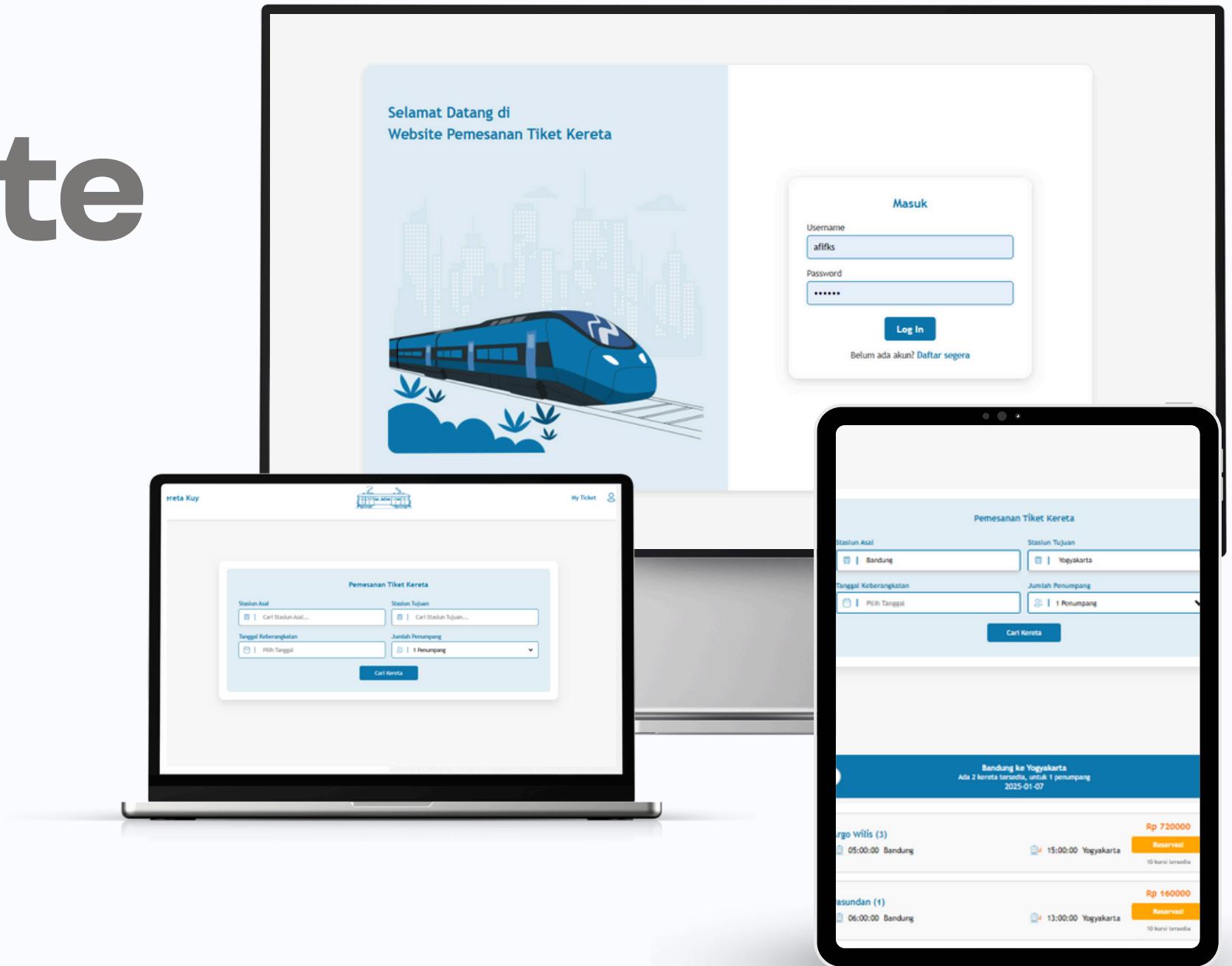


PEMROGRAMAN BERORIENTASI OBJEK
Kereta Kuy (Kelompok 03) | IF-46-07

KeretaKuy Website Pemesanan Tiket Kereta

Disusun Oleh

- Afif Kurniawan Supriadi - 1301223161
- Aydin Shidqi Athalla K - 1301223313379
- Christoper Daeng Kilantan - 1301223402
- Daffa Ananta Rachman -- 1301223140
- M Rafli Adhan S - 1301223042
- Muhammad Ribhy Rizwanul Islam - 1301223449



