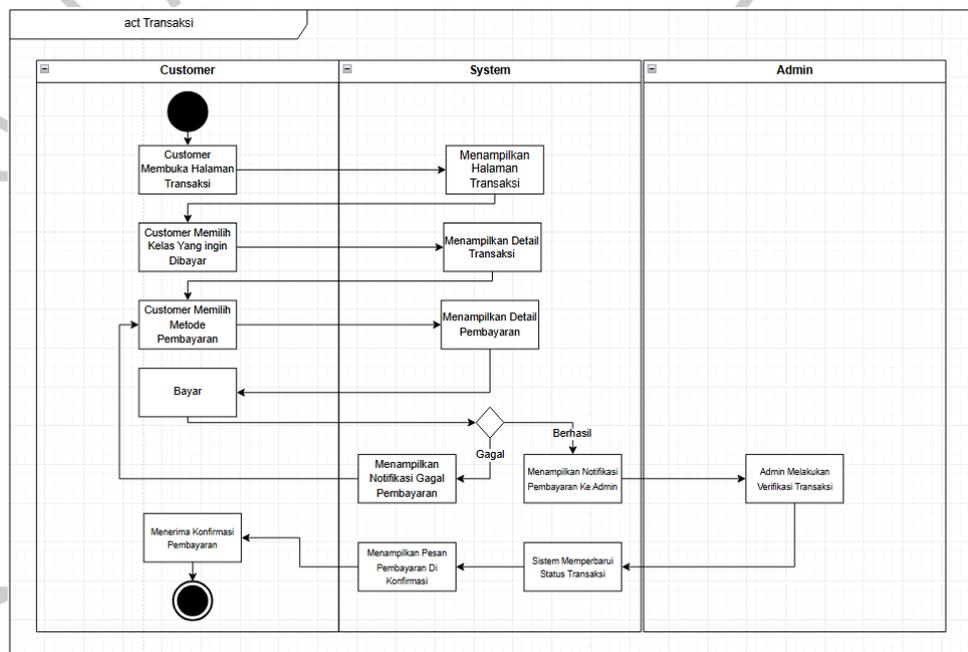
Gambar 4. 7 Activity Diagram Kelola Data Aplikasi Booking

Proses dimulai ketika *Admin* membuka halaman pengelolaan data aplikasi booking pada sistem. Sistem kemudian menampilkan halaman kelola data yang berisi berbagai opsi untuk mengelola data terkait kelas, booking, atau pengguna.

Setelah halaman ditampilkan, *Admin* memilih data yang ingin diperbarui atau dikelola, seperti memilih data kelas, booking, atau pengguna yang perlu diubah. Sistem kemudian menampilkan detail data yang dipilih oleh *Admin*.

*Admin* selanjutnya mengelola data dengan memperbarui informasi yang diperlukan, seperti mengubah detail kelas, status pemesanan, atau data pengguna. Setelah perubahan dilakukan, *Admin* mengklik tombol "Simpan" untuk menyimpan perubahan yang telah dibuat.

Sistem akan memverifikasi apakah data yang dimasukkan valid. Jika terjadi kesalahan atau verifikasi gagal, sistem akan menampilkan notifikasi error. Namun, jika perubahan berhasil disimpan, Sistem kemudian akan menyimpan data yang telah diperbarui dan menampilkan pemberitahuan bahwa pembaruan data berhasil dilakukan.



Gambar 4. 8 Activity Diagram Transaksi

Proses dimulai ketika *Customer* membuka halaman transaksi di aplikasi. Sistem kemudian menampilkan halaman transaksi yang berisi informasi tentang pemesanan kelas yang telah dilakukan oleh *Customer*. *Customer* kemudian memilih kelas yang ingin dibayar dari daftar transaksi yang tersedia. Setelah itu, sistem akan menampilkan detail transaksi untuk kelas yang dipilih.

Setelah melihat detail transaksi, *Customer* memilih metode pembayaran yang diinginkan, seperti transfer bank atau pembayaran melalui aplikasi pembayaran digital. Sistem kemudian menampilkan detail pembayaran yang mencakup informasi rekening atau pilihan pembayaran lainnya.

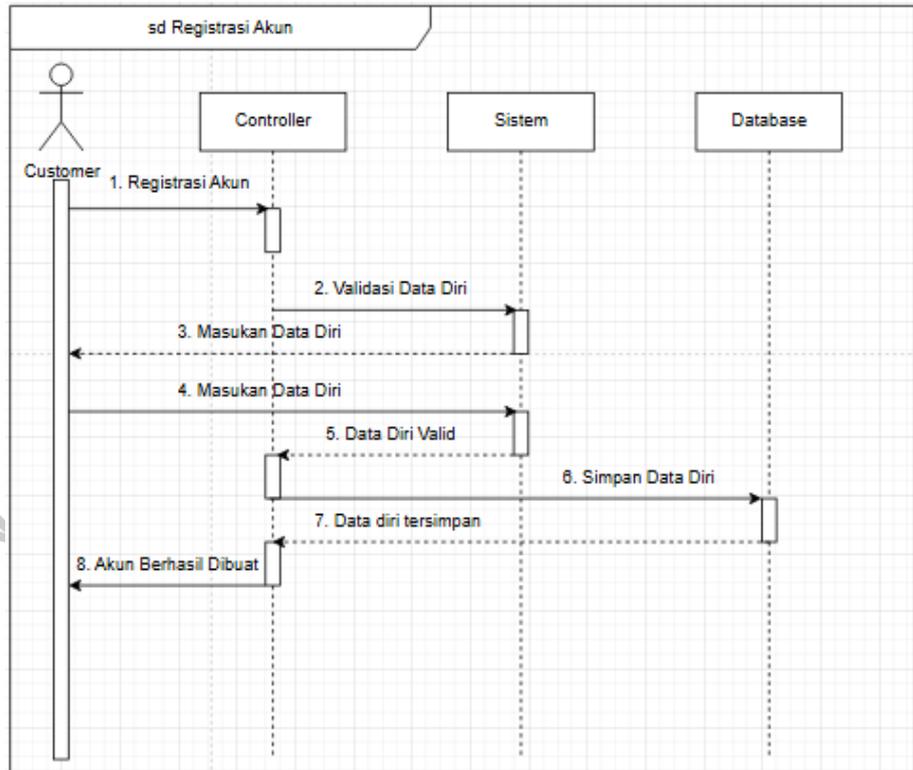
*Customer* melanjutkan dengan melakukan pembayaran sesuai metode yang dipilih. Sistem kemudian memeriksa apakah pembayaran berhasil dilakukan. Jika pembayaran gagal, sistem akan menampilkan notifikasi gagal dan meminta *Customer* untuk mencoba kembali.

Jika pembayaran berhasil, sistem akan mengirimkan notifikasi ke *Admin* bahwa pembayaran telah dilakukan. *Admin* kemudian memverifikasi transaksi dan memastikan bahwa pembayaran sesuai dengan data yang ada. Setelah *Admin* melakukan verifikasi, sistem akan memperbarui status transaksi menjadi "pembayaran diterima."

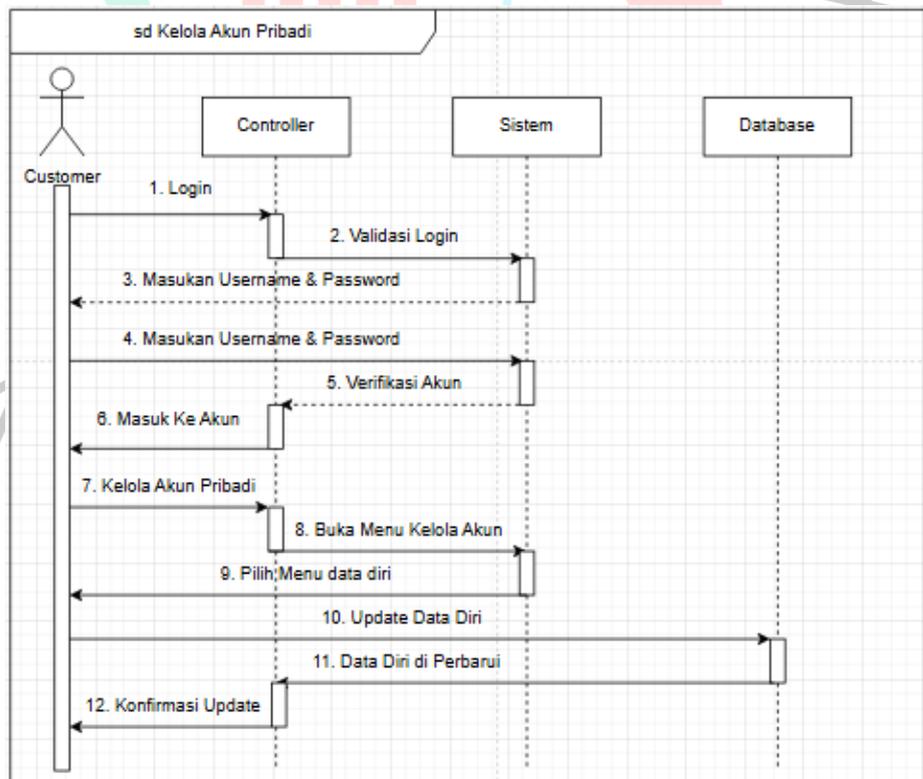
Setelah status transaksi diperbarui, sistem akan menampilkan pesan konfirmasi pembayaran kepada *Customer*, menginformasikan bahwa pembayaran telah diterima dan diverifikasi. *Customer* kemudian menerima konfirmasi pembayaran dan proses transaksi berakhir.

#### **4.2.4 Sequence Diagram ( Iterasi 1)**

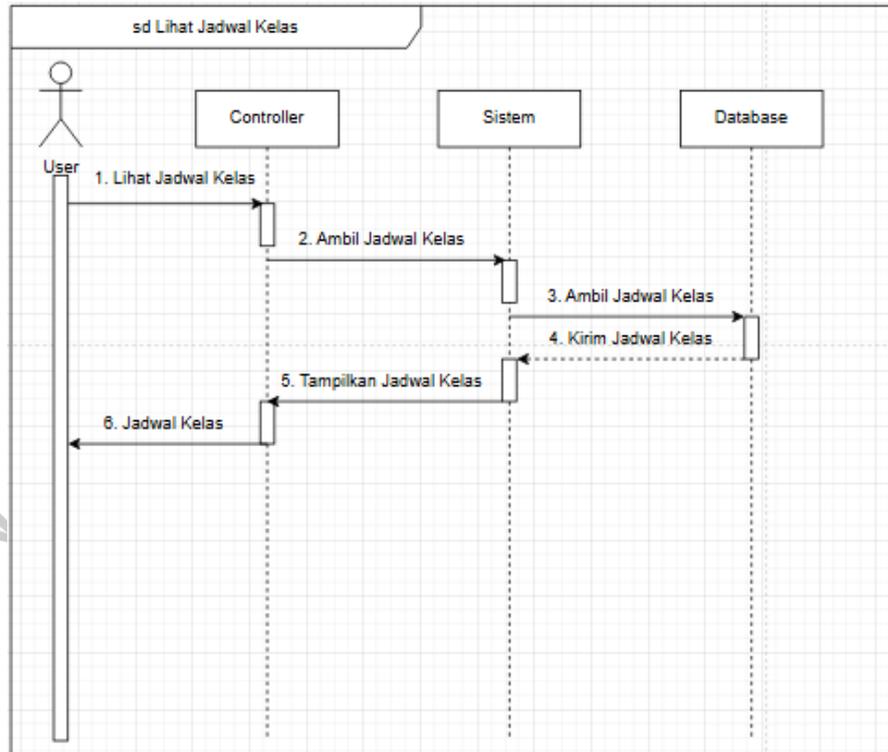
Sequence Diagram menggambarkan urutan interaksi antara objek-objek yang terlibat dalam suatu skenario, secara berurutan. Diagram ini akan digunakan untuk menggambarkan bagaimana pengguna, seperti Member atau Admin, berinteraksi dengan sistem dalam setiap proses, mulai dari Mendaftar Kelas Yoga, Melakukan Pembayaran, hingga Menerima Konfirmasi Booking. Sequence Diagram juga akan menunjukkan bagaimana sistem berinteraksi dengan database untuk menyimpan data seperti informasi pengguna, jadwal kelas, dan riwayat pembayaran. Misalnya, ketika seorang Member memilih kelas, sistem akan memverifikasi ketersediaan kelas, mengupdate status kelas, dan memproses pembayaran secara berurutan. Diagram ini memberikan gambaran yang lebih mendetail mengenai alur komunikasi dan data dalam sistem.



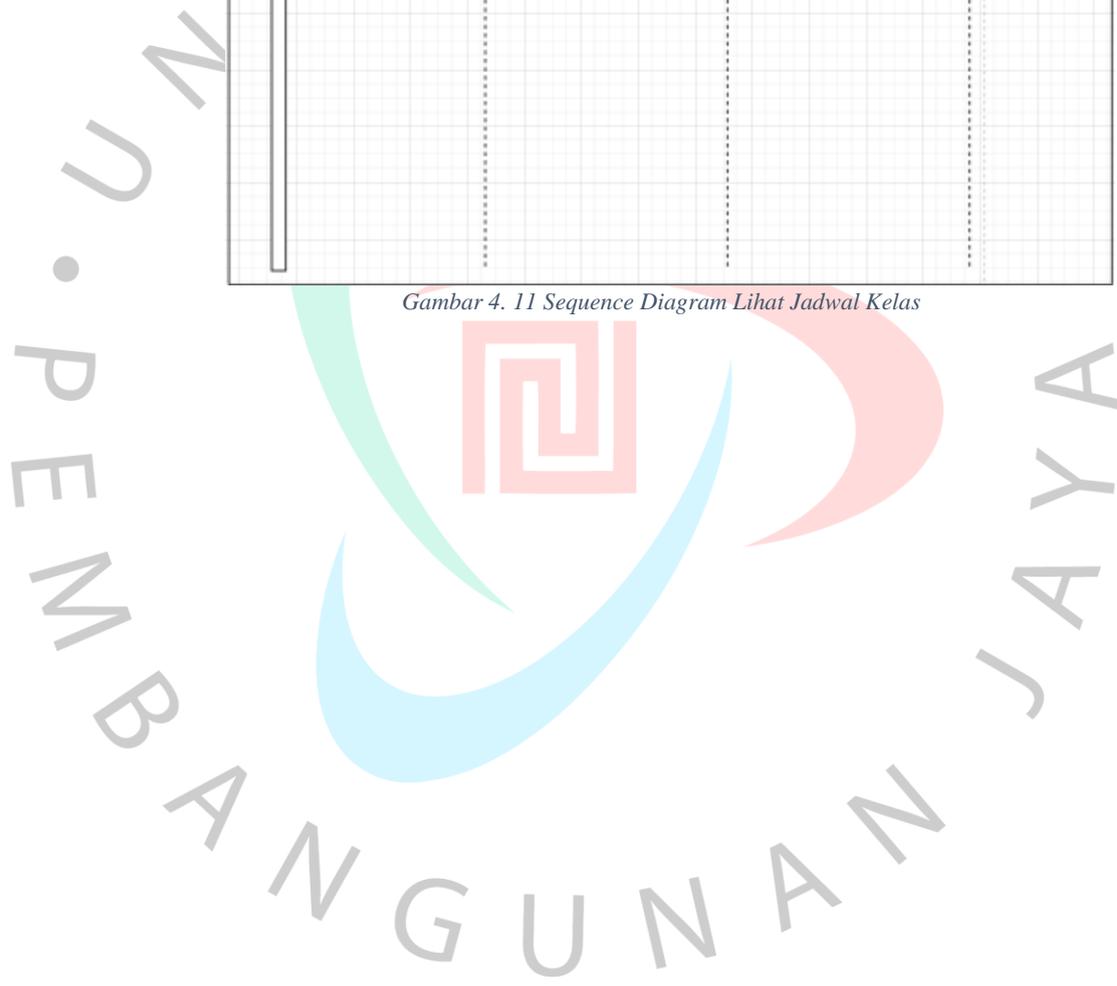
Gambar 4. 9 Sequence Diagram Registrasi Akun