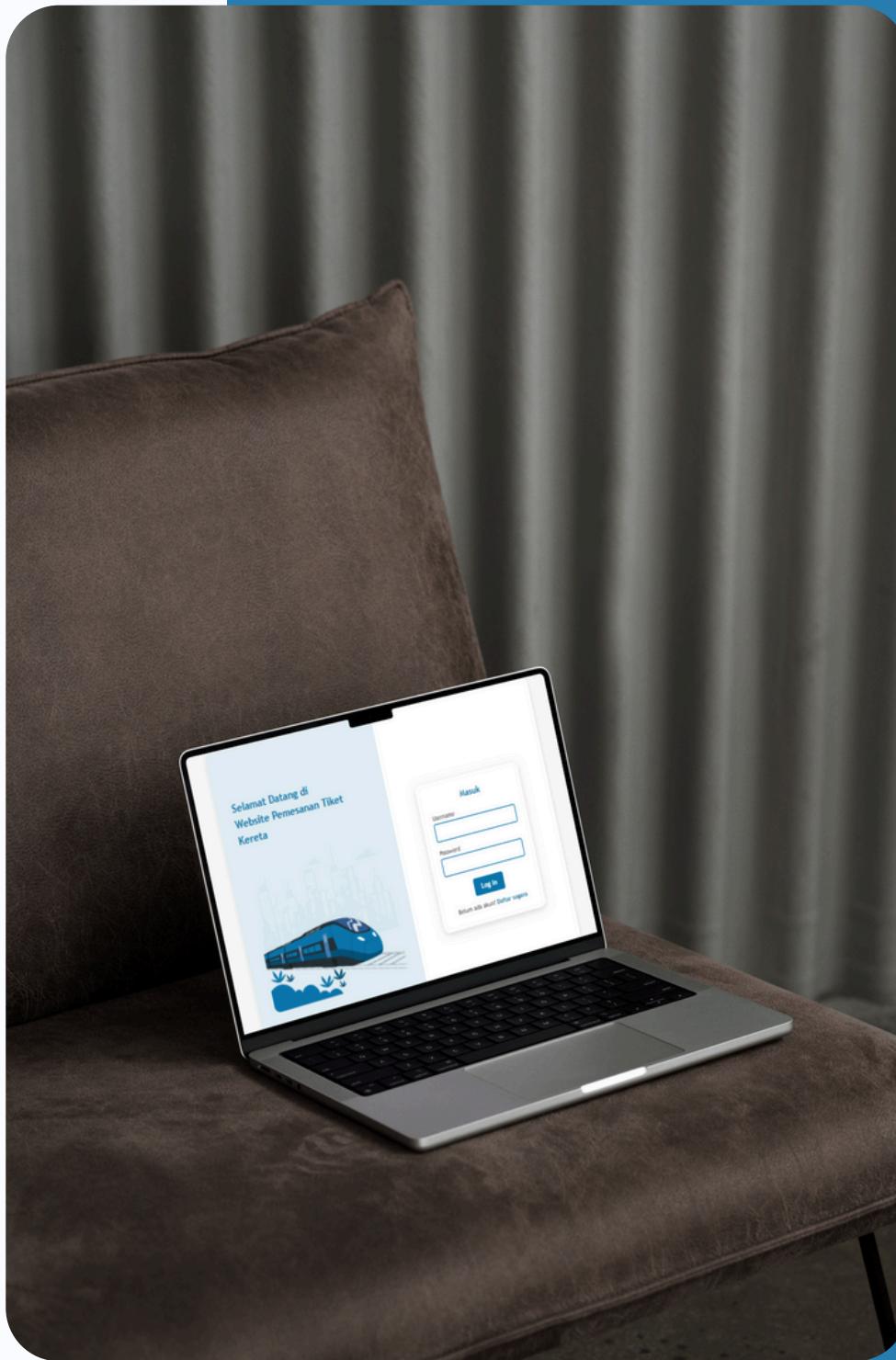
Deskripsi Perangkat Lunak

★ Website Pemesanan Tiket Kereta

KeretaKuy merupakan **aplikasi berbasis web** yang dirancang untuk memberikan kemudahan kepada pengguna dalam melakukan **pemesanan tiket kereta api**.

Fitur utama website ini adalah sebagai berikut:

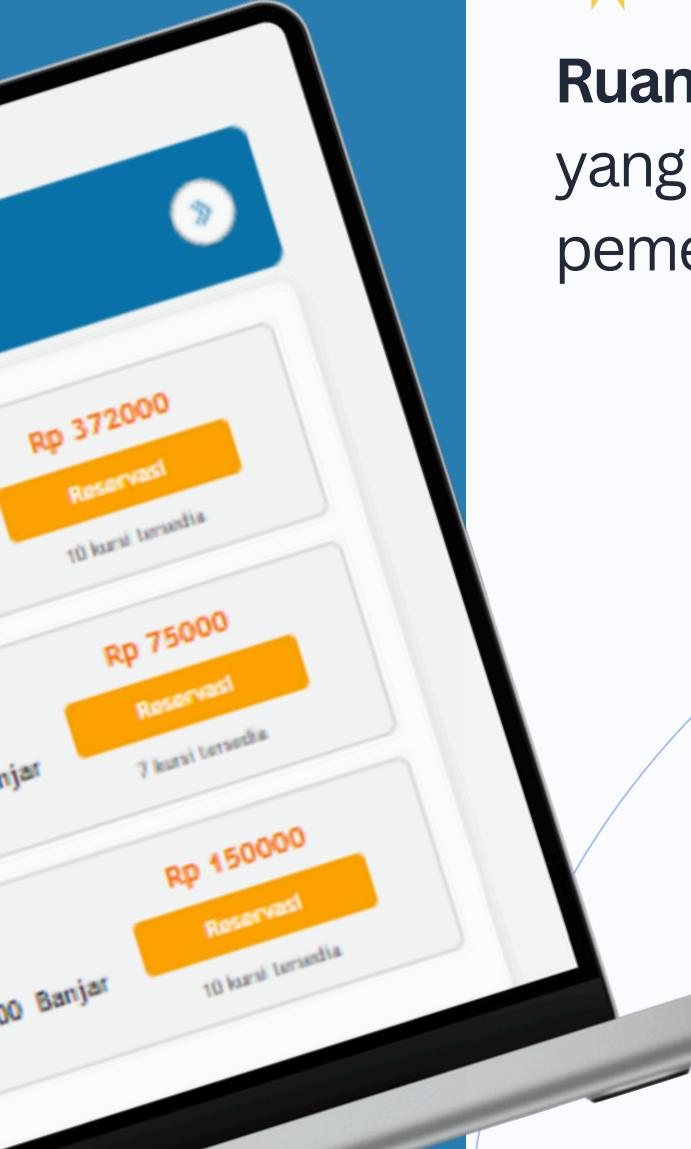
- 1. Pencarian Jadwal Kereta:** Mencari jadwal kereta berdasarkan stasiun asal, stasiun tujuan, tanggal keberangkatan, dan jumlah penumpang.
- 2. Pemesanan Tiket:** Memesan tiket kereta berdasarkan pencarian jadwal kereta yang telah dilakukan. Pengguna dapat melakukan pemesanan sampai dengan 3 tiket untuk 3 penumpang yang berbeda dalam satu kali proses pemesanan (booking).
- 3. Manajemen Tiket:** Melihat daftar tiket yang telah dipesan, termasuk informasi lengkap seperti nama kereta, jadwal keberangkatan, stasiun asal, dan stasiun tujuan.



Ruang Lingkup Aplikasi

Website Pemesanan Tiket Kereta

Ruang lingkup aplikasi web pemesanan tiket kereta ini mencakup berbagai **fungsi** atau **modul** yang dirancang untuk memberikan kemudahan kepada pengguna dalam melakukan pemesanan tiket kereta. Berikut adalah cakupan utamanya:



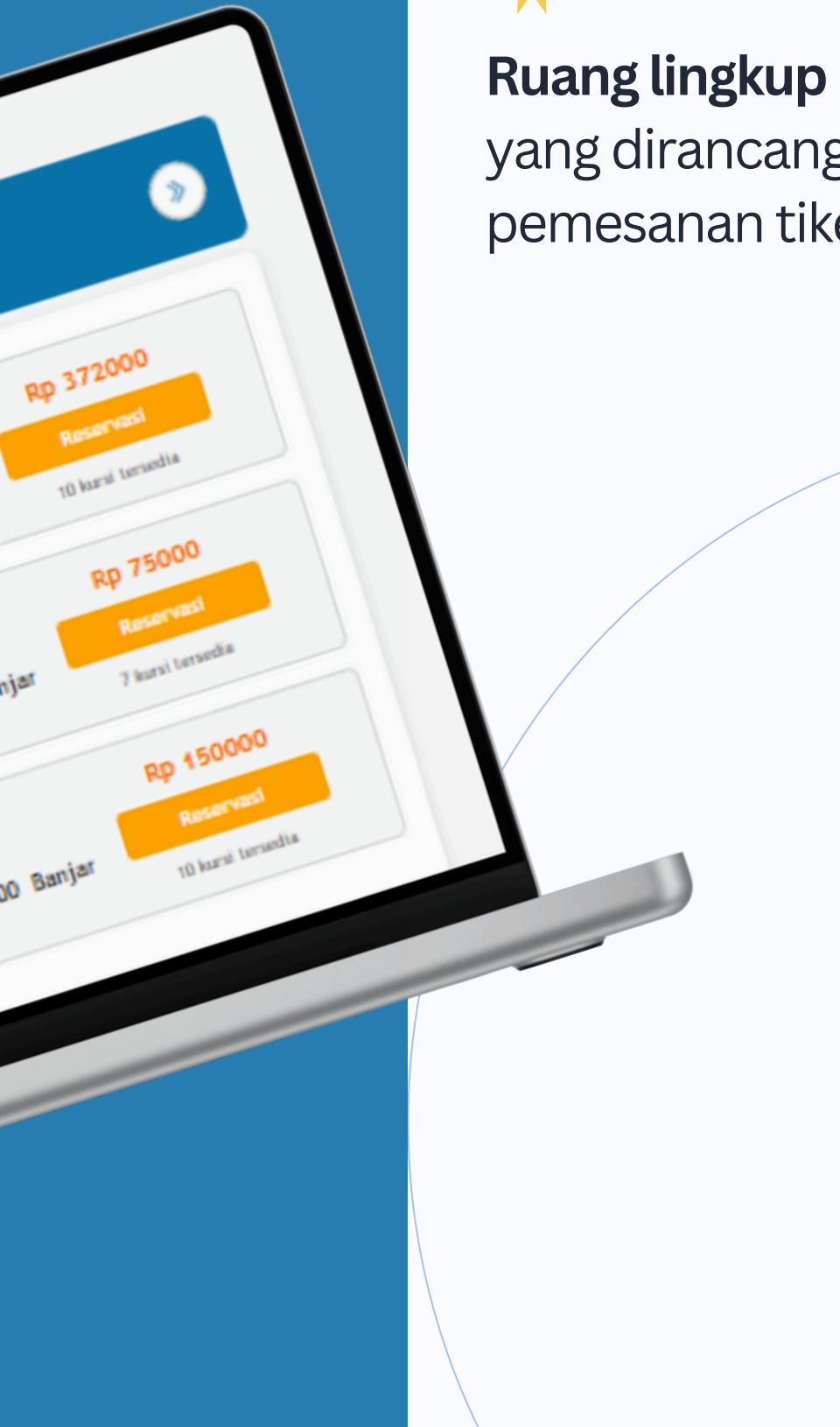
Modul/Fungsi	Nama Modul	Deskripsi
FR-01	Fungsi Registrasi	Pengguna dapat memasukkan informasi pribadi untuk membuat akun seperti, nama lengkap, email, tanggal lahir, jenis kelamin, username, dan password
FR-02	Fungsi Login	Sistem harus memverifikasi username dan password pengguna sebelum memberikan akses untuk menggunakan website
FR-03	Fungsi Mencari Tiket	Pengguna dapat mencari tiket dengan memasukkan informasi seperti stasiun asal, stasiun tujuan, tanggal keberangkatan, dan jumlah penumpang.