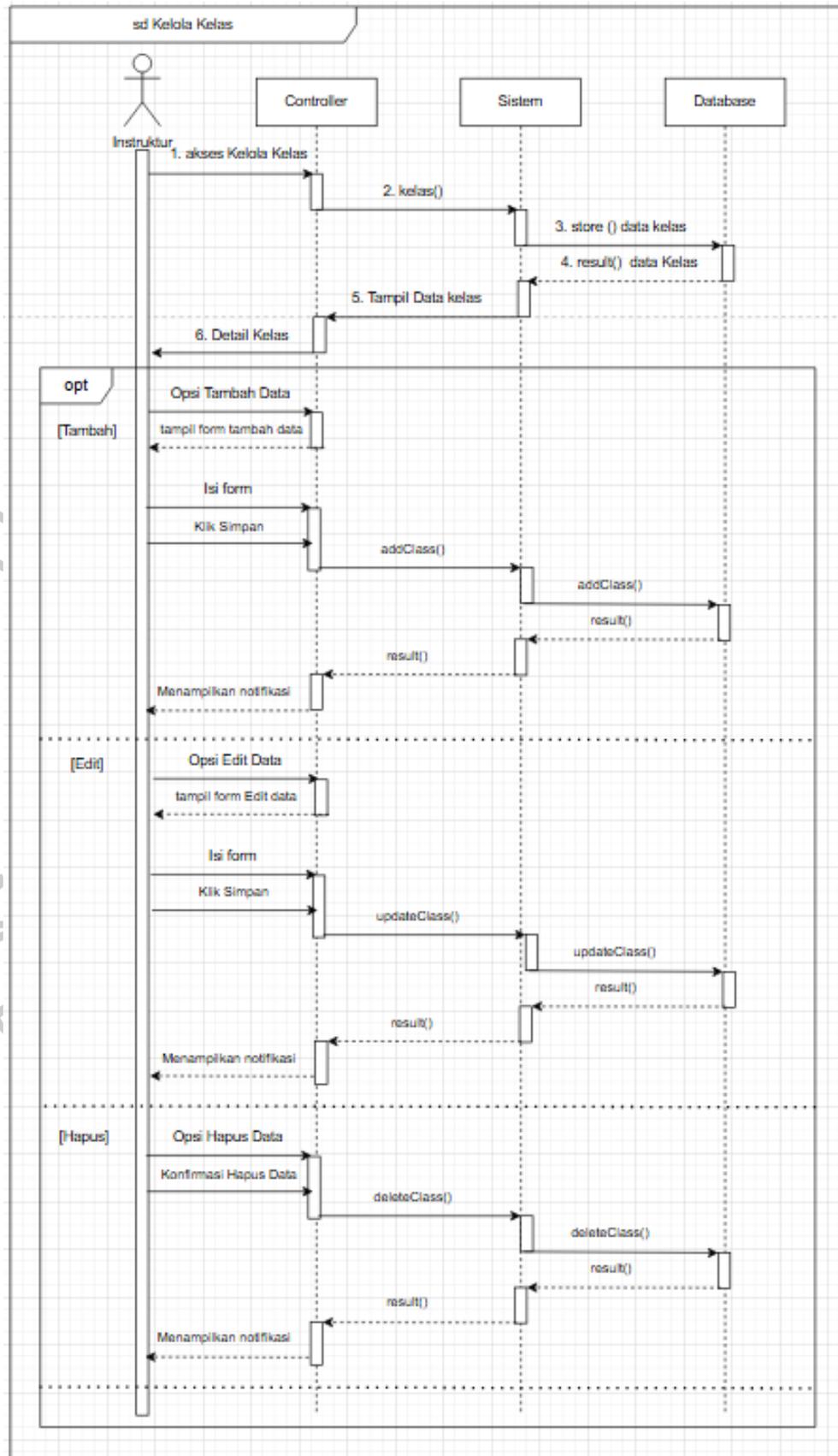


Gambar 4. 10 Sequence Diagram Kelola Akun Pribadi

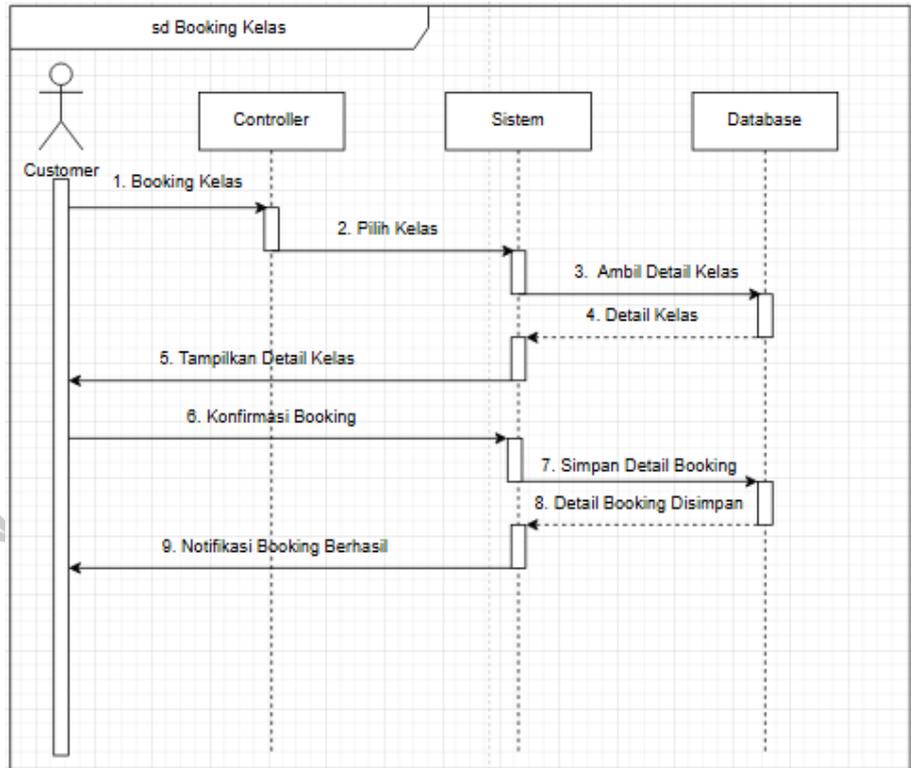


Gambar 4. 11 Sequence Diagram Lihat Jadwal Kelas

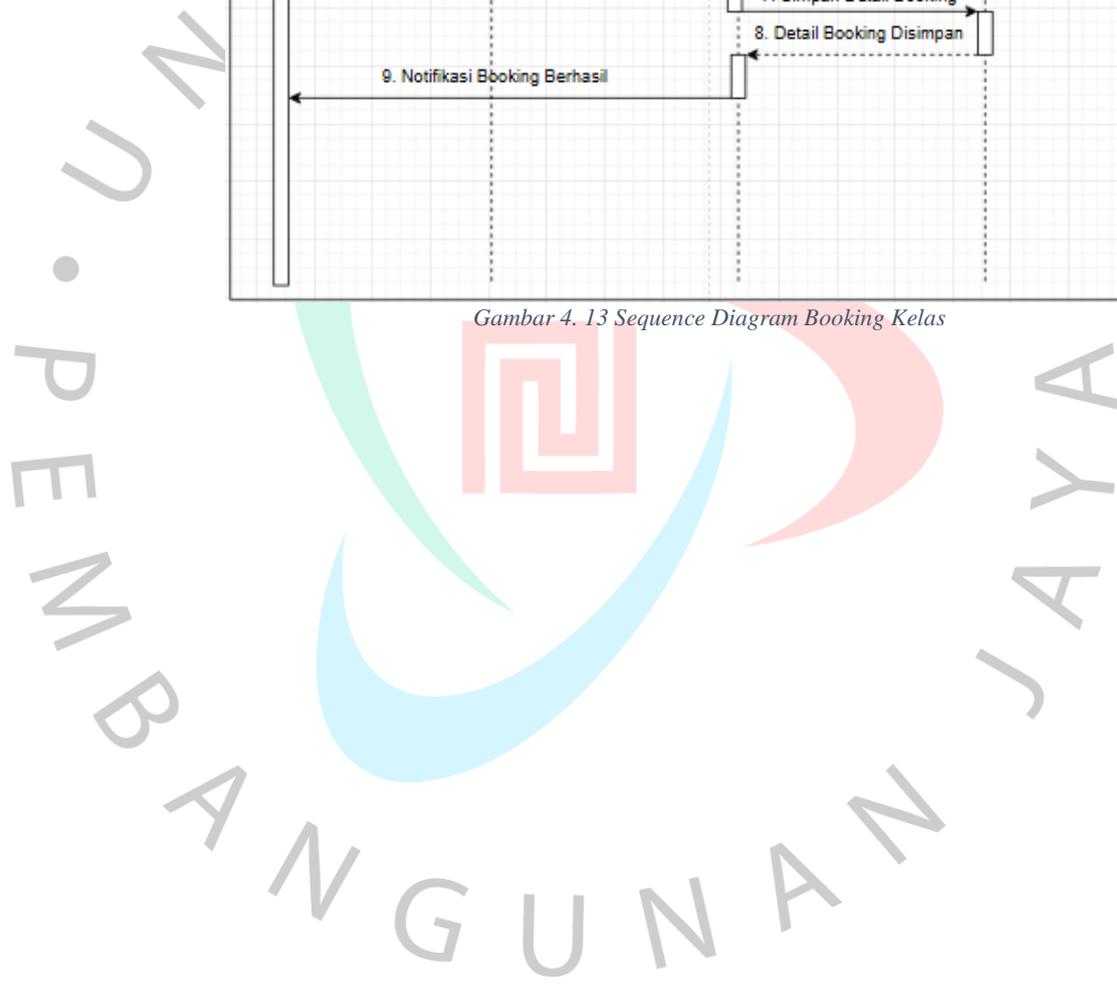


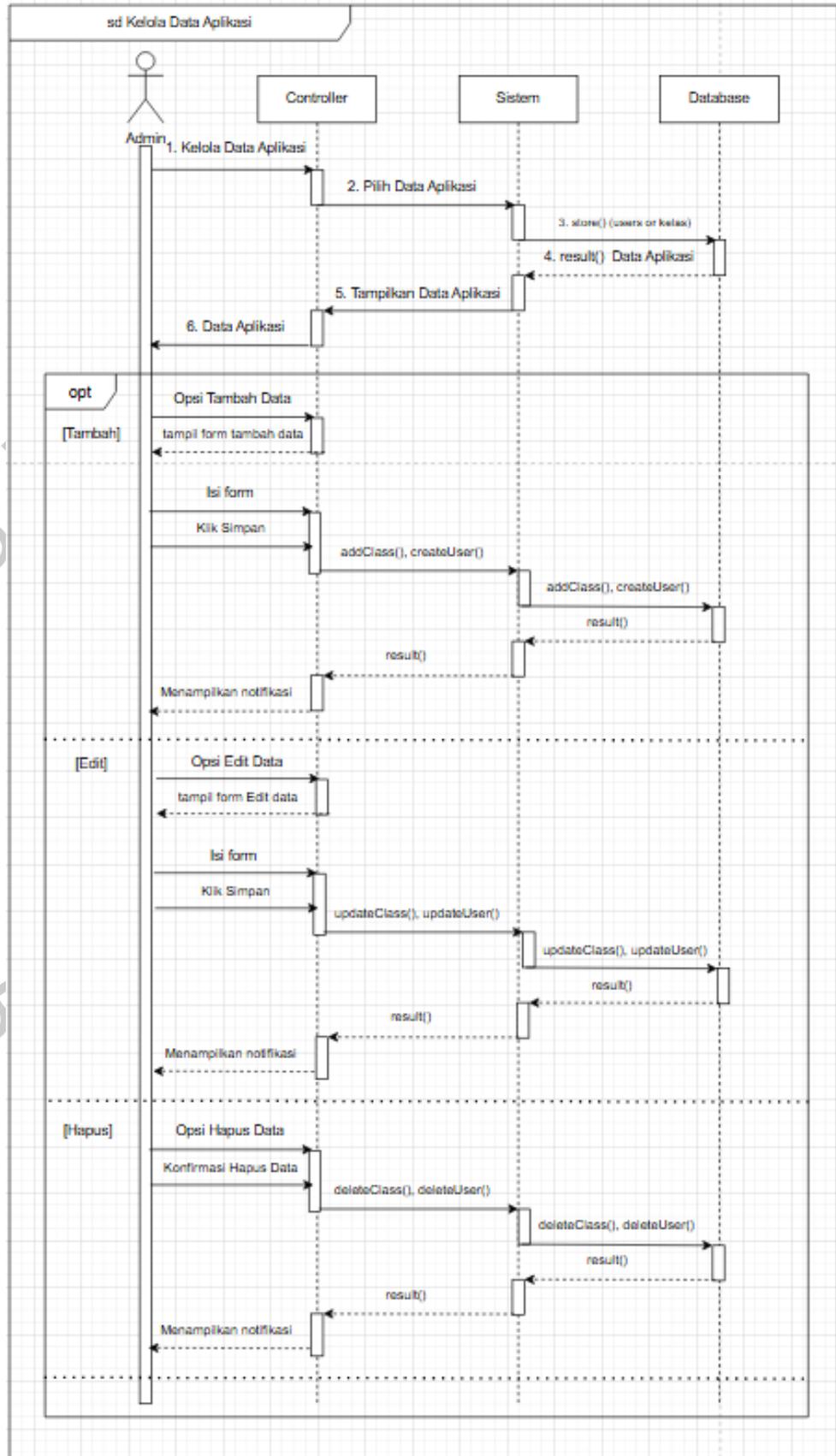


Gambar 4. 12 Sequence Diagram Kelola Kelas

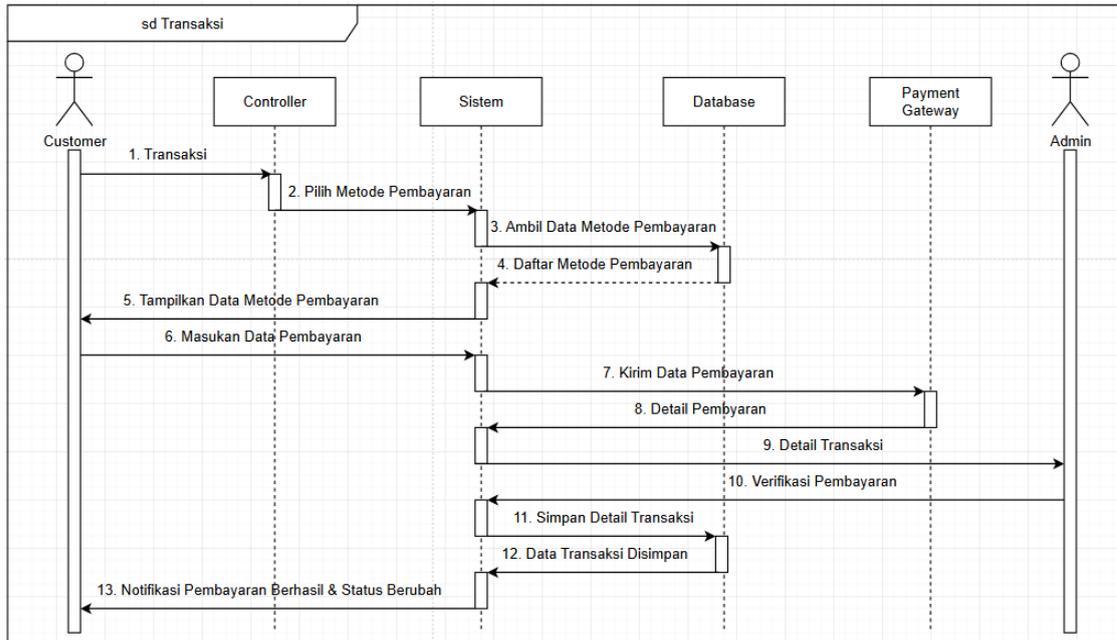


Gambar 4. 13 Sequence Diagram Booking Kelas





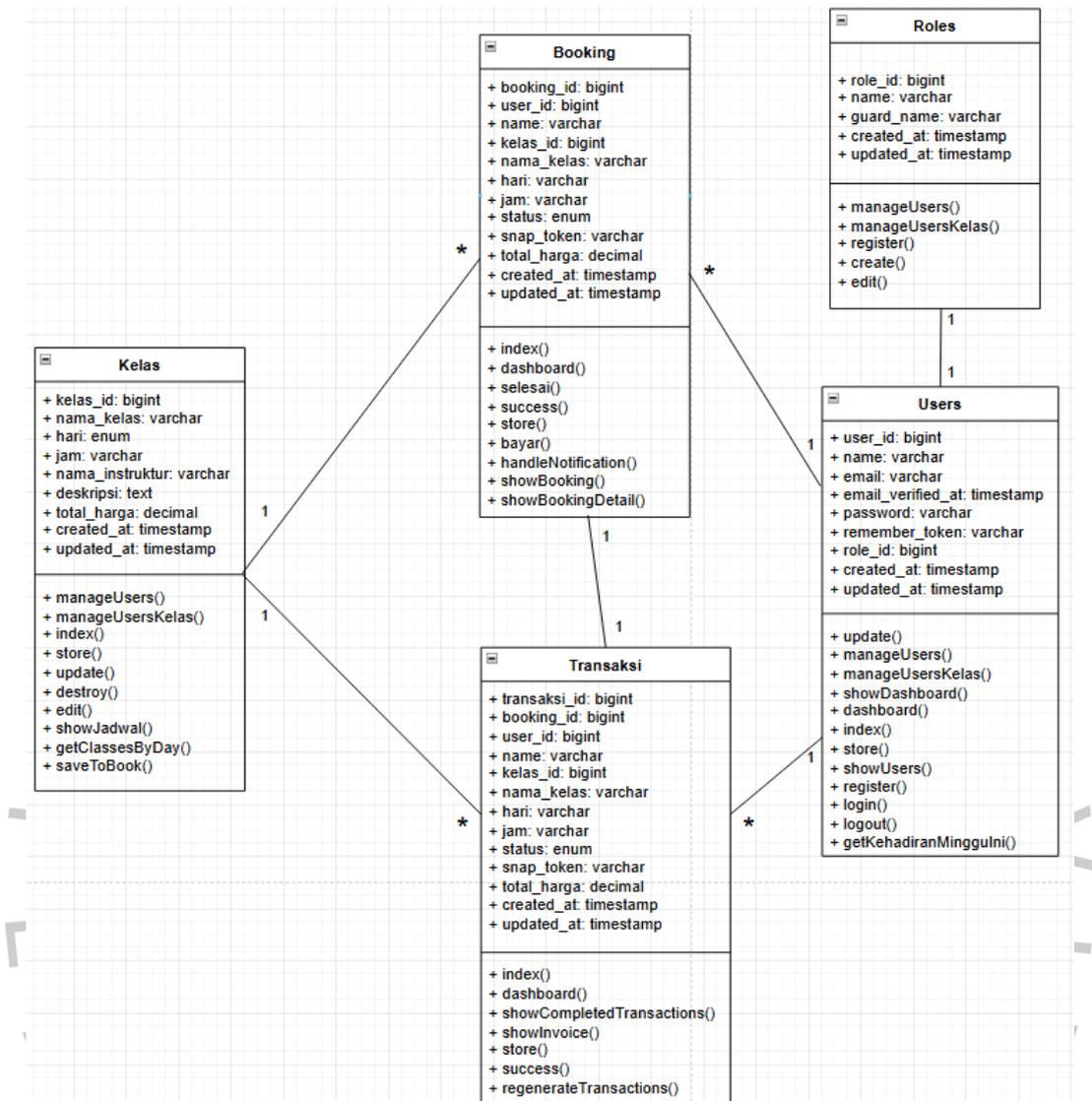
Gambar 4. 14 Sequence Diagram Kelola Data Aplikasi



Gambar 4. 15 Sequence Diagram Transaksi

#### 4.2.5 Class Diagram ( Iterasi 1)

Class Diagram adalah diagram yang menggambarkan struktur statis dari sistem, dengan menunjukkan berbagai kelas (entitas), atribut, dan relasi antar kelas yang ada dalam aplikasi. Dalam sistem pengelolaan booking kelas yoga, Class Diagram akan menggambarkan kelas-kelas seperti User, Kelas, Booking, Pembayaran, dan Instruktur, serta hubungan antar kelas tersebut. Diagram ini akan menunjukkan bagaimana setiap kelas berinteraksi satu sama lain, seperti bagaimana Booking berhubungan dengan User dan Kelas, serta bagaimana Pembayaran terkait dengan Booking. Dengan adanya Class Diagram, pengembang dapat merancang struktur database dan objek dalam sistem, serta memastikan bahwa semua entitas dan relasi di dalam aplikasi dapat dikelola dengan baik.