Ruang Lingkup Aplikasi

Website Pemesanan Tiket Kereta

Ruang lingkup aplikasi web pemesanan tiket kereta ini mencakup berbagai **fungsi** atau **modul** yang dirancang untuk memberikan kemudahan kepada pengguna dalam melakukan pemesanan tiket kereta. Berikut adalah cakupan utamanya:

FR-04	Fungsi Memesan Tiket	<p>Pengguna dapat memesan tiket kereta berdasarkan hasil pencarian. Informasi detail terkait jadwal perjalanan kereta akan ditampilkan seperti: nama kereta, jam keberangkatan, jam tiba, stasiun asal, stasiun tujuan, harga tiket, dan kapasitas kursi yang masih tersedia.</p> <p>Satu pengguna dapat melakukan pemesanan tiket sampai dengan 3 penumpang dalam satu kali proses pemesanan</p>
FR-05	Fungsi Melihat Tiket	Pengguna dapat melihat daftar tiket yang telah dipesan, informasi seperti data penumpang dan detail tiket akan ditampilkan. Selain itu, pengguna juga dapat melakukan pembatalan tiket atau menghapus tiket yang telah di pesan
FR-96	Fungsi Edit User	Pengguna dapat melakukan perubahan informasi akun atau data diri mereka pada edit profile



Batasan Aplikasi



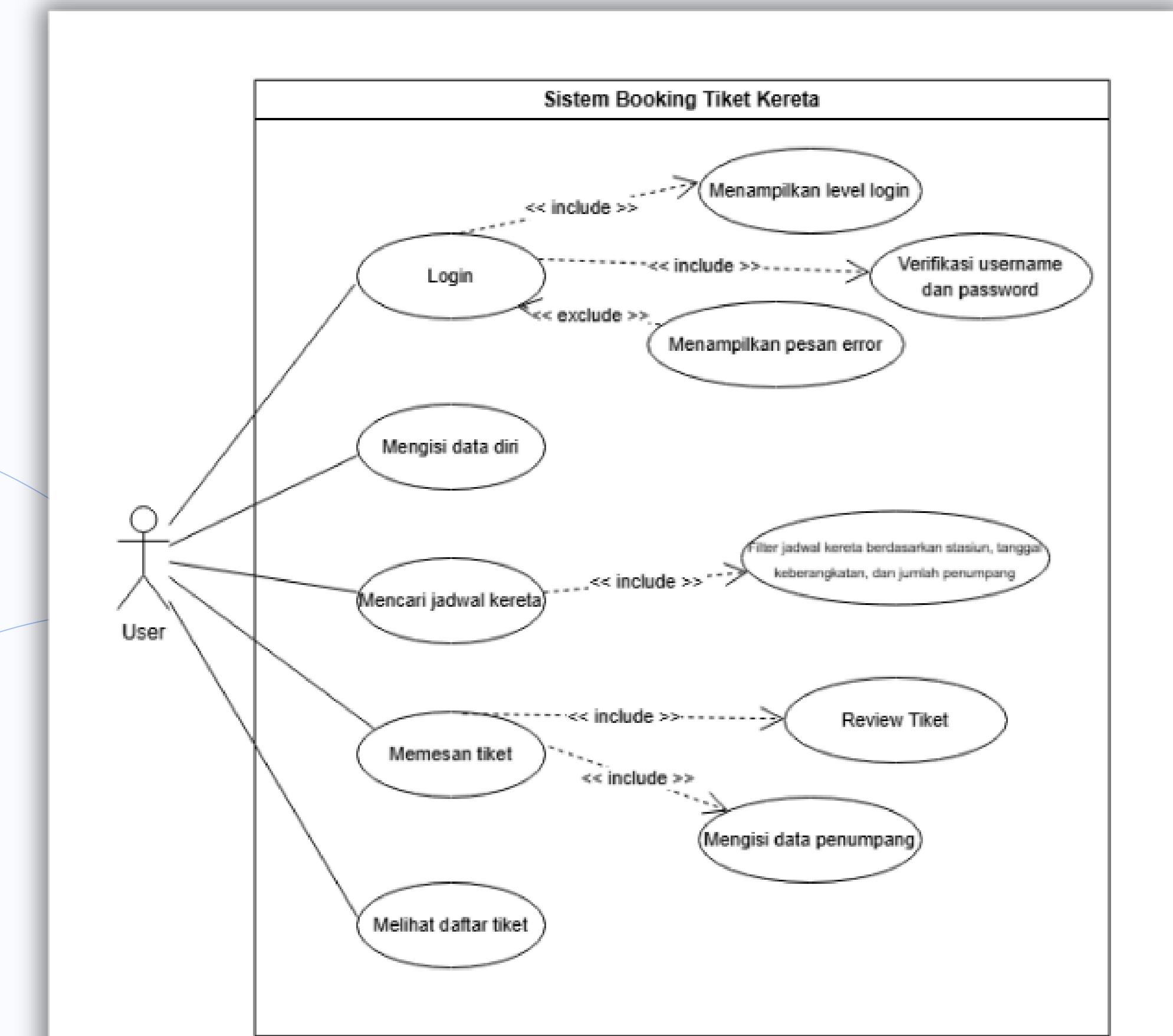
Website Pemesanan Tiket Kereta

Aplikasi pemesanan tiket kereta berbasis web yang dikembangkan memiliki beberapa **batasan**, seperti:

- 1. Tidak ada sistem pembayaran:** Website **tidak dilengkapi** dengan fitur pembayaran online. Proses pemesanan tiket hanya sebatas **pencatatan informasi** pemesanan hingga tiket terdaftar dalam sistem.
- 2. Data kereta dan stasiun terbatas pada database:** Data mengenai kereta, jadwal, stasiun, dan kapasitas kursi **diambil** dari **database internal** yang harus diinput secara manual. Jadi, Sistem tidak mendukung integrasi otomatis dengan API jadwal kereta eksternal.