Gambar 4. 16 Class Diagram

#### 4.2.6 Spesifikasi Database ( Iterasi 1)

Spesifikasi Database menggambarkan desain dan struktur database yang digunakan untuk menyimpan dan mengelola data dalam aplikasi pengelolaan sistem booking kelas yoga. Spesifikasi ini mencakup definisi tabel, atribut, tipe data, serta relasi antar tabel yang ada dalam database. Contoh tabel yang akan dijelaskan dalam spesifikasi database ini meliputi tabel User (untuk menyimpan data pengguna seperti nama, email, dan password), tabel Kelas (untuk menyimpan informasi tentang kelas yoga seperti nama kelas, instruktur, dan harga), dan tabel Booking (untuk mencatat pemesanan yang dilakukan oleh

pengguna). Spesifikasi ini juga akan mencakup aturan-aturan penting terkait integritas data, seperti foreign key constraints yang menghubungkan tabel Booking dengan User dan Kelas, serta aturan untuk menjaga konsistensi data dalam sistem.

#### A. Tabel booking

Tabel 4. 8 Spesifikasi Database Table Booking

Nama Field	Type	Size	Keterangan
booking_id	bigint	20	Kode Booking (Primary Key)
user_id	bigint	20	Kode User (Foreign Key)
name	varchar	255	Nama Pemesan
kelas_id	bigint	20	Kode Kelas (Foreign Key)
nama_kelas	varchar	255	Nama Kelas
hari	varchar	255	Hari Jadwal
jam	varchar	255	Jam Jadwal
status	enum('paid', 'unpaid', 'failed')	—	Status Pembayaran (Paid/Unpaid/Failed)
snap_token	varchar	255	Token Snap (optional)
total_harga	decimal	—	Total Harga Pembelian
created_at	timestamp	—	Waktu Dibuat
updated_at	timestamp	—	Waktu Diperbarui

Penjelasan:

1. Primary Key:  
booking\_id adalah kunci utama yang secara unik mengidentifikasi setiap baris data dalam tabel booking.
2. Foreign Keys:

user\_id: Menghubungkan tabel booking dengan tabel user (informasi pemesan).

3. kelas\_id: Menghubungkan tabel booking dengan tabel kelas (informasi kelas yang dipesan).

4. Enum Status:

Kolom status memiliki nilai enum yang memungkinkan status pembayaran untuk dibedakan: 'paid', 'unpaid', atau 'failed'.

5. Tanggal Waktu:

created\_at dan updated\_at menggunakan tipe data timestamp untuk melacak kapan data dibuat dan diperbarui.

## B. Tabel transaksi

Tabel 4. 9 Spesifikasi Database Table Transaksi

Nama Field	Type	Size	Keterangan
transaksi_id	bigint	20	Kode Transaksi (Primary Key)
user_id	bigint	20	Kode User (Foreign Key)
booking_id	bigint	20	Kode Booking
name	varchar	255	Nama Pemesan
kelas_id	bigint	20	Kode Kelas (Foreign Key)
nama_kelas	varchar	255	Nama Kelas
hari	varchar	255	Hari Jadwal
jam	varchar	255	Jam Jadwal
total_harga	decimal	—	Total Harga Pembelian
status	enum('paid', 'unpaid', 'failed')	—	Status Pembayaran (Paid/Unpaid/Failed)

snap_token	varchar	255	Token Snap (optional)
created_at	timestamp	—	Waktu Dibuat
updated_at	timestamp	—	Waktu Diperbarui

Penjelasan:

1. Primary Key:  
transaksi\_id adalah kunci utama yang secara unik mengidentifikasi setiap transaksi.
2. Foreign Keys:  
user\_id: Menghubungkan tabel transaksi dengan tabel user (informasi pengguna yang melakukan transaksi).
3. Enum Status:  
Kolom status berisi status pembayaran dengan opsi 'paid', 'unpaid', atau 'failed'.
4. Tanggal Waktu:  
created\_at dan updated\_at menggunakan tipe data timestamp untuk melacak kapan data dibuat dan diperbarui.

### C. Tabel kelas

Tabel 4. 10 Spesifikasi Database Table Kelas

Nama Field	Type	Size	Keterangan
kelas_id	bigint	20	Kode Kelas (Primary Key)
nama_kelas	varchar	255	Nama Kelas
jam	varchar	255	Jam Kelas
nama_instruktur	varchar	255	Nama Instruktur
total_harga	decimal	—	Total Harga Kelas
deskripsi	text	utf8mb4	Deskripsi Kelas
created_at	timestamp	—	Waktu Dibuat

updated_at	timestamp	—	Waktu Diperbarui
hari	enum('Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu', 'Minggu')	—	Hari Kelas

Penjelasan:

1. Primary Key:

id adalah kunci utama yang secara unik mengidentifikasi setiap kelas.

2. Enum Hari:

Kolom hari menggunakan tipe data enum yang memungkinkan untuk memilih hari yang sesuai dari pilihan seperti 'Senin', 'Selasa', 'Rabu', 'Kamis', dan 'Jumat'.

3. Tanggal Waktu:

created\_at dan updated\_at menggunakan tipe data timestamp untuk melacak kapan data dibuat dan diperbarui.

#### D. Tabel Roles

Tabel 4. 11 Spesifikasi Database Table Roles

Nama Field	Type	Size	Keterangan
role_id	bigint	20	Kode Role (Primary Key)
name	varchar	255	Nama Role
guard_name	varchar	255	Nama Guard (untuk pengamanan role)
created_at	timestamp	—	Waktu Dibuat
updated_at	timestamp	—	Waktu Diperbarui

Penjelasan:

1. Primary Key:

id adalah kunci utama yang secara unik mengidentifikasi setiap

role.

2. Unique Index pada name:

Kolom name memiliki indeks unik, yang memastikan setiap nama role adalah unik dalam tabel ini.

3. Tanggal Waktu:

created\_at dan updated\_at menggunakan tipe data timestamp untuk melacak kapan data dibuat dan diperbarui.

**E. Tabel Users**

*Tabel 4. 12 Spesifikasi Database Table Users*

<b>Nama Field</b>	<b>Type</b>	<b>Size</b>	<b>Keterangan</b>
user_id	bigint	20	Kode User (Primary Key)
name	varchar	255	Nama Pengguna
email	varchar	255	Alamat Email
email_verified_at	timestamp	—	Waktu Verifikasi Email
password	varchar	255	Password Pengguna
remember_token	varchar	utf8mb4	Token untuk Mengingat Pengguna (opsional)
created_at	timestamp	—	Waktu Dibuat
updated_at	timestamp	—	Waktu Diperbarui
role_id	bigint	20	Kode Role (Foreign Key)

Penjelasan:

1. Primary Key:

id adalah kunci utama yang secara unik mengidentifikasi setiap pengguna.

2. Unique Index pada email:

Kolom email memiliki indeks unik, yang memastikan setiap alamat email adalah unik dalam tabel ini.

3. Foreign Key pada role\_id:

role\_id adalah kunci asing yang menghubungkan pengguna dengan tabel roles untuk menentukan peran atau hak akses pengguna.

4. Tanggal Waktu:

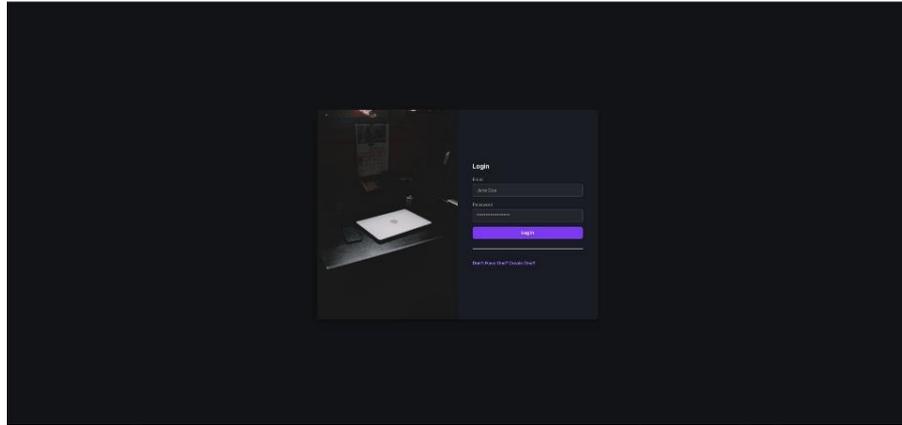
created\_at dan updated\_at menggunakan tipe data timestamp untuk melacak kapan data dibuat dan diperbarui.

### 4.3 Prototipe User Interface ( Iterasi 1)

Antarmuka pengguna (user interface) adalah komponen krusial dalam suatu sistem, yang memungkinkan interaksi langsung antara pengguna dan sistem tersebut. Desain yang baik akan menentukan kenyamanan dan kemudahan pengguna dalam mengakses serta menggunakan aplikasi. Pada tahap pertama ini, prototipe antarmuka difokuskan pada sistem informasi booking kelas yoga padmaprana yoga, khususnya untuk modul seperti pendaftaran, penjadwalan, dan pembayaran tahap 1. Prototipe ini dirancang untuk memastikan bahwa alur kerja dan interaksi pengguna dapat berjalan dengan efektif, dengan memperhatikan kemudahan navigasi dan fungsionalitas dasar yang diperlukan.

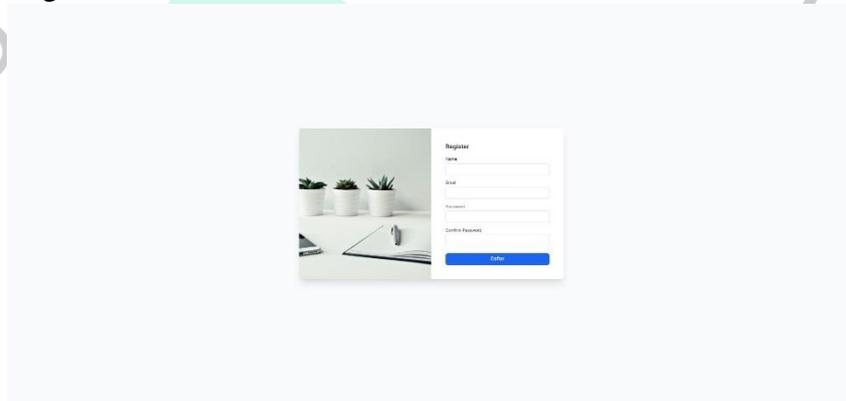
Pada fase ini, prototipe yang dikembangkan masih bersifat awal dan belum final. Prototipe tersebut akan diuji coba untuk mendapatkan umpan balik dari pengguna, yang kemudian digunakan untuk melakukan revisi dan perbaikan. Tujuan utama dari tahap ini adalah memastikan desain antarmuka yang diusulkan dapat memenuhi kebutuhan dan harapan pengguna, sebelum dilanjutkan ke tahapan pengembangan lebih lanjut yang lebih kompleks.

### 1) Login



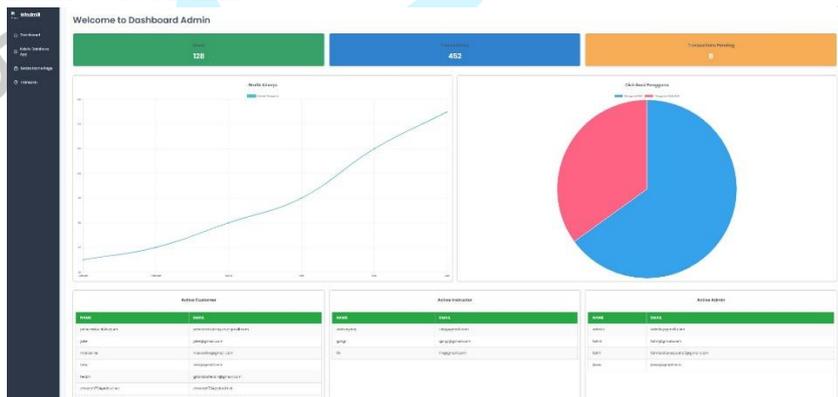
Gambar 4. 17 Tampilan Login

### 2) Register



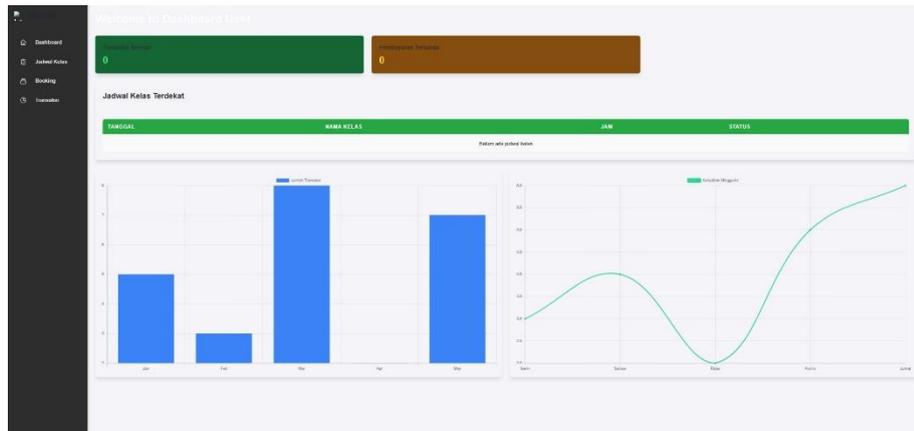
Gambar 4. 18 Tampilan Register

### 3) Dashboard Admin



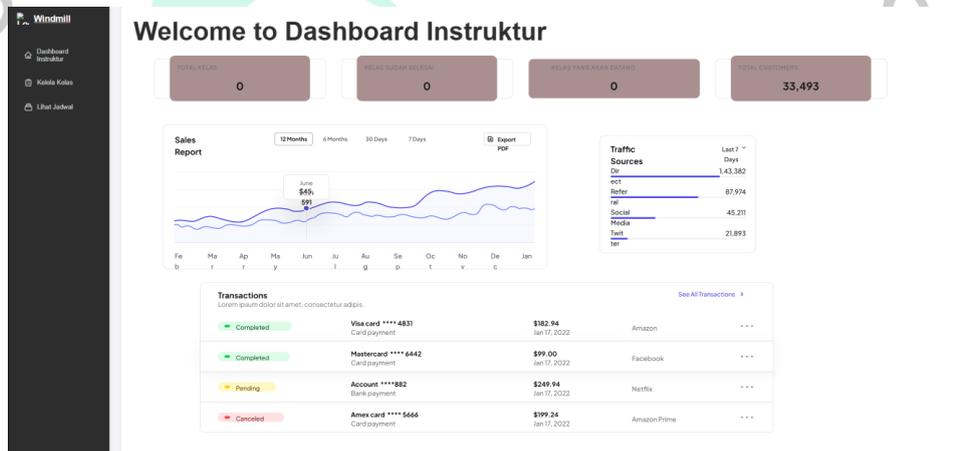
Gambar 4. 19 Tampilan Dashboard Admin

#### 4) Dashboard User



Gambar 4. 20 Tampilan Dashboard User

#### 5) DashboardInstruktur



Gambar 4. 21 Tampilan Dashboard Instruktur

#### 6) KelolaDataAdmin

ID	NAMA KELAS	HARI	JAM	DESKRIPSI	TOTAL PENDAPATAN	ACTIONS
1	Kelas Visualisasi	Senin	12:00	menjelaskan salah satu प्रकार प्रणाली yang terintegrasi dengan mudah di dalam berbagai bahasa. Selain itu, aplikasi ini diinstal dapat meningkatkan konektivitas dan stabilitas lebih lanjut.	76,000.00	[edit] [delete]
26	Kelas Pengembangan	Jumat	17:00	menjelaskan jenis teknik yang dapat meningkatkan nilai manajemen, manajemen pada dasarnya, dan manajemen lebih lanjut.	1,900.00	[edit] [delete]
31	Kelas Satu Dimensi	Kamis	08:00	Desain proses, menggunakan alat pengujian, menggunakan cara dan biaya yang berbeda, menggunakan masalah bisnis.	800.00	[edit] [delete]
32	Kelas Pemrosesan	Kamis	15:00	Menjelaskan prosedur, meningkatkan ke efisien, menggunakan pada dasarnya, meningkatkan, pengujian, perbaikan, dan pemrosesan.	180,000.00	[edit] [delete]
33	Kelas Satu Dimensi	Senin	08:00	Faktor-faktor yang mempengaruhi biaya, menggunakan pada dasarnya, meningkatkan, pengujian, perbaikan, dan pemrosesan.	58,000.00	[edit] [delete]
37	Kelas Tabulasi	Senin	08:00	Proses pengujian, diinstal sebagai bagian dari semua jenis yang.	50,000.00	[edit] [delete]
37	45202682	Senin	21:00	21312	123,123.00	[edit] [delete]
38	321912	Senin	12:25	123	123,123.00	[edit] [delete]
39	21312	Senin	12:12	123123	12,312.00	[edit] [delete]
40	2131	Senin	02:21	12312	123,123.20	[edit] [delete]
41	1231	Senin	08:25	312	12,311.00	[edit] [delete]

Gambar 4. 22 Tampilan Kelola Data Admin

## 7) Transaksi Admin

ID TRANSAKSI	NAMA USER	NAMA KELAS	TOTAL HARGA	STATUS	TANGGAL TRANSAKSI	AKSI
	pryresbabibegum	NA	10.000	PBI	20-12-2024 09:15	<a href="#">Detail Transaksi</a>

Gambar 4. 23 Tampilan Transaksi Admin

## 8) Jadwal User

Welcome to Jadwal User

Jadwal Kelas

Senin  
Available Classes: 6 Kelas  
[VIEW KELAS](#)