Alur Website

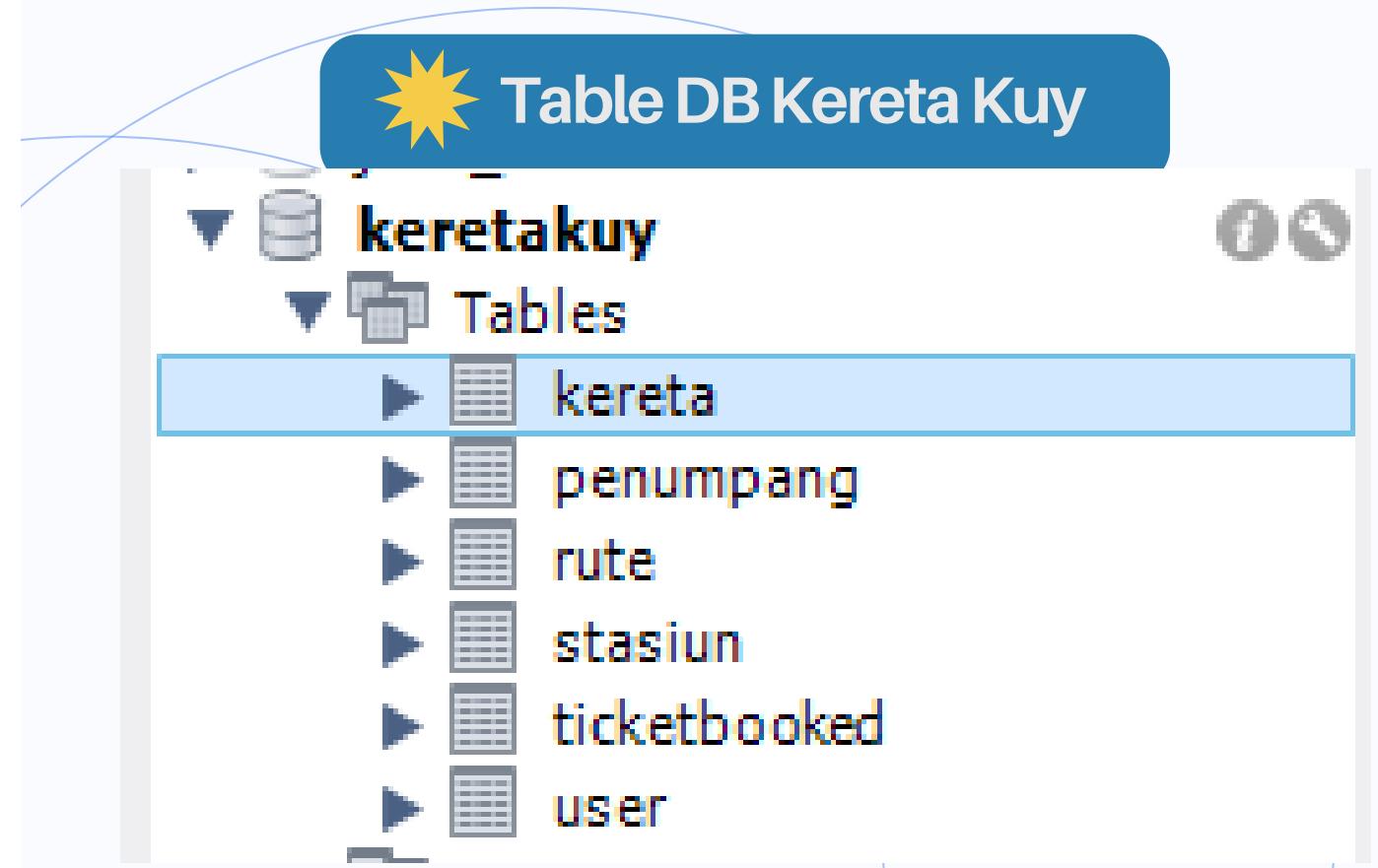
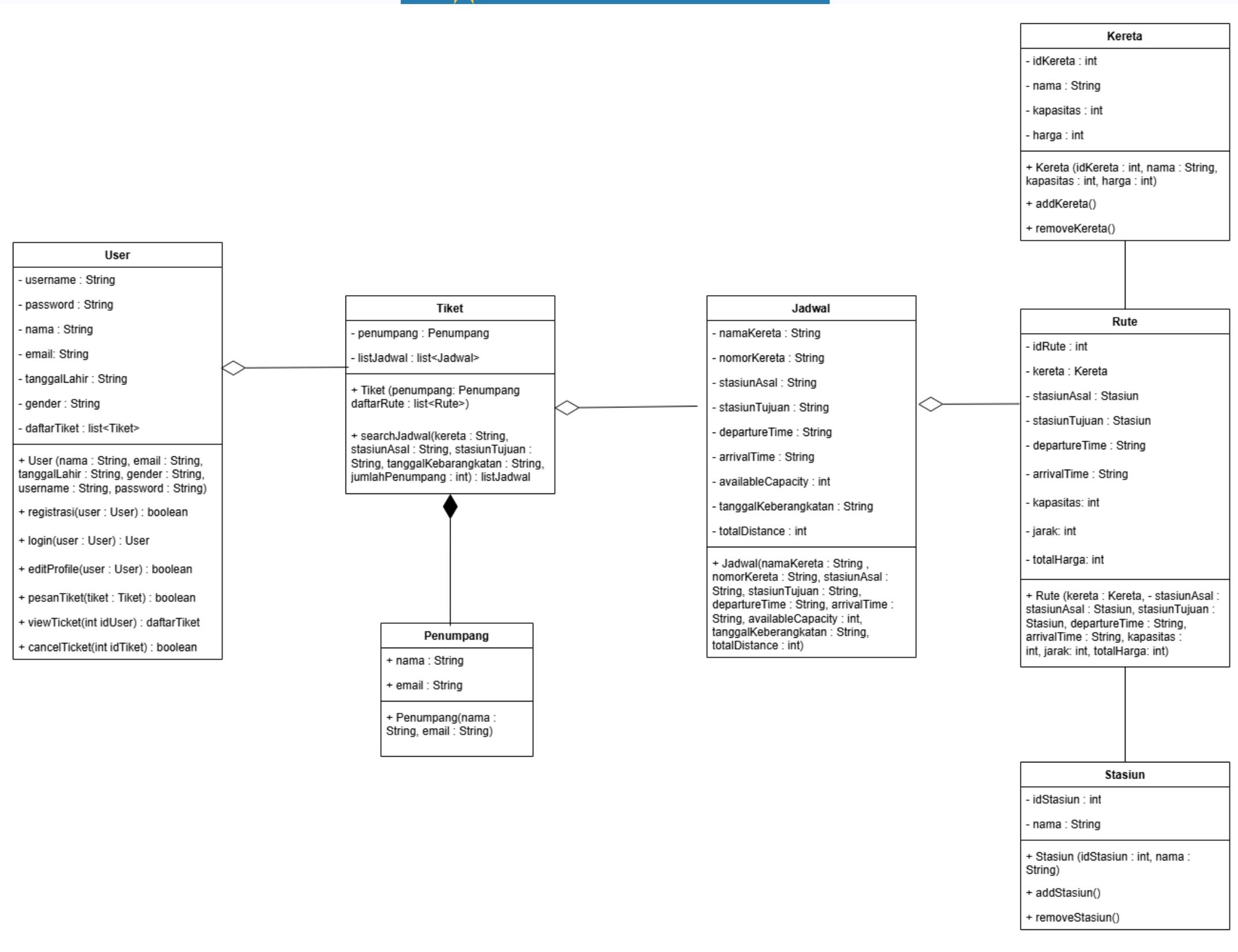
★ UseCase Diagram