Selasa  
Available Classes: 1 Kelas  
[VIEW KELAS](#)

Rabu  
Available Classes: 4 Kelas  
[VIEW KELAS](#)

Kamis  
Available Classes: 1 Kelas  
[VIEW KELAS](#)

Jumat  
Available Classes: 1 Kelas  
[VIEW KELAS](#)

Sabtu  
Available Classes: 2 Kelas  
[VIEW KELAS](#)

Minggu  
Available Classes: 0 Kelas  
[VIEW KELAS](#)

Classes on Selasa

NAMA KELAS	JAM	DESKRIPSI	INSTRUKTUR	TOTAL HARGA	AKSI
Kelas Aulas Muhiha Strassana	09:00	Chit's point, gerakan yang help nguat dengan tali elastik atau dikawat	F.ahri/Andri Saibana	10000.00	<a href="#">DETAIL</a>

Gambar 4. 24 Tampilan Jadwal User

## 9) Booking User

Welcome to Booking User

BOOKING ID	NAMA KELAS	HARGA	TOTAL HARGA	STATUS	AKSI
------------	------------	-------	-------------	--------	------

Gambar 4. 25 Tampilan Booking User

## 10) Transaksi User

TRANSAKSI ID	NAMA	NAMA KELAS	HARI	JAM	TELEAHU PADA	FOTO
21	james	Kelas Satu Sandrasana	Kamis	15:00	18-10-2021 16:26	0/0/0
22	james	Kelas Pasihindrasana	Rabu	15:00	18-10-2021 16:30	0/0/0

Gambar 4. 26 Tampilan Transaksi User

## 11) Jadwal Instruktur

ID	Nama Kelas	Instruktur	Hari	Jam	Jumlah Peserta
1	Kelas Vindrasana	Sivik Iny	Sabtu	12:00	0
26	Kelas Dhjungasana	Sivik Iny	Jumat	17:00	0
31	Kelas Satu Sandrasana	Sivik Iny	Kamis	18:00	0
32	Kelas Pasihindrasana	Sivik Iny	Rabu	15:00	0
33	Kelas Adha Mula Suanasana	Fahri Aidan Saputra	Senin	09:00	0
35	Kelas Tobasana	Sivik Iny	Senin	06:00	0
37	412021412	3123	Senin	21:03	0
38	321912	123	Senin	12:23	0
39	21312	123123	Sabtu	12:12	0
40	2191	12912	Senin	09:23	0

Gambar 4. 27 Tampilan Jadwal Instruktur

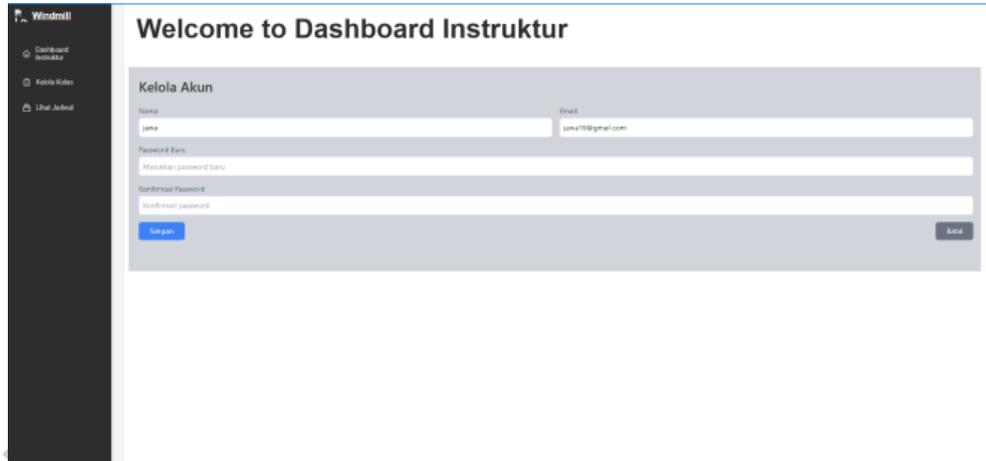
## 12) Kelola Kelas Instruktur

ID	Nama Kelas	Instruktur	Hari	Jam	Harga	Aksi
1	Kelas Vindrasana	Sivik Iny	Sabtu	12:00	Rp 75.000	[Add] [Edit] [Delete]
26	Kelas Dhjungasana	Sivik Iny	Jumat	17:00	Rp 7.000	[Add] [Edit] [Delete]
31	Kelas Satu Sandrasana	Sivik Iny	Kamis	18:00	Rp 100	[Add] [Edit] [Delete]
32	Kelas Pasihindrasana	Sivik Iny	Rabu	15:00	Rp 100.000	[Add] [Edit] [Delete]
33	Kelas Adha Mula Suanasana	Fahri Aidan Saputra	Senin	09:00	Rp 50.000	[Add] [Edit] [Delete]
35	Kelas Tobasana	Sivik Iny	Senin	06:00	Rp 50.000	[Add] [Edit] [Delete]
37	412021412	3123	Senin	21:03	Rp 123.123	[Add] [Edit] [Delete]
38	321912	123	Senin	12:23	Rp 123.123	[Add] [Edit] [Delete]
39	21312	123123	Sabtu	12:12	Rp 12.312	[Add] [Edit] [Delete]
40	2191	12912	Senin	09:23	Rp 3.123.12	[Add] [Edit] [Delete]
41	1291	312	Senin	09:23	Rp 7.231	[Add] [Edit] [Delete]

Nama Kelas:  Nama Instruktur:   
 Hari:  Jam:   
 Harga:   
 Deskripsi:

Gambar 4. 28 Tampilan Kelola Kelas Instruktur

### 13) Kelola Akun



Gambar 4. 29 Tampilan Kelola Akun User

#### 4.4 Requirements Planning ( Iteration 2)

Iterasi tahap final ini merupakan bentuk *feedback* atau umpan balik dari user setelah menerima rancangan diagram tahap i maka berikut merupakan perbaikan yang ada pada perancangan kebutuhan user tahap final setelah melewati tahap user review.

##### A. Evaluasi

Tabel 4. 13 Tabel Evaluasi User Requirement

No	Evaluasi	Penjelasan
1	perbaikan untuk beberapa logika fitur	beberapa logika fitur disesuaikan dengan kemauan user
2	penambahan fitur sorting dan search	fitur sorting dan search perlu ada semua bentuk tampilan database di dalam aplikasi
3	penambahan landing page di depan sebelum masuk ke halaman login	penambahan halaman profile di depan sebagai "wajah" kelas yoga padmaprana perlu ditambahkan

- 4 penambahan fitur chart dan card informasi
- fitur chart dan card informasi perlu ditambahkan untuk beberapa dashboard pengguna untuk menampilkan beberapa informasi.

**a) Elisitasi Tahap I**

*Tabel 4. 14 Tabel Elisitasi Final Tahap I*

<b>Functional</b>	
<b>Analisa Kebutuhan</b>	
<b>Sistem Dapat :</b>	
1	Memiliki profile landing page.
2	Memiliki fitur pendaftaran akun.
3	Dapat melakukan login ke dalam dashboard.
4	Dapat melakukan logout dari aplikasi booking.
5	Dapat menampilkan transaksi dan status transaksi untuk customer dan admin.
6	Dapat menampilkan detail transaksi dan invoice untuk customer dan admin.
7	Admin dapat melakukan konfirmasi pembayaran.
8	Memiliki halaman akses yang berbeda antara semua user (admin, customer, instruktur).
9	Memiliki fitur pengelolaan data kelas dan customer untuk admin.
10	Memiliki fitur pengelolaan akun pribadi untuk semua user.
11	Memiliki fitur pengelolaan data kelas untuk instruktur.
12	Dapat menampilkan total user untuk admin dan instruktur.
13	Dapat menampilkan grafik untuk customer dan instruktur.
14	Dapat menampilkan total kelas, kelas yang sudah selesai, dan kelas yang perlu diajar.
15	Memiliki fitur lihat detail kelas dan booking kelas untuk customer.
16	Dapat menampilkan kelas yang sudah diboeking untuk customer.
17	Memiliki fitur rubah jadwal kelas untuk instruktur yoga.

18	Memiliki halaman khusus tentang biodata pribadi instruktur yoga.
19	Memiliki fitur tracking kuota kelas dan penyelesaian kelas untuk instruktur.
20	Memiliki fitur sort dan search untuk semua database yang ditampilkan dalam aplikasi.
21	Memiliki fitur pembayaran untuk customer.
<b>Non - Functional</b>	
<b>Analisa Kebutuhan</b>	
<b>Sistem Dapat :</b>	
1	Sistem ramah pengguna
2	Sistem dapat diakses dari semua jenis web browser (minimal chrome dan firefox)
3	Sistem harus dapat diakses secara online sepanjang hari

#### b) Elisitasi Tahap II

Tabel 4. 15 Tabel Elisitasi Final Tahap II

<b>Functional</b>					
<b>Analisa Kebutuhan</b>					
<b>Sistem Dapat :</b>			<b>M</b>	<b>D</b>	<b>I</b>
1	Memiliki profile landing page.	√			
2	Memiliki fitur pendaftaran akun.	√			
3	Dapat melakukan login ke dalam dashboard.	√			
4	Dapat melakukan logout dari aplikasi booking.	√			
5	Dapat menampilkan transaksi dan status transaksi untuk customer dan admin.	√			
6	Dapat menampilkan detail transaksi dan invoice untuk customer dan admin.	√			
7	Admin dapat melakukan konfirmasi pembayaran.				√
8	Memiliki halaman akses yang berbeda antara semua user (admin, customer, instruktur).	√			
9	Memiliki fitur pengelolaan data kelas dan customer untuk		√		

	admin.			
10	Memiliki fitur pengelolaan akun pribadi untuk semua user.	√		
11	Memiliki fitur pengelolaan data kelas untuk instruktur.		√	
12	Dapat menampilkan total user untuk admin dan instruktur.		√	
13	Dapat menampilkan grafik untuk customer dan instruktur.	√		
14	Dapat menampilkan total kelas, kelas yang sudah selesai, dan kelas yang perlu diajar.	√		
15	Memiliki fitur lihat detail kelas dan booking kelas untuk customer.		√	
16	Dapat menampilkan kelas yang sudah dibooking untuk customer.	√		
17	Memiliki fitur rubah jadwal kelas untuk instruktur yoga.			√
18	Memiliki halaman khusus tentang biodata pribadi instruktur yoga.			√
19	Memiliki fitur tracking kuota kelas dan penyelesaian kelas untuk instruktur.		√	
20	Memiliki fitur sort dan search untuk semua database yang ditampilkan dalam aplikasi.		√	
21	Memiliki fitur pembayaran untuk customer.	√		
<b>Non - Functional</b>				
<b>Analisa Kebutuhan</b>				
<b>Sistem Dapat :</b>		<b>M</b>	<b>D</b>	<b>I</b>
1	Sistem ramah pengguna	√		
2	Sistem dapat diakses dari semua jenis web browser (minimal chrome dan firefox)		√	
3	Sistem harus dapat diakses secara online sepanjang hari			√

### c) Elisitasi Tahap III

Tabel 4. 16 Tabel Elisitasi Final Tahap III

<b>Functional</b>			
<b>Analisa Kebutuhan</b>	<b>T</b>	<b>O</b>	<b>E</b>

Sistem Dapat :		H	M	L	H	M	L	H	M	L
1	Memiliki profile landing page.		√			√			√	
2	Memiliki fitur pendaftaran akun.		√			√			√	
3	Dapat melakukan login ke dalam dashboard.		√			√				√
4	Dapat melakukan logout dari aplikasi booking.			√			√		√	
5	Dapat menampilkan transaksi dan status transaksi untuk customer dan admin.			√		√			√	
6	Dapat menampilkan detail transaksi dan invoice untuk customer dan admin.		√			√				√
7	Memiliki halaman akses yang berbeda antara semua user (admin, customer, instruktur).			√		√				√
8	Memiliki fitur pengelolaan data kelas dan customer untuk admin.			√		√			√	
9	Memiliki fitur pengelolaan akun pribadi untuk semua user.		√			√				√
10	Memiliki fitur pengelolaan data kelas untuk instruktur.			√			√		√	
11	Dapat menampilkan total user untuk admin dan instruktur.			√		√			√	
12	Dapat menampilkan grafik untuk customer dan instruktur.		√				√		√	
13	Dapat menampilkan total kelas, kelas yang sudah selesai, dan kelas yang perlu diajar.			√		√			√	
14	Memiliki fitur lihat detail kelas dan booking kelas untuk customer.		√			√				√
15	Dapat menampilkan kelas yang			√			√		√	

	sudah dibooking untuk customer.									
16	Memiliki fitur tracking kuota kelas dan penyelesaian kelas untuk instruktur.		√		√			√		
17	Memiliki fitur sort dan search untuk semua database yang ditampilkan dalam aplikasi.	√				√		√		
18	Memiliki fitur pembayaran untuk customer.		√		√			√		
<b>Non - Functional</b>										
<b>Analisa Kebutuhan</b>		<b>T</b>			<b>O</b>			<b>E</b>		
<b>Sistem Dapat :</b>		<b>H</b>	<b>M</b>	<b>L</b>	<b>H</b>	<b>M</b>	<b>L</b>	<b>H</b>	<b>M</b>	<b>L</b>
1	Sistem ramah pengguna		√			√			√	
2	Sistem dapat diakses dari semua jenis web browser (minimal chrome dan firefox)		√			√			√	

**d) Elisitasi Final**

*Tabel 4. 17 Tabel Elisitasi Final*

<b>Functional</b>	
<b>Analisa Kebutuhan</b>	
<b>Sistem Dapat :</b>	
1	Memiliki profile landing page.
2	Memiliki fitur pendaftaran akun.
3	Dapat melakukan login ke dalam dashboard.
4	Dapat melakukan logout dari aplikasi booking.
5	Dapat menampilkan transaksi dan status transaksi untuk customer dan admin.
6	Dapat menampilkan detail transaksi dan invoice untuk customer dan admin.
7	Memiliki halaman akses yang berbeda antara semua user (admin,

	customer, instruktur).
8	Memiliki fitur pengelolaan data kelas dan customer untuk admin.
9	Memiliki fitur pengelolaan akun pribadi untuk semua user.
10	Memiliki fitur pengelolaan data kelas untuk instruktur.
11	Dapat menampilkan total user untuk admin dan instruktur.
12	Dapat menampilkan grafik untuk customer dan instruktur.
13	Dapat menampilkan total kelas, kelas yang sudah selesai, dan kelas yang perlu diajar.
14	Memiliki fitur lihat detail kelas dan booking kelas untuk customer.
15	Dapat menampilkan kelas yang sudah diboooking untuk customer.
16	Memiliki fitur tracking kuota kelas dan penyelesaian kelas untuk instruktur.
17	Memiliki fitur sort dan search untuk semua database yang ditampilkan dalam aplikasi.
18	Memiliki fitur pembayaran untuk customer.
<b>Non - Functional</b>	
Analisa Kebutuhan	
Sistem Dapat :	
1	Sistem ramah pengguna
2	Sistem dapat diakses dari semua jenis web browser (minimal chrome dan firefox)

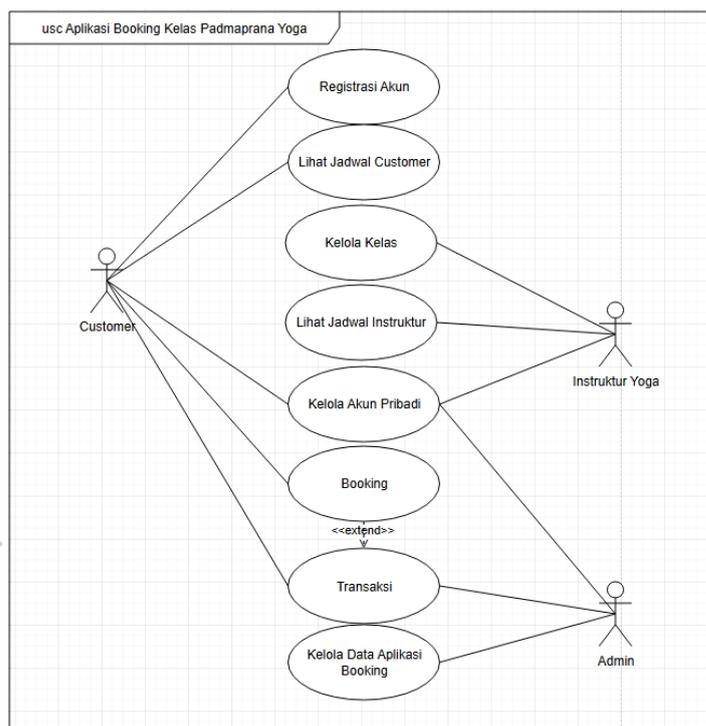
#### 4.4.1 Perancangan Use Case Diagram (Iterasi 2)

##### A. Evaluasi

Tabel 4. 18 Tabel Evaluasi Use Case Diagram

No	Evaluasi	Penjelasan
1	Pemecahan lihat jadwal untuk customer dan instruktur	Perubahan logika lihat jadwal untuk kedua user membuat fitur lihat jadwal harus dipecah menjadi 2 use case task

a) Rancangan Final Use Case Diagram



Gambar 4. 30 Use Case Diagram Final

#### 4.4.2 Perancangan Spesifikasi Use Case Diagram (Iterasi 2)

##### A. Evaluasi

Tabel 4. 19 Tabel Evaluasi Spesifikasi Use Case Diagram

No	Evaluasi	Penjelasan
1	Spesifikasi Use Case Lihat Jadwal Customer	penambahan fungsi untuk melakukan pemilihan dan pembookingan kelas
2	Spesifikasi Use Case Lihat Jadwal Instruktur	penambahan fungsi untuk melakukan penyelesaian kelas
3	Spesifikasi Use Case Kelola Akun Pribadi	Penambahan fungsi mematikan session ketika sudah selesai melakukan pengelolaan data akun pribadi

4	Spesifikasi Use Case Booking	Perubahan fungsi utama menjadi proses pembayaran dan menampilkan table transaksi sementara setelah selesai melakukan booking
5	Spesifikasi Use Case Transaksi	perubahan fungsi utama menjadi untuk menampilkan daftar transaksi dan menampilkan media invoice

**a) Spesifikasi Use Case Registrasi Akun Final**

Dalam perancangan spesifikasi use case registrasi tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem

**b) Spesifikasi Use Case Kelola Akun Pribadi Final**

Pada perancangan spesifikasi use case *Kelola Akun Pribadi*, terdapat penambahan fungsi untuk mematikan sesi pengguna secara otomatis setelah selesai mengelola data akun pribadi. Penambahan ini bertujuan untuk meningkatkan keamanan data pengguna dan mencegah akses yang tidak sah.

Tabel 4. 20 Spesifikasi Use Case Kelola Akun Pribadi Final

<b>Use Case Name</b>	Use Case Kelola Akun Pribadi
<b>Actor</b>	Customer,Instruktur, Admin
<b>Deskripsi</b>	User dapat melihat dan memperbaiki informasi akun mereka, seperti nama, email, kata sandi, atau informasi lainnya.
<b>Pre-Condition</b>	User sudah terdaftar dan login
<b>Post-</b>	Informasi akun User berhasil diperbaharui

Condition		
Scenario	User	System
	1. User membuka halaman kelola Akun Pribadi.	2. Menampilkan halaman profil
	3. User mengubah informasi yang diperlukan.	
	4. User menekan tombol "Simpan".	5. menyimpan data User baru.
	6. Customer berhasil mengubah profil	7. menampilkan pesan bahwa perubahan berhasil disimpan.
		8. Mematikan Session.
Alternative Flows	—	

c) Spesifikasi Use Case Lihat Jadwal Customer Final

Dalam perancangan spesifikasi use case *Lihat Jadwal Customer*, terdapat penambahan fungsi untuk memungkinkan pengguna melakukan pemilihan dan pembookingan kelas secara langsung melalui antarmuka jadwal. Perubahan ini bertujuan untuk meningkatkan kenyamanan dan efisiensi pengguna dalam melakukan reservasi kelas sesuai kebutuhan.

Tabel 4. 21 Spesifikasi Use Case Lihat Jadwal Customer Final

<b>Use Case Name</b>	Use case lihat jadwal customer
<b>Actor</b>	Customer
<b>Deskripsi</b>	Customer dapat melihat jadwal kelas yang tersedia, termasuk informasi seperti nama kelas, hari, jam, instruktur, dan deskripsi serta melakukan

	pembukaan.	
<b>Pre-Condition</b>	Jadwal kelas telah diatur di dalam sistem	
<b>Post-Condition</b>	Jadwal kelas ditampilkan kepada Customer	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Customer membuka halaman jadwal kelas.	2.Menampilkan halaman jadwal
	3.Customer memilih hari dan menekan tombol "lihat kelas"	4.Menampilkan kelas yang tersedia di hari itu
	5. Customer memilih kelas dan menekan "save to book"	6.Menampilkan detail kelas yang dipilih beserta data Customer
	7.Customer menekan tombol "Book"	8.menyimpan data booking ke database
		9.Menampilkan notif kelas berhasil di booking
<b>Alternative Flows</b>	-	

d) Spesifikasi Use Case Lihat Jadwal Instruktur Final

Dalam perancangan spesifikasi use case *Lihat Jadwal Instruktur*, terjadi penambahan fungsi untuk mendukung instruktur menyelesaikan kelas yang telah berlangsung. Fitur ini ditambahkan untuk memberikan fleksibilitas kepada instruktur dalam mengelola status kelas yang mereka ajar.

Tabel 4. 22 Spesifikasi Use Case Lihat Jadwal Instruktur Final

<b>Use Case Name</b>	Use case lihat jadwal instruktur	
<b>Actor</b>	Instruktur Yoga	
<b>Deskripsi</b>	Instruktur dapat melihat jadwal kelas yang tersedia, termasuk informasi seperti nama kelas, hari, jam, instruktur, kuota, dan deskripsi serta melakukan penyelesaian kelas.	
<b>Pre-Condition</b>	Jadwal kelas telah diatur di dalam sistem	
<b>Post-Condition</b>	Jadwal kelas ditampilkan kepada Instruktur Yoga	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1. Instruktur membuka halaman jadwal kelas.	2. Menampilkan halaman jadwal
	3. Instruktur memilih kelas yang ingin di selesaikan	
	4. Instruktur menekan tombol "selesai"	5. Menghapus seluruh data booking customer yang melakukan pemesanan pada hari itu
		6. menampilkan table data terbaru
<b>Alternative Flows</b>	-	

e) Spesifikasi Use Case Kelola Kelas Final

Dalam perancangan spesifikasi use case kelola kelas tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem

f) Spesifikasi Use Case Booking Final

Dalam perancangan spesifikasi use case *Booking*, fungsi utama mengalami perubahan, yaitu menjadi proses pembayaran. Selain itu, sistem juga menampilkan tabel transaksi sementara setelah proses booking selesai dilakukan. Perubahan ini bertujuan untuk memberikan transparansi dan kemudahan kepada pengguna dalam memeriksa detail transaksi mereka.

Tabel 4. 23 Spesifikasi Use Case Booking Kelas Final

<b>Use Case Name</b>	Use Case Booking	
<b>Actor</b>	Customer	
<b>Deskripsi</b>	Pengguna dapat membayar kelas yang sudah di booking.	
<b>Pre-Condition</b>	Customer sudah login dan kelas tersedia	
<b>Post-Condition</b>	kelas berhasil di bayar oleh customer	
<b>Scenario</b>	<b>Customer</b>	<b>System</b>
	1.Customer membuka halaman booking.	2.Menampilkan halaman booking
	3.Customer memilih kelas yang ingin dibayar dan menekan tombol "bayar".	4.Menampilkan detail pembayaran kelas yang dipilih beserta metode pembayaran
	5.Customer memilih metode pembayaran	
	6.Customer membayar	7. Mengubah status data menjadi "paid"
		8. Menampilkan notif pembayaran telah berhasil dan mengirim invoice lewat email
<b>Alternative Flows</b>	Jika Customer tidak melakukan pembayaran dalam waktu yang ditentukan maka pembayaran akan terbatalkan	

g) Spesifikasi Use Case Transaksi Final

Pada perancangan spesifikasi use case Transaksi, fungsi utama mengalami perubahan, yaitu menjadi menampilkan daftar transaksi yang telah dilakukan oleh pengguna. Selain itu, ditambahkan fitur untuk menampilkan media invoice, sehingga mempermudah pengguna dalam melakukan dokumentasi dan pelacakan transaksi.

Tabel 4. 24 Spesifikasi Use Case Transaksi Final

<b>Use Case Name</b>	Use Case Transaksi	
<b>Actor</b>	Customer, Admin	
<b>Deskripsi</b>	Detail Invoice setelah Customer melakukan pembayaran	
<b>Pre-Condition</b>	Customer telah melakukan pemesanan kelas dan membayar	
<b>Post-Condition</b>	Invoice dapat di lihat	
<b>Scenario</b>	User	System
	1.User membuka halaman transaksi.	2.Menampilkan Halaman Trasnaksi
	3.User memilih bekas bukti transaksi atau invoice	4.Memapilkan detail traksaksi atau invoice
<b>Alternative Flows</b>	-	

h) Spesifikasi Use Case Kelola Data Aplikasi Booking Final

Dalam perancangan spesifikasi use case kelola data aplikasi booking tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem

### 4.4.3 Perancangan Activity Diagram (Iterasi 2)

#### A. Evaluasi

Tabel 4. 25 Tabel Evaluasi Activity Diagram

No	Evaluasi	Penjelasan
1	Activity Diagram Lihat Jadwal Customer	Pada tahap final, ditambahkan aktivitas untuk memungkinkan pelanggan melakukan pemilihan dan pembookingan kelas langsung dari jadwal.
2	Activity Diagram Lihat Jadwal Instruktur	Pada tahap final, instruktur dapat menyelesaikan kelas langsung melalui antarmuka jadwal setelah proses pengajaran selesai.
3	Activity Diagram Kelola Akun Pribadi	Pada tahap final, ditambahkan fungsi untuk mematikan sesi pengguna secara otomatis setelah selesai mengelola data akun pribadi, meningkatkan keamanan.
4	Activity Diagram Booking Kelas	Pada tahap final, langkah pembayaran diintegrasikan dalam alur proses booking, termasuk tabel transaksi sementara setelah proses selesai.
5	Activity Diagram Transaksi	Pada tahap final, ditambahkan langkah untuk menampilkan daftar transaksi yang telah dilakukan, serta menyediakan media invoice bagi pengguna

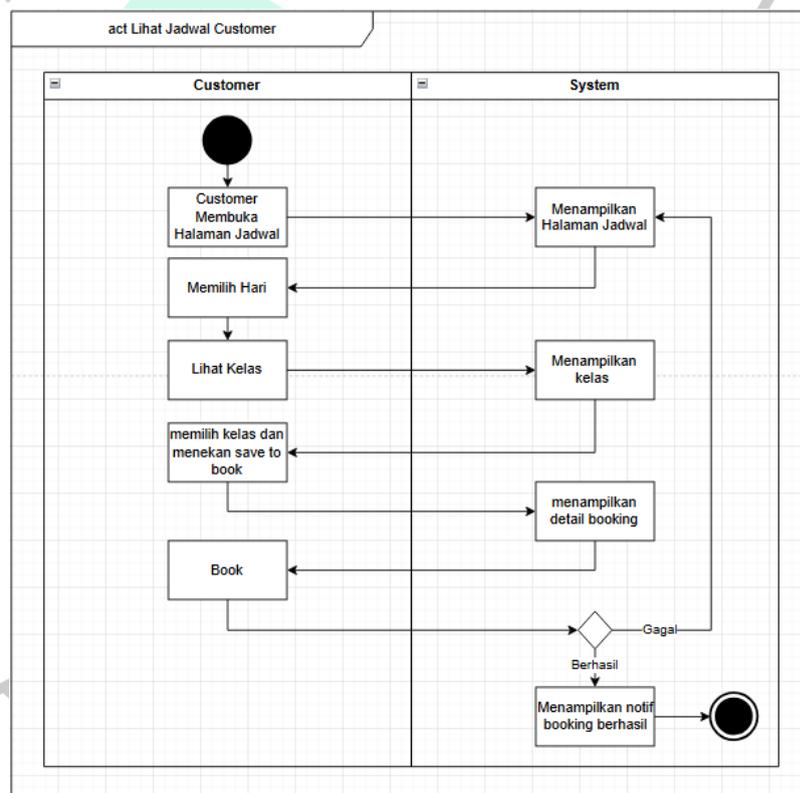
a) Activity Diagram Registrasi Akun Final

Dalam perancangan activity diagram registrasi akun tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk

menjaga konsistensi sistem

b) Activity Diagram Lihat Jadwal Customer Final

Pada perancangan Activity Diagram Lihat Jadwal Customer, fungsi utama mengalami perubahan pada tahap final dengan penambahan aktivitas untuk memungkinkan pelanggan melakukan pemilihan dan pembookingan kelas secara langsung melalui jadwal yang tersedia. Perubahan ini bertujuan untuk memberikan kemudahan dan fleksibilitas bagi pelanggan dalam mengakses layanan.

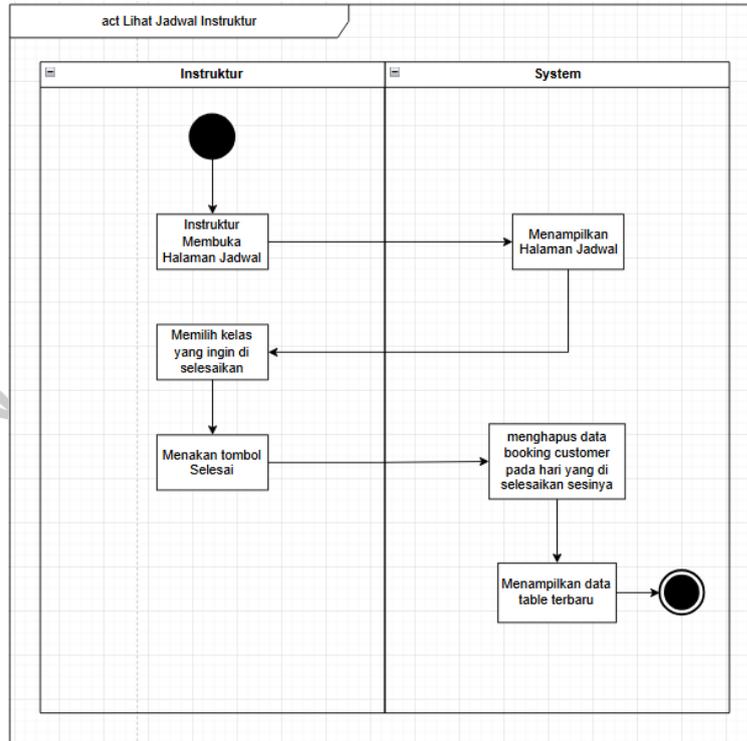


Gambar 4. 31 Activity Diagram Lihat Jadwal Customer Final

c) Activity Diagram Lihat Jadwal Instruktur Final

Pada perancangan Activity Diagram Lihat Jadwal Instruktur, terdapat penambahan aktivitas pada tahap final yang memungkinkan instruktur menyelesaikan kelas langsung melalui antarmuka jadwal setelah proses pengajaran selesai. Perubahan ini ditambahkan untuk mendukung pengelolaan kelas secara

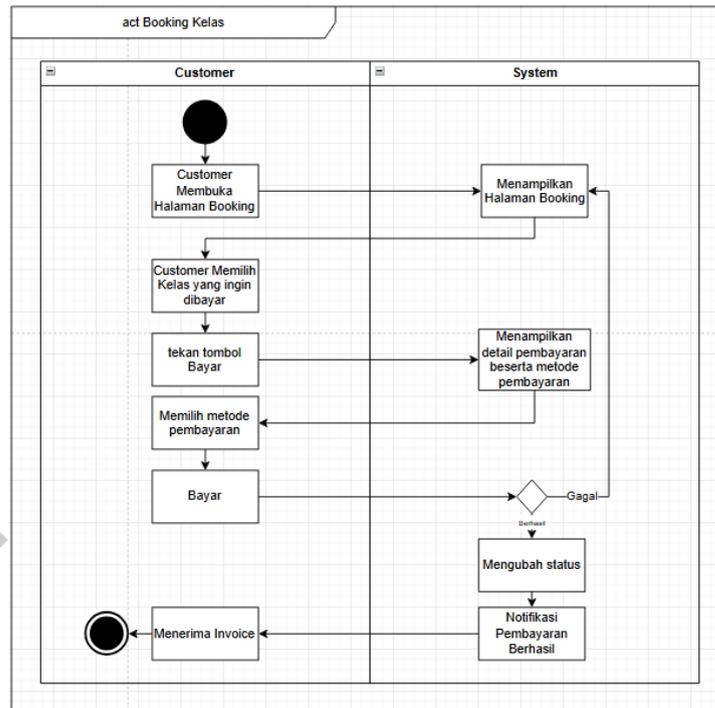
efisien oleh instruktur.



Gambar 4. 32 Activity Diagram Lihat Jadwal Instruktur Final

d) Activity Diagram Booking Kelas Final

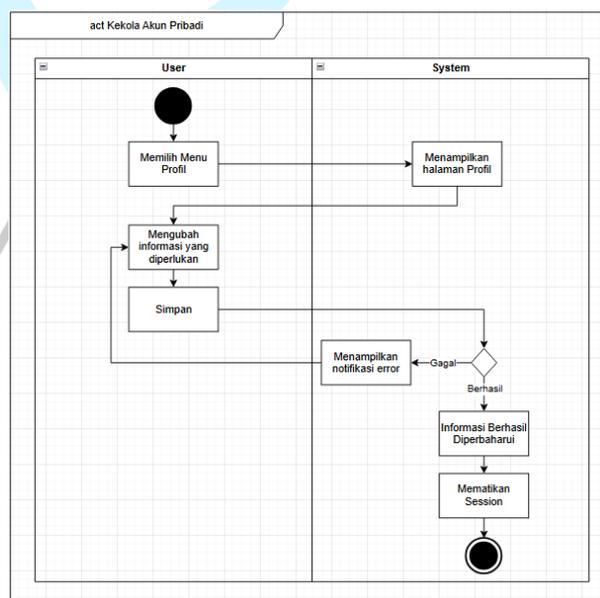
Pada perancangan *Activity Diagram Booking Kelas*, terdapat integrasi langkah pembayaran ke dalam alur proses booking pada tahap final. Selain itu, sistem menampilkan tabel transaksi sementara setelah proses booking selesai dilakukan. Perubahan ini bertujuan untuk memberikan transparansi dan kemudahan bagi pelanggan dalam menyelesaikan transaksi.



Gambar 4. 33 Activity Diagram Booking Kelas Final

e) Activity Diagram Kelola Akun Pribadi Final

Pada perancangan *Activity Diagram Kelola Akun Pribadi*, terdapat penambahan fungsi pada tahap final untuk mematikan sesi pengguna secara otomatis setelah selesai mengelola data akun pribadi. Penambahan ini bertujuan untuk meningkatkan keamanan data pengguna dan mencegah akses yang tidak sah.