Implementasi Basis Data

Implementasi Basis Data

Class Diagram



Implementasi Basis Data

Struktur Table User

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	idUser	int(3)		UNSIGNED ZEROFILL	Tidak	Tidak ada		AUTO_INCREMENT
2	nama	varchar(50)	utf8mb4_general_ci		Ya	NULL		
3	email	varchar(30)	utf8mb4_general_ci		Ya	NULL		
4	gender	enum('Laki-laki', 'Perempuan')	utf8mb4_general_ci		Ya	NULL		
5	tanggalLahir	date			Ya	NULL		
6	username	varchar(20)	utf8mb4_general_ci		Ya	NULL		
7	passwrd	varchar(20)	utf8mb4_general_ci		Ya	NULL		

NOTE:

Table User akan digunakan untuk mengimplementasikan **Modul Login** dan **Modul Register** pada website

Contoh Data

	Ubah	Salin	Hapus	idUser	nama	email	gender	tanggalLahir	username	passwrd

	Ubah	Salin	Hapus	idUser	nama	email	gender	tanggalLahir	username	passwrd
				001	Afif Kurniawan Supriyadi	afifiksupriyadi@gmail.com	Laki-laki	2024-12-28	afifks	afifks
				002	Muhammad Fathir Arraihan	fathir@gmail.com	Laki-laki	2024-12-01	fathir	fathir
				003	Christoper Daeng Kilantan	christoper.daeng@gmail.com	Laki-laki	2024-12-15	christoper	christoper
				004	Nazhmi Ahmad Fauzan	nazhmi.fauzan@gmail.com	Laki-laki	2024-12-31	nazhmi	nazhmi
				005	Muhammad Ribhy Rizwanul Islam	ribhy.rizwanul@gmail.com	Laki-laki	2025-01-05	ribhy	ribhy
				006	Sutan Rifky Tedjasukmana	sutanrifky.tedjasukmana@gmail.com	Laki-laki	2024-12-29	sutanrifky	sutanrifky

Implementasi Basis Data

Struktur Table Kereta

Server: 127.0.0.1 » Database: keretakuy » Tabel: kereta

Jelajahi Struktur SQL Cari Tambahan Eksport Impor Hak Akses

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	idKereta	int(2)	UNSIGNED ZEROFILL	Tidak	Tidak ada	AUTO_INCREMENT		
2	nama	varchar(100)	utf8mb4_general_ci	Tidak	Tidak ada			
3	kapasitas	int(11)		Tidak	Tidak ada			
4	hargaKereta	int(11)		Ya	NULL			