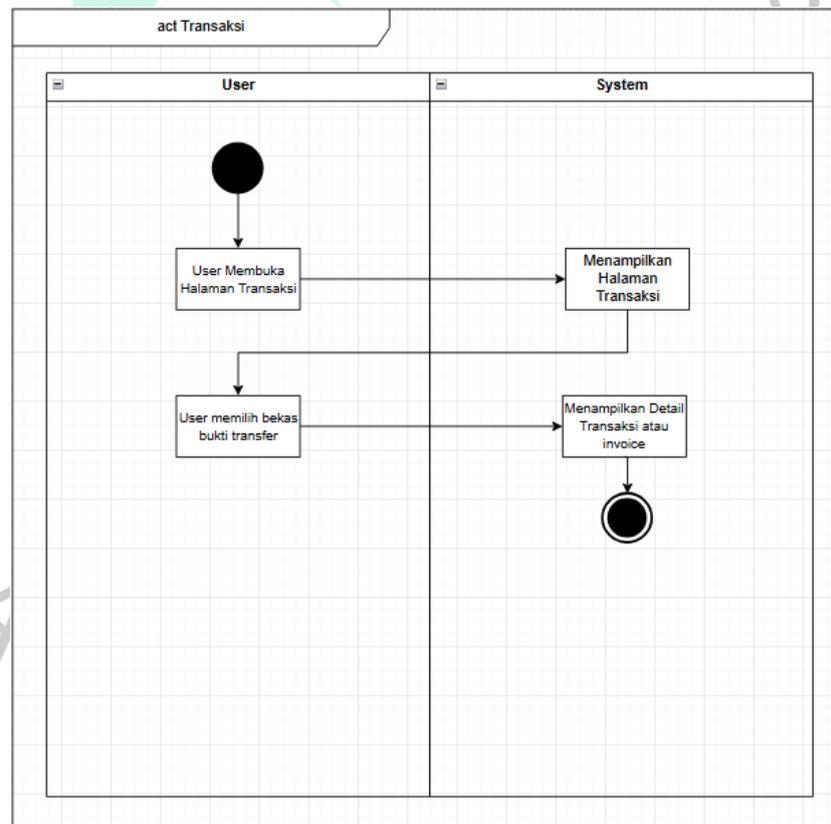
Gambar 4. 34 Activity Diagram Kelola Akun Pribadi Final

f) Activity Diagram Kelola Kelas Final

Dalam perancangan activity diagram kelola kelas tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem.

g) Activity Diagram Kelola Data Aplikasi Booking Final

Dalam perancangan activity diagram kelola data aplikasi booking tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem



Gambar 4. 35 Activity Diagram Transaksi Final

h) Activity Diagram Transaksi Final

Pada perancangan *Activity Diagram Transaksi*, terdapat penambahan fungsi pada tahap final untuk mematikan sesi pengguna secara otomatis setelah selesai mengelola data akun

pribadi. Penambahan ini bertujuan untuk meningkatkan keamanan data pengguna dan mencegah akses yang tidak sah.

#### 4.4.4 Perancangan Sequence Diagram (Iterasi 2)

##### A. Evaluasi

Tabel 4. 26 Tabel Evaluasi Use Case Diagram

No	Evaluasi	Penjelasan
1	Sequence Diagram Lihat Jadwal Customer	Pada tahap final, pelanggan dapat memilih jadwal kelas dan melakukan booking langsung melalui daftar jadwal yang ditampilkan.
2	Sequence Diagram Lihat Jadwal Instruktur	Pada tahap final, instruktur dapat memilih jadwal kelas dan menandai kelas sebagai selesai setelah pengajaran berakhir.
3	Sequence Diagram Kelola Akun Pribadi	Pada tahap final, terdapat verifikasi tambahan untuk memastikan sesi pengguna ditutup otomatis setelah selesai mengelola akun.
4	Sequence Diagram Booking Kelas	Pada tahap final, langkah pembayaran diintegrasikan dalam alur sequence setelah booking, termasuk konfirmasi dan tampilan tabel transaksi sementara.
5	Sequence Diagram Transaksi	Pada tahap final, ditambahkan langkah untuk menampilkan daftar transaksi, menampilkan media invoice, dan konfirmasi status pembayaran.

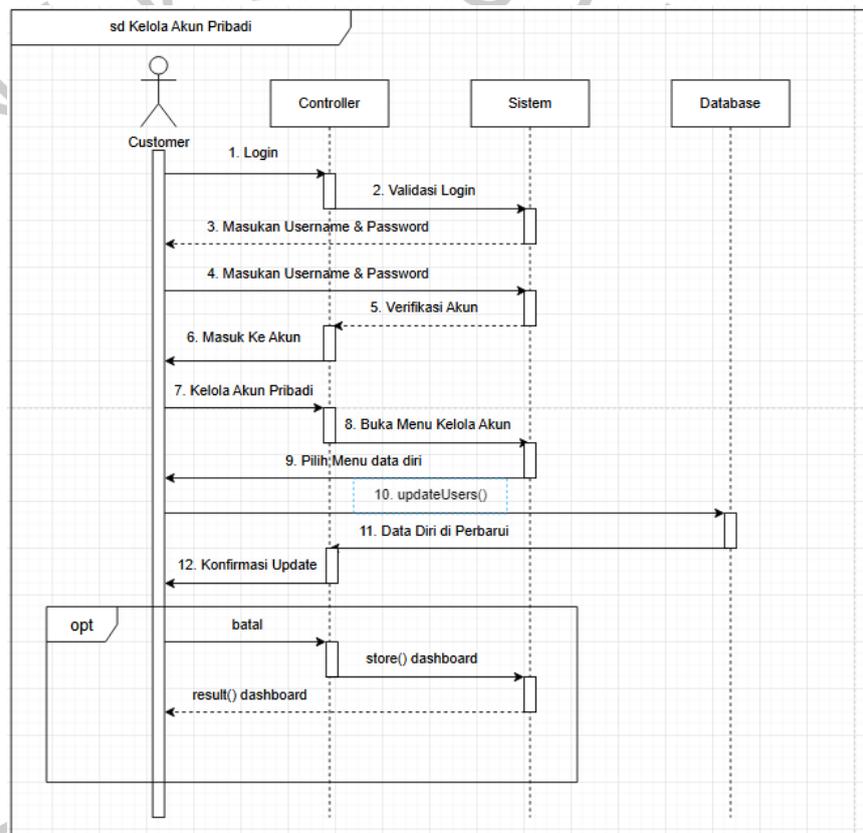
##### a) Sequence Diagram Registrasi Akun Final

Dalam perancangan sequence diagram registrasi akun tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap

1, untuk menjaga konsistensi sistem.

b) Sequence Diagram Kelola Akun Pribadi Final

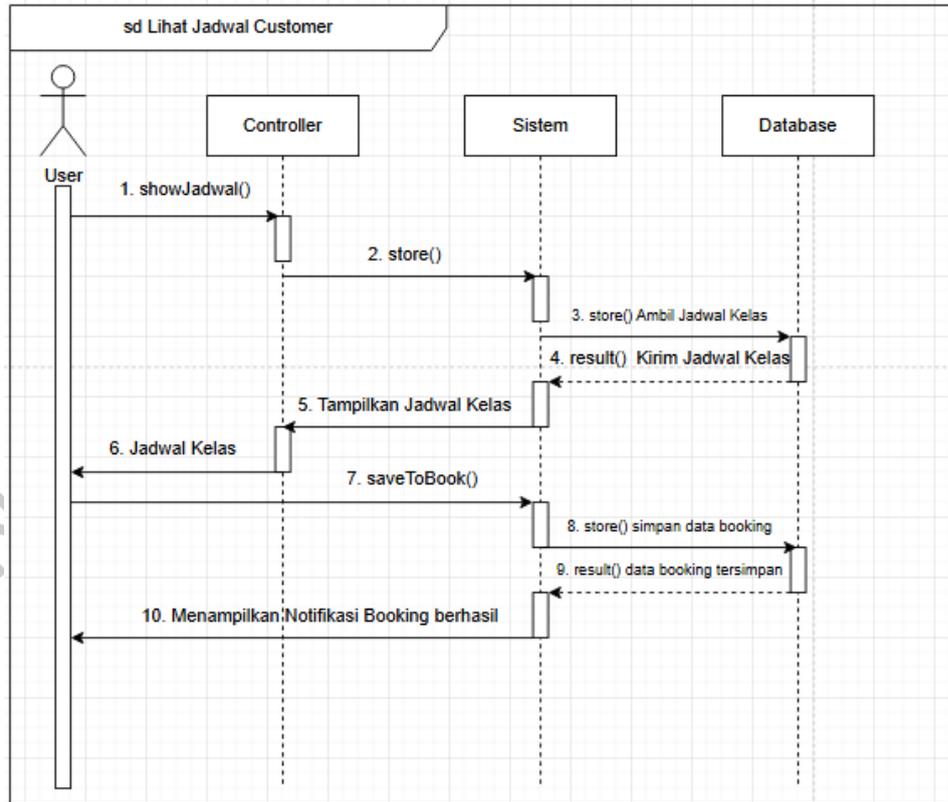
Pada perancangan *Sequence Diagram Kelola Akun Pribadi*, ditambahkan langkah verifikasi tambahan pada tahap final untuk memastikan sesi pengguna ditutup otomatis setelah selesai mengelola akun. Perubahan ini meningkatkan keamanan data pribadi pengguna.



Gambar 4. 36 Sequence Diagram Kelola Akun Pribadi Final

c) Sequence Diagram Lihat Jadwal Customer Final

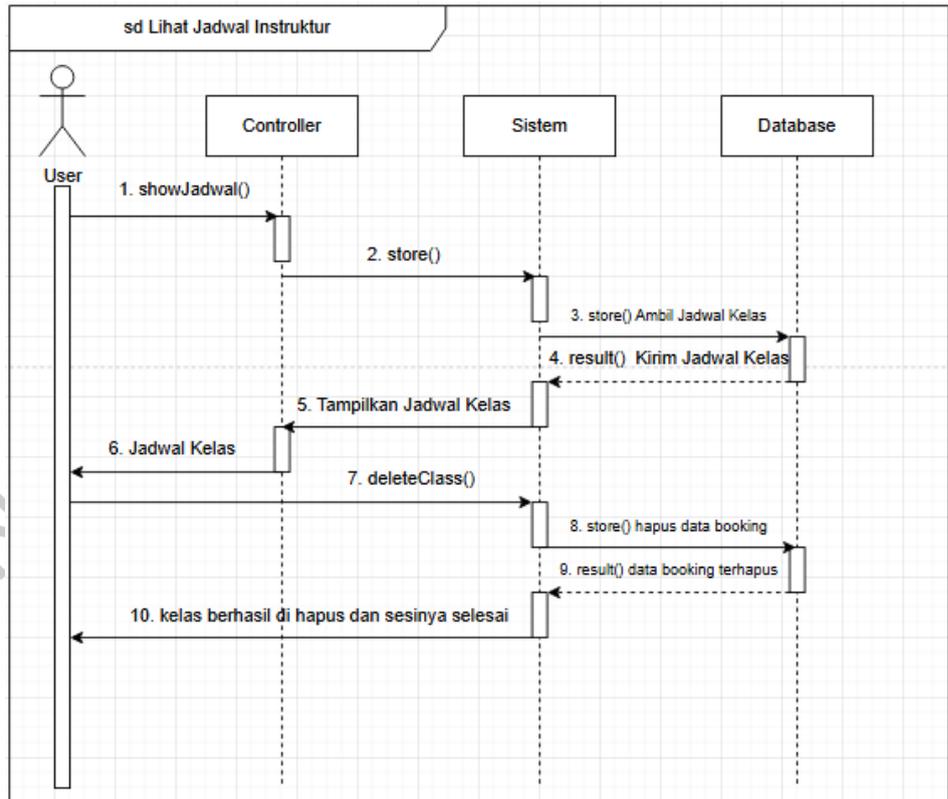
Pada perancangan *Sequence Diagram Lihat Jadwal Customer*, tahap final mencakup penambahan interaksi yang memungkinkan pelanggan memilih jadwal kelas dan melakukan booking langsung dari daftar jadwal yang ditampilkan. Perubahan ini mendukung pengalaman pengguna yang lebih intuitif dan efisien.



Gambar 4. 37 Sequence Diagram Lihat Jadwal Customer Final

d) Sequence Diagram Lihat Jadwal Instruktur Final

Pada perancangan *Sequence Diagram Lihat Jadwal Instruktur*, tahap final memungkinkan instruktur memilih jadwal kelas dan menandai kelas sebagai selesai setelah proses pengajaran berakhir. Perubahan ini mempermudah instruktur dalam mengelola status kelas mereka.



Gambar 4. 38 Sequence Diagram Lihat Jadwal Instruktur Final

e) Sequence Diagram Kelola Kelas Final

Dalam perancangan sequence diagram kelola kelas tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem.

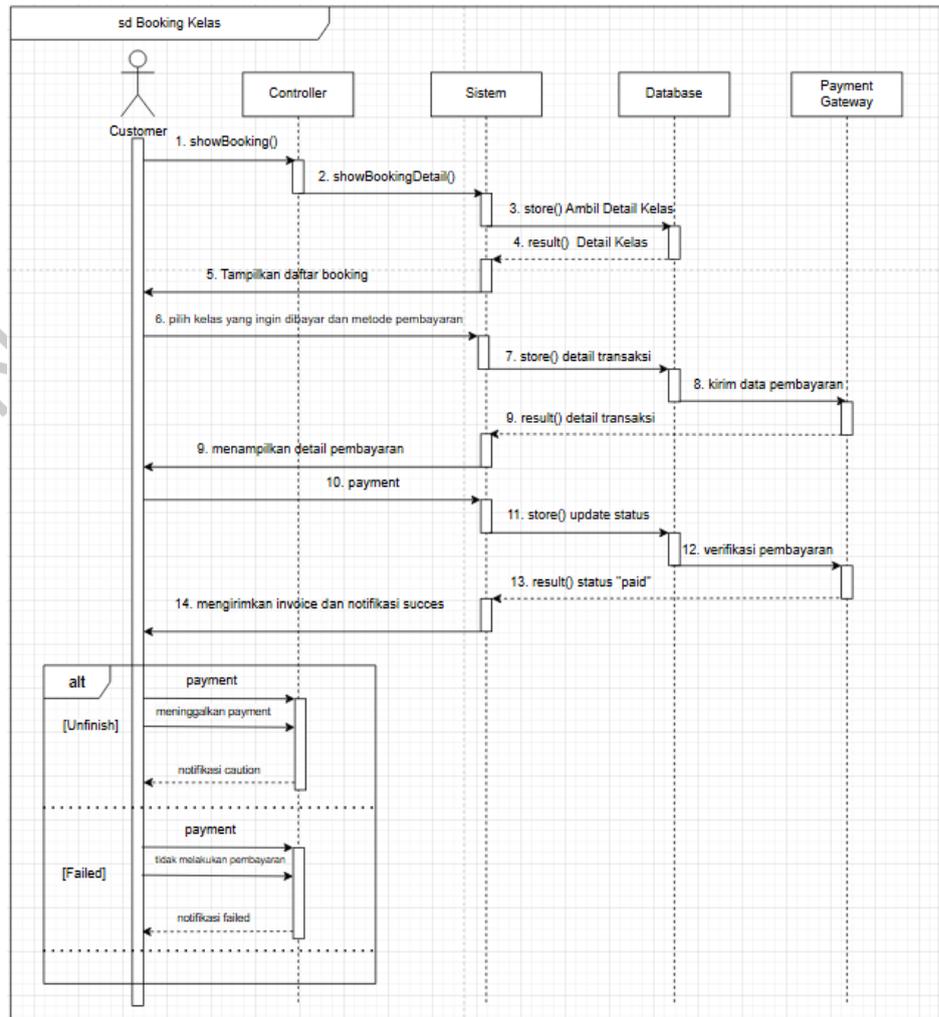
f) Sequence Diagram Kelola Data Aplikasi Final

Dalam perancangan sequence diagram kelola data Aplikasi tahap 1 tidak menghasilkan perubahan pada tahap final, rancangan ini tetap dipertahankan prosesnya yang sudah dibangun pada tahap 1, untuk menjaga konsistensi sistem.

g) Sequence Diagram Booking Kelas Final

Pada perancangan *Sequence Diagram Booking Kelas*, tahap final mencakup integrasi langkah pembayaran setelah proses booking dilakukan. Selain itu, sistem menampilkan tabel transaksi

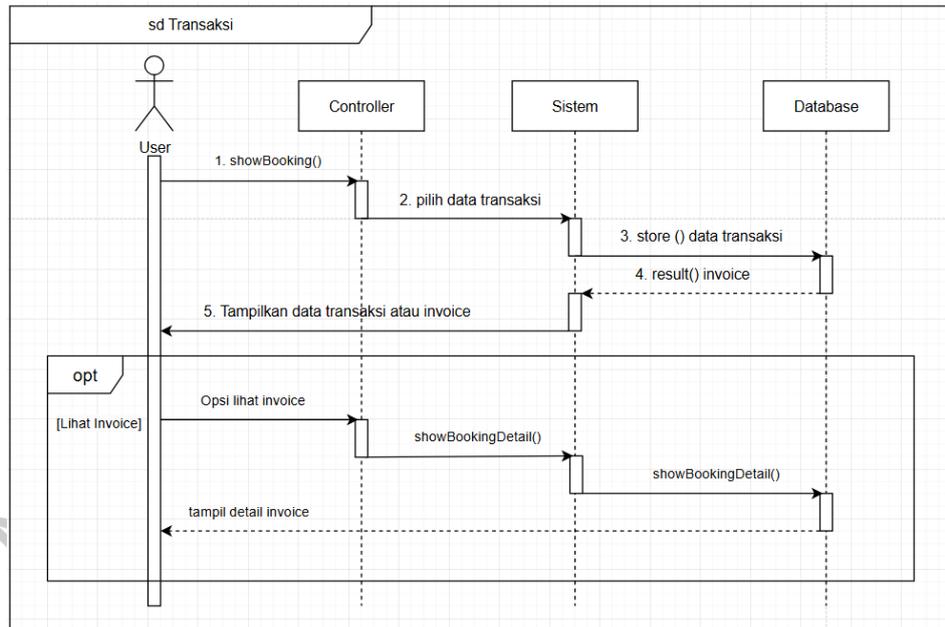
sementara sebagai bagian dari konfirmasi proses booking. Perubahan ini menambah transparansi bagi pelanggan.



Gambar 4. 39 Sequence Diagram Booking Kelas Final

h) Sequence Diagram Transaksi Final

Pada perancangan *Sequence Diagram Transaksi*, tahap final mencakup penambahan langkah untuk menampilkan daftar transaksi secara rinci. Selain itu, disediakan fitur media invoice untuk membantu pengguna mendokumentasikan dan melacak transaksi yang telah dilakukan.



Gambar 4. 40 Sequence Diagram Transaksi Final

#### 4.4.5 Perancangan Class Diagram (Iterasi 2)

Tidak ada perubahan yang signifikan

#### 4.4.6 Perancangan Spesifikasi Database (Iterasi 2)

Tidak ada perubahan yang signifikan

### 4.5 Prototipe User Interface (Iterasi 2)

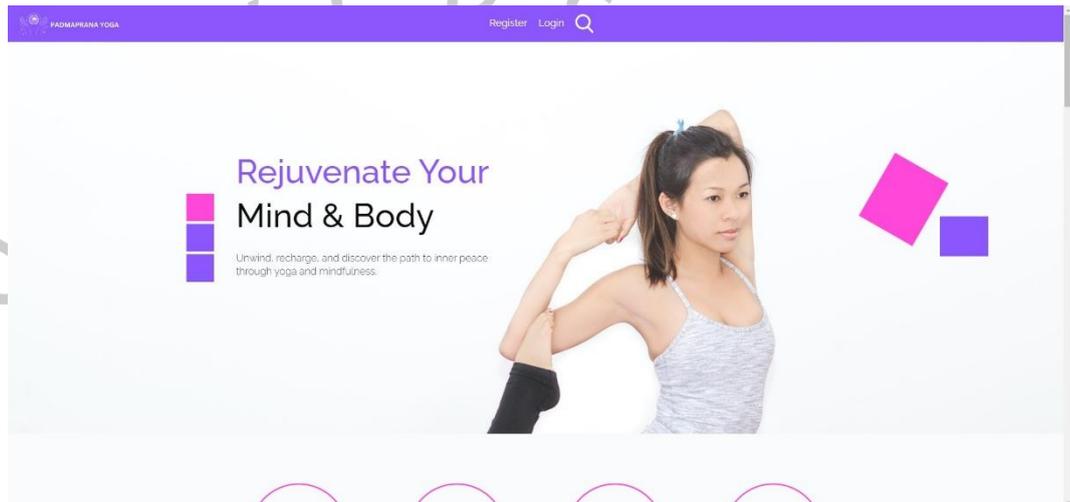
Tabel 4. 27 Evaluasi Prototipe User Interface

No	Evaluasi	Penjelasan
1	Login	Perbaikan UI menjadi warna yang lebih cerah dan konsisten dengan warna register
2	Dashboard Admin	Perbaikan UI secara keseluruhan dengan mengkonsistensikan desain dan penambahan fitur sort dan search

3	Dashboard Customer	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain dan penambahan fitur sort, search, chart, informasi mengenai transaksi yang sedang dilakukan, serta tabel jadwal kelas terdekat
4	Dashboard Instruktur	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain dan penambahan fitur chart
5	Kelola Data Admin	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain
6	Kelola Data kelas Instruktur	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain
7	Transaksi	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain serta mengganti fitur menjadi menampilkan daftar transaksi dan button lihat invoice dengan lebih detail
8	Lihat Jadwal	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain serta perubahan fungsi utama menjadi untuk menampilkan daftar transaksi dan menampilkan media invoice
9	Booking	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain serta perubahan fungsi utama menjadi fitur pembayaran
10	Kelola data akun pribadi	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain serta menambahkan button batal untuk

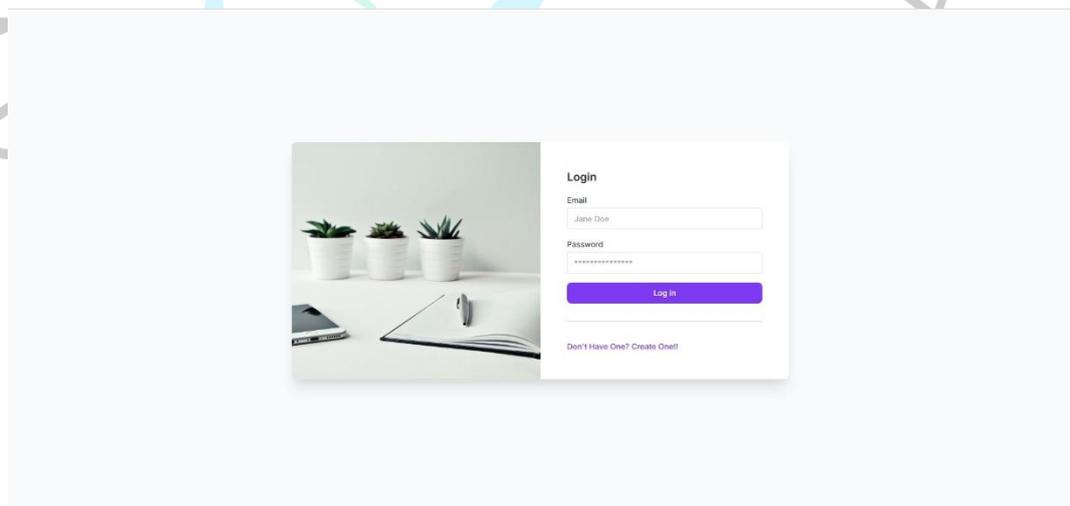
		membatalakan proses keloal akun pribadi
11	Form Tambah data	Perbaiki UI secara keseluruhan dengan mengkonsistensikan desain secara menyeluruh

1) Profile / Landing Page



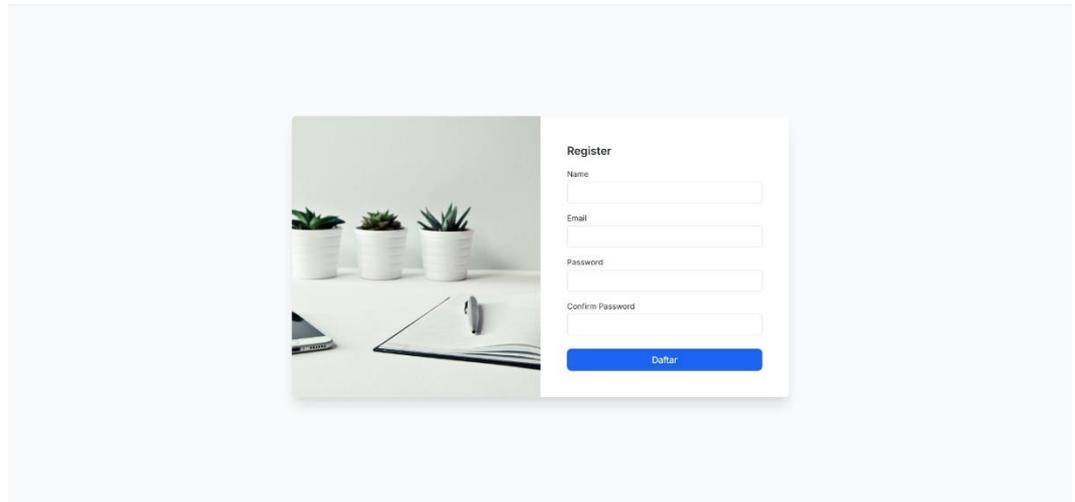
Gambar 4. 41 Profile / Landing Page

2) Login



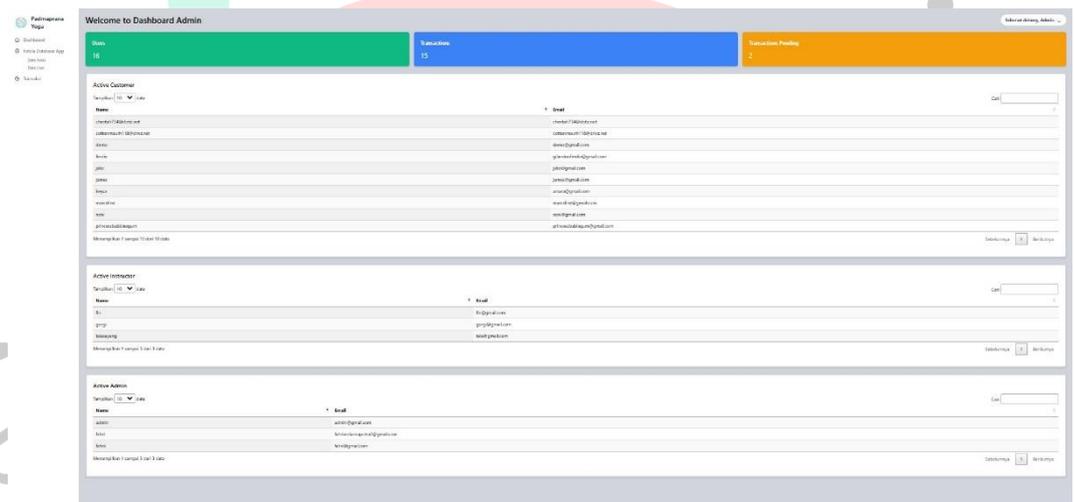
Gambar 4. 42 Login Final

### 3) Register



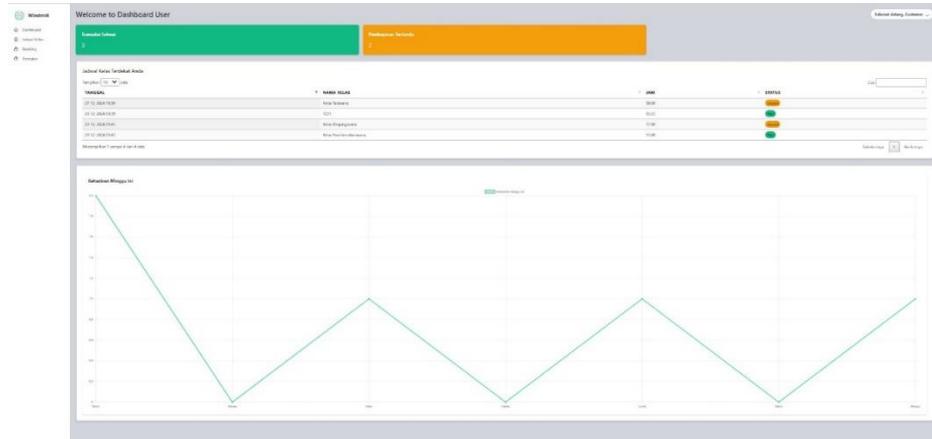
Gambar 4. 43 Register Final

### 4) DashboardAdmin



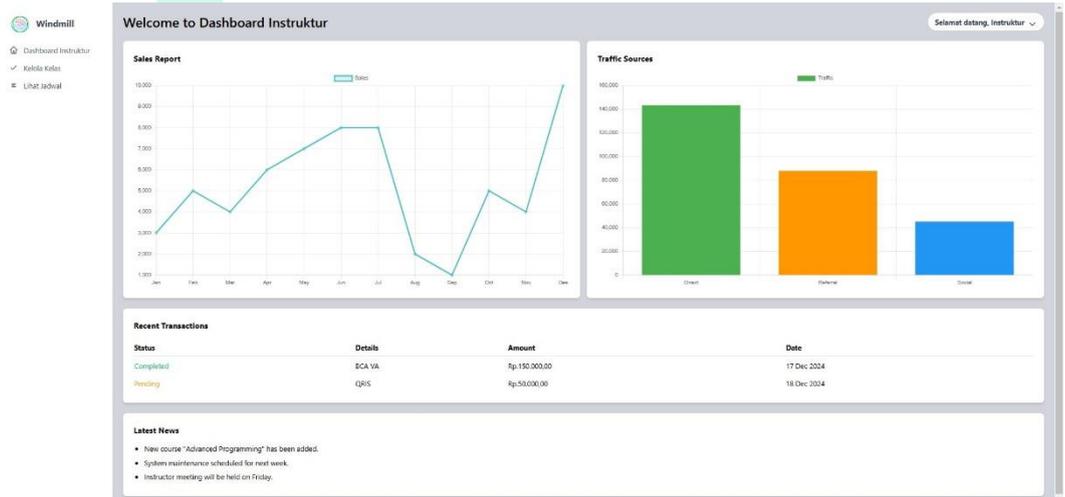
Gambar 4. 44 Dashboard Admin Final

## 5) DashboardUser



Gambar 4. 45 Dashboard User Final

## 6) DashboardInstruktur



Gambar 4. 46 Dashboard Instruktur Final

## 7) KelolaDataKelasAdmin

**Kelola Data Kelas**

Data Kelas

ID	NAMA KELAS	HARI	JAM	INSTRUKTUR	DESKRIPSI	TOTAL HARGA	ACTIONS
1	Kelas Virabhadrasana	Sabtu	12:00	Silvia Irtany	merupakan salah satu gerakan yoga yang berperan aktif dalam melatih otot tubuh bagian bawah. Selain itu, gerakan ini diyakini dapat meningkatkan keseimbangan dan stamina tubuh kamu.	75.000	[edit] [delete]
26	Kelas Rhejangama	Jumat	17:00	Silvia Irtany	merupakan pose klasik yang dapat menciptakan efek menenangkan, menganggun paha bawah, dan melenturkan tubuh.	1.000	[edit] [delete]
31	Kelas Setu Bandhasana	Kamis	18:00	Silvia Irtany	Rhigi pose, menguatkan otot punggung, meregangkan dada dan tulang belakang, mengurangi masalah tidur	100	[edit] [delete]
32	Kelas Paschimottasana	Kabu	15:00	Silvia Irtany	Membuka pinggul, membungkuk ke depan, meregangkan paha, selangkang, punggung, piriformis, dan pias	100.000	[edit] [delete]
23	Kelas Adho Mukha Svarasana	Selasa	09:00	Fahri Ardan Saputra	Child's pose, gerakan yang mirip sajdi dengan kaki diteluk atau diduduki	50.000	[edit] [delete]
35	Kelas Tadasana	Senin	08:00	Silvia Irtany	Pose gunung, dikenal sebagai ibu dari semua gerakan yoga	50.000	[edit] [delete]
27	012021412	Senin	21:03	2132	21312	123.123	[edit] [delete]
38	321312	Senin	12:23	123	21312	123.123	[edit] [delete]
39	21312	Sabtu	12:12	123123	112321	12.312	[edit] [delete]
41	1231	Minggu	03:23	312	32131	1.111	[edit] [delete]

Menampilkan 1 sampai 10 dari 10 data

Gambar 4. 47 Kelola Data Kelas Admin Final

## 8) KelolaDataUserAdmin

**Kelola Data User**

Data User

ID	NAME	EMAIL	ROLE	ACTIONS
2	admin	admin@gmail.com	admin	[edit] [delete]
11	fahri	fahri@gmail.com	admin	[edit] [delete]
12	talosyong	talos@gmail.com	instruktur	[edit] [delete]
24	fahri	fahriardansaputra2@gmail.com	admin	[edit] [delete]
26	gorgi	gorgi@gmail.com	instruktur	[edit] [delete]
27	fin	fin@gmail.com	instruktur	[edit] [delete]
28	princesubblegum	princesubblegum@gmail.com	user	[edit] [delete]
29	jake	jake@gmail.com	user	[edit] [delete]
30	marceline	marceline@gmail.com	user	[edit] [delete]
31	demo	demo@gmail.com	user	[edit] [delete]

Menampilkan 1 sampai 10 dari 10 data

Gambar 4. 48 Kelola Data User Admin Final

## 9) TransaksiAdmin

**Detail Transaksi Admin**

ID TRANSAKSI	NAMA USER	NAMA KELAS	TOTAL HARGA	STATUS	TANGGAL TRANSAKSI	AKSI
22	james	Kelas Paschimottarasana	100.000	Paid	19-12-2024 16:30	Lihat Invoice
23	james	Kelas Setu Sandhasana	100	Paid	19-12-2024 16:38	Lihat Invoice
24	princeosubbleegum	Kelas Virabhadrasana I	75.000	Paid	19-12-2024 17:09	Lihat Invoice
25	princeosubbleegum	Kelas Virabhadrasana I	75.000	Paid	19-12-2024 17:12	Lihat Invoice
26	princeosubbleegum	Kelas Tadana	50.000	Paid	20-12-2024 09:19	Lihat Invoice
27	james	Kelas Bhujangasana	1.000	Paid	24-12-2024 12:45	Lihat Invoice
28	admin	Kelas Tadana	50.000	Paid	25-12-2024 09:19	Lihat Invoice
29	keyra	Kelas Virabhadrasana	75.000	Paid	25-12-2024 05:22	Lihat Invoice
30	dimo	Kelas Tadana	50.000	Paid	26-12-2024 28:42	Lihat Invoice
35	james	Kelas Virabhadrasana	75.000	Paid	27-12-2024 19:00	Lihat Invoice

Menampilkan 1 sampai 10 dari 13 data

Gambar 4. 49 Transaksi Admin Final

## 10) KelolaDataKelasInstruktur

**Kelola Kelas**

Total Kelas: 10 | Kelas Selesai: 13 | Kelas Mendatang: 2 | Total Customers: 16

**Daftar Kelas**

ID	Nama Kelas	Instruktur	Hari	Jam	Harga	Aksi
1	Kelas Virabhadrasana	Silvia Itany	Sabtu	12:00	Rp 75.000	Lihat Jadwal
25	Kelas Bhujangasana	Silvia Itany	Jumat	17:00	Rp 1.000	Lihat Jadwal
31	Kelas Setu Sandhasana	Silvia Itany	Kamis	18:00	Rp 100	Lihat Jadwal
32	Kelas Paschimottarasana	Silvia Itany	Rabu	15:00	Rp 100.000	Lihat Jadwal
33	Kelas Adho Mukha Swanasana	Fahri Ardian Saputra	Selasa	09:00	Rp 50.000	Lihat Jadwal
35	Kelas Tadana	Silvia Itany	Senin	08:00	Rp 50.000	Lihat Jadwal
27	413d74c12	3123	Senin	21:03	Rp 123.123	Lihat Jadwal
38	321912	123	Senin	12:23	Rp 123.123	Lihat Jadwal
39	21212	123123	Sabtu	12:12	Rp 12.312	Lihat Jadwal
41	1221	312	Minggu	00:23	Rp 1.111	Lihat Jadwal

Menampilkan 1 sampai 10 dari 10 data

Gambar 4. 50. Kelola Data Kelas Instruktur Final

## 11) JadwalInstruktur

**Lihat Jadwal**

Daftar Jadwal Kelas

Tampilkan 10 data

ID	Nama Kelas	Instruktur	Hari	Jam	Jumlah Peserta	Aksi
1	Kelas VirahNadrasana	Silvia Irary	Sabtu	12:00	0	Selasa
25	Kelas Bhujangasana	Silvia Irary	Jumat	17:00	1	Selasa
31	Kelas Setu Sandhurasana	Silvia Irary	Kamis	18:00	0	Selasa
32	Kelas Paschimottasana	Silvia Irary	Rabu	15:00	1	Selasa
33	Kelas Adho Mukha Swarasana	Fatri Andri Saputra	Selasa	09:00	0	Selasa
35	Kelas Tadasana	Silvia Irary	Senin	08:00	1	Selasa
37	012021012	3123	Senin	21:03	0	Selasa
38	321112	123	Senin	12:23	0	Selasa
39	21012	123123	Sabtu	12:12	0	Selasa
41	1231	312	Minggu	09:23	1	Selasa

Menampilkan 1 sampai 10 dari 10 data

Sebelumnya 1 | Berikutnya

Gambar 4. 51 Jadwal Instruktur Final

## 12) JadwalUser

**Jadwal Kelas**

Serlin  
Available Classes  
3 kelas  
LIHAT KELAS

**Jadwal Kelas pada Senin**

Tampilkan 10 data

NAMA KELAS	JAM	DESKRIPSI	INSRUKTUR	TOTAL HARGA	Aksi
321012	12:23	21312	123	Rp 123.000	Save to Book
012021012	21:03	21312	3123	Rp 123.000	Save to Book
Kelas Tadasana	09:00	Pose gunung, diarah sebagai ibu dari semua gerakan yoga	Silvia Irary	Rp 50.000,00	Save to Book

Menampilkan 1 sampai 3 dari 3 data

Sebelumnya 1 | Berikutnya

**Form Booking**

0090  
012012  
0090  
0090  
123123 00  
Book

Gambar 4. 52 Jadwal User Final

### 13) Booking

BOOKING ID	NAMA KELAS	HARI	TOTAL HARGA	STATUS	AKSI
70	Kelas Tadasana	Senin	Rp 50.000	Unpaid	Bayar
71	1231	Minggu	Rp 1.111	Paid	Sudah Dibayar
72	Kelas Paschimottanasana	Rabu	Rp 100.000	Paid	Sudah Dibayar
73	Kelas Bhujangasana	Jumat	Rp 1.000	Unpaid	Bayar

Gambar 4. 53 Booking Final

### 14) TransaksiUser

TRANSAKSI ID	NAMA	NAMA KELAS	HARI	JAM	SELESAI PADA	AKSI
30	Jawa	Kelas Tadasana	Senin	08.00	26-12-2024 23:42	Lihat Invoice
38	Jawa	1231	Minggu	09:23	29-12-2024 09:39	Lihat Invoice
39	Jawa	Kelas Paschimottanasana	Rabu	15.00	29-12-2024 09:45	Lihat Invoice

Gambar 4. 54 Transaksi User Final

### 15) FormTambahDataKelasAdmin

**Data Kelas**

**Nama Kelas:**  
Masukkan nama kelas

**Hari:**  
Senin

**Jam:**  
Masukkan jam kelas

**Instruktur:**  
Masukkan Nama Instruktur

**Deskripsi:**  
Masukkan deskripsi kelas

**Total Harga:**  
Masukkan total harga

Simpan

Gambar 4. 55 Form Tambah Kelas Admin Final

### 16) FormTambahDataKelasInstruktur

**Nama Kelas:**

**Nama Instruktur:**

**Hari:**  
Senin

**Jam:**

**Harga:**

**Deskripsi:**

Simpan

Gambar 4. 56Form Tambah Data Kelas Instruktur Final

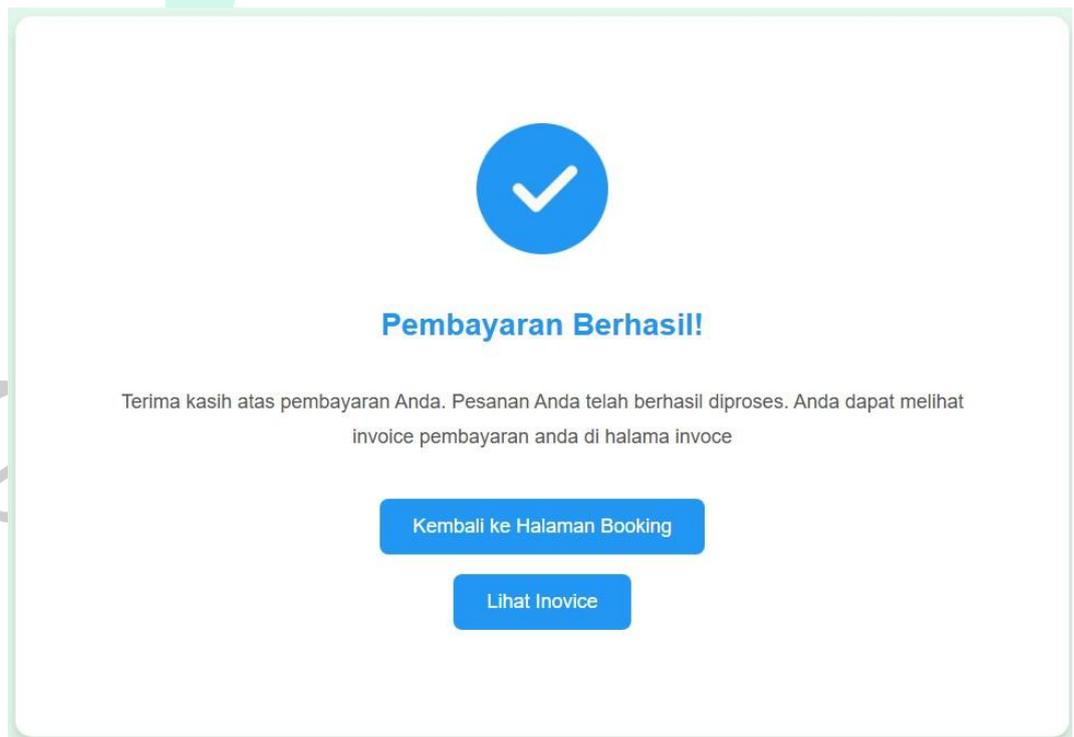
## 17) FormTambahDataUserAdmin

Data User

<b>Nama:</b> <input type="text" value="Masukkan nama"/>	<b>Email:</b> <input type="text" value="Masukkan email"/>	<b>Password:</b> <input type="text" value="Masukkan password"/>
<b>Konfirmasi Password:</b> <input type="text" value="Konfirmasi password"/>	<b>Role:</b> <input type="text" value="admin"/>	<input type="button" value="Simpan"/>

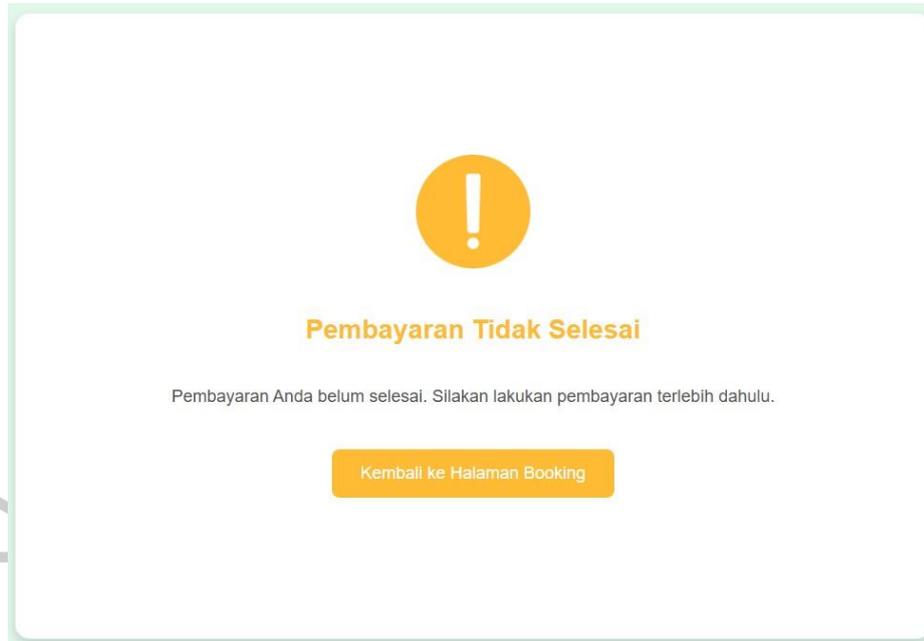
Gambar 4. 57 Form Tambah Data User Admin Final

## 18) Notifikasi Success



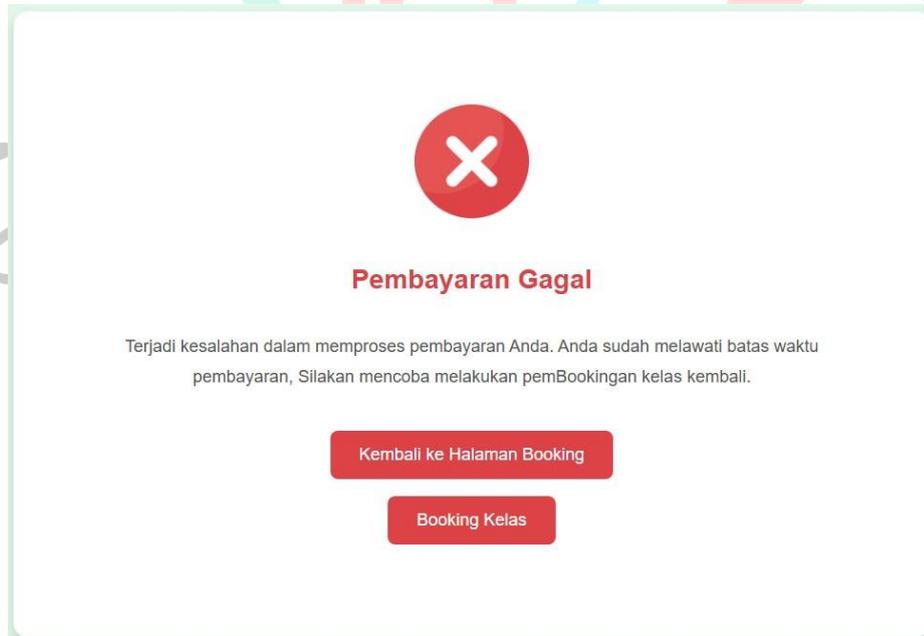
Gambar 4. 58 Notifikasi Berhasil Final

## 19) Notifikasi Caution



Gambar 4. 59 Notifikasi Belum Selesai Final

## 20) Notifikasi Failed



Gambar 4. 60 Notifikasi Gagal Final

## 21) Invoice

**Invoice Transaksi**  
Terima kasih telah melakukan transaksi dengan kami!

**Detail Booking**

Booking ID: 55  
Nama: james  
Nama Kelas: Kelas Paschimottanasana  
Hari: Rabu  
Jam: 15.00  
Total Harga: Rp 100.000  
Status: Paid

**Detail Transaksi Midtrans**

Order ID: ORDER-55  
Transaction ID: f14ec114-37e1-4ea0-bd20-f37c05cfc852  
Payment Type: Bank\_transfer  
Gross Amount: Rp 100.000  
Status Transaksi: Settlement  
Dibayar Pada: 2024-12-19 23:30:03

[Kembali](#)

Gambar 4. 61 Invoice Final

## 22) KelolaAkunPribadiAdmin

**Kelola Akun**

Nama: admin Email: admin@gmail.com

Password Baru

Masukkan password baru

Konfirmasi Password

Konfirmasi password

[Simpan](#) [Batal](#)

Gambar 4. 62 Kelola Akun Pribadi Admin Final

### 23) KelolaAkunPribadiUser

**Kelola Akun**

Nama: demo | Email: demo@gmail.com

Password Baru

Masukkan password baru

Konfirmasi Password

Konfirmasi password

[Simpan](#) [Batal](#)

Gambar 4. 63 Kelola Akun Pribadi User Final

### 24) KelolaAkunPribadiInstruktur

**Kelola Akun**

Nama: talasyang | Email: tala@gmail.com

Password Baru

Masukkan password baru

Konfirmasi Password

Konfirmasi password

[Simpan](#) [Batal](#)

Gambar 4. 64 Kelola Akun Pribadi Instruktur Final

#### 4.6 Tahap konstruksi (Construction)

Pada tahap ini, pengembangan aplikasi dilakukan dengan mengacu pada hasil analisis kebutuhan dan rancangan dari tahap sebelumnya. Website dibangun secara manual menggunakan database MySQL. Setiap fitur dirancang untuk memenuhi kebutuhan pengguna, seperti pendaftaran, penjadwalan, pembayaran serta pengelolaan data. Seluruh fungsi utama dikembangkan secara langsung dengan melibatkan tiga aktor yang berbeda.

#### 4.7 Perancangan Implementasi

##### 4.7.1 Jadwal Implementasi

Tabel 4. 28 Tabel Jadwal Implementasi

No	Nama Kegiatan	Bulan															
		Bulan 1				Bulan 2				Bulan 3				Bulan 4			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Identifikasi Kebutuhan dan Analisis Awal	■															
2	Desain dan Prototyping		■	■													
3	Pembangunan dan Pengembangan			■	■	■	■										
4	Uji Coba dan Validasi								■								
5	Penyempurnaan dan Iterasi								■	■	■	■	■				
6	Penulisan Laporan													■	■	■	■

#### 4.7.2 Tahap Implementasi

Tabel 4. 29 Tabel Test Case

Test Case ID	Deskripsi	Langkah Uji	Keluaran	Hasil Aktual
TC001	Pendaftaran Akun	Pengguna mengisi formulir pendaftaran dengan data valid.	Akun berhasil didaftarkan, dan muncul notifikasi "Pendaftaran berhasil".	Sesuai
TC002	Pendaftaran Gagal	Pengguna mengisi formulir pendaftaran dengan data tidak valid (misal. email kosong).	Muncul notifikasi "Pendaftaran gagal, silakan periksa data Anda".	Sesuai
TC003	Login Berhasil	Pengguna memasukkan username dan password yang valid.	Pengguna diarahkan ke dashboard sesuai role mereka.	Sesuai
TC004	Login Gagal	Pengguna memasukkan username atau password yang salah.	Muncul notifikasi "Login gagal, username atau password salah".	Sesuai
TC005	Logout Berhasil	Pengguna melakukan logout dari aplikasi.	Pengguna diarahkan kembali ke halaman login.	Sesuai
TC006	Logout Gagal	Sistem mengalami error saat logout.	pengguna tetap berada di halaman sebelumnya.	Sesuai
TC007	Akses Halaman Customer	Customer login ke aplikasi.	Customer hanya bisa mengakses halaman customer.	Sesuai

<b>TC008</b>	Akses Halaman Admin	Admin login ke aplikasi.	Admin hanya bisa mengakses halaman admin.	Sesuai
<b>TC009</b>	Transaksi Berhasil Ditampilkan	Customer dan admin melihat transaksi pada dashboard mereka.	Daftar transaksi beserta statusnya berhasil ditampilkan.	Sesuai
<b>TC010</b>	Transaksi Gagal Ditampilkan	Customer dan admin membuka halaman transaksi saat server down.	Muncul notifikasi "Gagal memuat data transaksi".	Sesuai
<b>TC011</b>	Pengelolaan Data oleh Admin	Admin menambah atau memperbarui data.	Data berhasil ditambahkan atau diperbarui, muncul notifikasi sukses.	Sesuai
<b>TC012</b>	Pengelolaan Data Gagal	Admin menambah data dengan informasi yang tidak lengkap.	Muncul notifikasi "silahkan isi field yang kosong".	Sesuai
<b>TC013</b>	Sort dan Search Database	Pengguna melakukan pencarian atau penyortiran data table.	Data ditampilkan sesuai kriteria pencarian atau penyortiran.	Sesuai
<b>TC014</b>	Sort dan Search Database Gagal	Pengguna mencoba mencari data dengan filter yang tidak valid.	Muncul notifikasi "Tidak ada data yang sesuai".	Sesuai
<b>TC015</b>	Tracking Kuota Kelas	Instruktur melihat kuota kelas pada kelola kelas.	Kuota kelas berhasil ditampilkan sesuai	Sesuai

			dengan data yang tersedia.	
<b>TC019</b>	Booking Kelas	Customer melakukan pembayaran untuk sebuah kelas yang di booking.	pembayaran berhasil, dan muncul notifikasi.	Sesuai
<b>TC016</b>	Booking Kelas Belum selesai	Customer meninggalkan pembayaran tanpa menyelesaikan pembayaran	muncul notifikasi pembayaran belum selesai	Sesuai
<b>TC017</b>	Booking Kelas Gagal	Customer gagal melakukan pembayaran di waktu yang sudah ditentukan.	muncul notifikasi pembayaran gagal	Sesuai
<b>TC018</b>	Lihat Invoice	Customer dan admin melihat detail invoice untuk transaksi tertentu.	Invoice berhasil ditampilkan dengan detail lengkap.	Sesuai
<b>TC019</b>	Pengelolaan Data oleh Instruktur	Instruktur menambah atau memperbarui data.	Data berhasil ditambahkan atau diperbarui, muncul notifikasi sukses.	Sesuai
<b>TC020</b>	Pengelolaan Data Gagal	Instruktur menambah data dengan informasi yang tidak lengkap.	Muncul notifikasi "silahkan isi field yang kosong".	Sesuai

<b>TC021</b>	Lihat Jadwal Customer	customer melihat jadwal kelas yang tersedia dan melakukan booking	jadwal kelas berhasil ditampilkan dan kelas berhasil di booking	Sesuai
<b>TC022</b>	Lihat Jadwal Instruktur	instruktur melihat jadwal kelas yang tersedia	jadwal kelas berhasil ditampilkan	Sesuai
<b>TC023</b>	Gagal lihat jadwal	jadwal kelas sudah tidak lagi tersedia	muncul "kelas tidak tersedia pada hari ini"	Sesuai
<b>TC024</b>	Kelola Akun Pribadi	Pengguna memperbarui informasi akun pribadi	informasi berhasil diperbarui dan sesi berakhir	Sesuai
<b>TC025</b>	Pengelolaan Akun Gagal	Pengguna menambah data dengan informasi yang belum lengkap	Muncul notifikasi "silahkan isi field yang kosong".	Sesuai