Contoh Data

Extra options

	Ubah	Salin	Hapus	idKereta	nama	kapasitas	hargaKereta
				01	Pasundan	10	500
				02	Pangandaran	10	1000
				03	Argo Wilis	10	1200
				04	Brawijaya	10	1300

Struktur Table Stasiun

Server: 127.0.0.1 » Database: keretakuy » Tabel: stasiun

Jelajahi Struktur SQL Cari Tambahan Eksport Impor Hak Akses

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	idStasiun	int(3)	UNSIGNED ZEROFILL	Tidak	Tidak ada	AUTO_INCREMENT		
2	nama	varchar(100)	utf8mb4_general_ci	Tidak	Tidak ada			

Contoh Data

Extra options

	Ubah	Salin	Hapus	idStasiun	nama
				001	Bandung
				002	Tasikmalaya
				003	Ciamis
				004	Banjar
				005	Kroya
				006	Yogyakarta
				007	Sragen
				008	Madiun
				009	Jombang
				010	Surabaya
				011	Jakarta
				012	Bekasi
				013	Cimahi
				014	Cilacap
				015	Cipeundeuy
				016	Kutoarjo
				017	Solo Balapan
				018	Mojokerto
				019	Surabaya Gubeng
				020	Malang
				021	Malang Kotalama
				022	Kepanjen
				023	Wlingi
				024	Blitar
				025	Tulungagung

Implementasi Basis Data

Struktur Table Rute

Server: 127.0.0.1 » Database: keretaku » Tabel: rute

Jelajahi Struktur SQL Cari Tambahan Eksport Impor Hak Aks

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	idRute	int(11)			Tidak	Tidak ada		AUTO_INCREMENT
2	idKereta	int(2)			UNSIGNED ZEROFILL	Ya	NULL	
3	idStasiunAsal	int(3)			UNSIGNED ZEROFILL	Ya	NULL	
4	idStasiunTujuan	int(3)			UNSIGNED ZEROFILL	Ya	NULL	
5	departureTime	time			Tidak	Tidak ada		
6	arrivalTime	time			Tidak	Tidak ada		
7	distance	int(11)			Tidak	Tidak ada		
8	capacity	int(11)			Tidak	Tidak ada		
9	date	date			Tidak	Tidak ada		

Contoh Query Routing (Insert) Kereta Pasundan, Tanggal 06 Januari 2025

```
103 -- Data untuk kereta Pasundan pada tanggal 6, 7, 8, 9, dan 10 Januari 2025
104 • INSERT INTO Rute (idKereta, idStasiunAsal, idStasiunTujuan, departureTime, arrivalTime, distance, capacity, date)
105 VALUES
106 -- 6 Januari 2025
107 (01, 001, 002, '06:00:00', '07:30:00', 70, 10, '2025-01-06'),-- Bandung to Tasikmalaya
108 (01, 002, 003, '07:45:00', '08:15:00', 40, 10, '2025-01-06'),-- Tasikmalaya to Ciamis
109 (01, 003, 004, '08:30:00', '09:00:00', 40, 10, '2025-01-06'),-- Ciamis to Banjar
110 (01, 004, 005, '09:15:00', '10:45:00', 50, 10, '2025-01-06'),-- Banjar to Kroya
111 (01, 005, 006, '11:00:00', '13:00:00', 120, 10, '2025-01-06'),-- Kroya to Yogyakarta
112 (01, 006, 007, '13:15:00', '14:45:00', 80, 10, '2025-01-06'),-- Yogyakarta to Sragen
113 (01, 007, 008, '15:00:00', '16:30:00', 100, 10, '2025-01-06'),-- Sragen to Madiun
114 (01, 008, 009, '16:45:00', '17:45:00', 110, 10, '2025-01-06'),-- Madiun to Jombang
115 (01, 009, 010, '18:00:00', '19:30:00', 90, 10, '2025-01-06'),-- Jombang to Surabaya
116
```

Contoh Data Routing Kereta Pasundan, Tanggal 06 Januari 2025

Tampilkan semua Jumlah baris: Semua Saring baris: Cari di tabel ini Sort by key: Tidak ada

	Ubah	Salin	Hapus	idRute	idKereta	idStasiunAsal	idStasiunTujuan	departureTime	arrivalTime	distance	capacity	date
				1	01	001	002	06:00:00	07:30:00	70	7	2025-01-06
				2	01	002	003	07:45:00	08:15:00	40	7	2025-01-06
				3	01	003	004	08:30:00	09:00:00	40	7	2025-01-06
				4	01	004	005	09:15:00	10:45:00	50	10	2025-01-06
				5	01	005	006	11:00:00	13:00:00	120	10	2025-01-06
				6	01	006	007	13:15:00	14:45:00	80	10	2025-01-06
				7	01	007	008	15:00:00	16:30:00	100	10	2025-01-06
				8	01	008	009	16:45:00	17:45:00	110	10	2025-01-06
				9	01	009	010	18:00:00	19:30:00	90	10	2025-01-06

NOTE:

Table Rute akan digunakan untuk mengimplementasikan **Modul Searching** pada website

Implementasi Basis Data

Contoh Query Searching:

Mencari jadwal perjalanan kereta yang berangkat dari stasiun "Bandung" ke "Tasikmalaya", pada tanggal 10 Januari 2025, untuk 3 penumpang

```
30    -- Query Searching Kereta DENGAN tanggal
31 •  SELECT
32      k.nama AS namaKereta,
33      k.idKereta AS nomorKereta,
34      s1.nama AS stasiunAsal,
35      s2.nama AS stasiunTujuan,
36      MIN(r1.departureTime) AS departureTime,
37      MAX(r2.arrivalTime) AS arrivalTime,
38      MIN(r1.capacity) AS availableCapacity,
39      SUM(rJoin.distance) AS totalDistance,
40      (SUM(rJoin.distance) * k.hargaKereta) AS harga,
41      r1.date AS tanggalKeberangkatan
42
43  FROM Rute r1
44  JOIN Rute r2 ON r1.idKereta = r2.idKereta
45  JOIN Kereta k ON r1.idKereta = k.idKereta
46  JOIN Stasiun s1 ON r1.idStasiunAsal = s1.idStasiun
47  JOIN Stasiun s2 ON r2.idStasiunTujuan = s2.idStasiun
48  JOIN Rute rJoin ON r1.idKereta = rJoin.idKereta
49    AND rJoin.idRute BETWEEN r1.idRute AND r2.idRute
50  WHERE s1.nama = 'Bandung' -- Stasiun Asal (Input User)
51    AND s2.nama = 'Tasikmalaya' -- Stasiun Tujuan (Input User)
52    AND r1.date = '2025-01-10' -- Tanggal Keberangkatan (Input User)
53    AND r2.date = '2025-01-10' -- Tanggal Keberangkatan sama
54  GROUP BY k.nama, k.idKereta, s1.nama, s2.nama, r1.date, k.hargaKereta
55  HAVING MIN(r1.capacity) >= 3 -- Kapasitas minimal (Input User)
56  ORDER BY departureTime;
```

Hasil Searching

namaKereta	nomorKereta	stasiunAsal	stasiunTujuan	departureTime	arrivalTime	availableCapacity	totalDistance	harga	tanggalKeberangkatan
Argo Wilis	03	Bandung	Tasikmalaya	05:00:00	08:15:00	10	230	276000	2025-01-10
Pasundan	01	Bandung	Tasikmalaya	06:00:00	07:30:00	10	70	35000	2025-01-10
Pangandaran	02	Bandung	Tasikmalaya	11:15:00	13:00:00	10	70	70000	2025-01-10

Hasil Searching di Website

Bandung ke Tasikmalaya
Ada 3 kereta tersedia, untuk 3 penumpang
2025-01-10

Argo Wilis (3) 05:00:00 Bandung 08:15:00 Tasikmalaya Rp 276000 Reservasi 10 kursi tersedia
Pasundan (1) 06:00:00 Bandung 07:30:00 Tasikmalaya Rp 35000 Reservasi 10 kursi tersedia
Pangandaran (2) 11:15:00 Bandung 13:00:00 Tasikmalaya Rp 70000 Reservasi 10 kursi tersedia

Implementasi Basis Data

Struktur Table Penumpang

Server: 127.0.0.1 Database: keretakuy Tabel: penumpang

Jelajahi Struktur SQL Cari Tambahan Eksport Impor Hak Akses Operasi

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	idPenumpang	int(11)			Tidak	Tidak ada		AUTO_INCREMENT
2	namaPenumpang	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada		
3	emailPenumpang	varchar(30)	utf8mb4_general_ci		Tidak	Tidak ada		
4	idUser	int(3)		UNSIGNED ZEROFILL	Tidak	Tidak ada		

Contoh Data

Extra options

	Ubah	Salin	Hapus	idPenumpang	namaPenumpang	emailPenumpang	idUser
<input type="checkbox"/>				1	Aff Kurniawan Supriyadi	afiksupriyadi@gmail.com	001
<input type="checkbox"/>				2	Aydin Shidqi Athalla Karyanto	aydin@gmail.com	001
<input type="checkbox"/>				3	Reza Ilham Maulana	rezaIlham@gmail.com	001
<input type="checkbox"/>				4	Sutan Rifky Tedjasukmana	sutanrifky@gmail.com	006
<input type="checkbox"/>				5	Daffa Ananta Rachman	daffa@gmail.com	006
<input type="checkbox"/>				6	Mochamad Khoirullutfansyah	ori@gmail.com	006

NOTE:

Table Penumpang dan **Table TicketBooked**
akan digunakan untuk mengimplementasikan
Modul Booking dan **Modul Lihat Tiket** pada
website

Struktur Table TiketBooked

Server: 127.0.0.1 Database: keretakuy Tabel: ticketbooked

Jelajahi Struktur SQL Cari Tambahan Eksport Impor Hak Akses Operasi

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	idBooked	int(11)				Tidak	Tidak ada	AUTO_INCREMENT
2	idUser	int(3)			UNSIGNED ZEROFILL	Tidak	Tidak ada	
3	idPenumpang	int(11)				Tidak	Tidak ada	
4	idKereta	int(2)			UNSIGNED ZEROFILL	Tidak	Tidak ada	
5	idStasiunAsal	int(3)			UNSIGNED ZEROFILL	Tidak	Tidak ada	
6	idStasiunTujuan	int(3)			UNSIGNED ZEROFILL	Tidak	Tidak ada	
7	departureTime	time				Tidak	Tidak ada	
8	arrivalTime	time				Tidak	Tidak ada	
9	tanggalKeberangkatan	date				Tidak	Tidak ada	
10	harga	int(11)				Tidak	Tidak ada	
11	waktuBooking	datetime				Tidak	current_timestamp()	

Contoh Data

SELECT * FROM `ticketbooked`

Profil [Edit dikotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

Extra options

	Ubah	Salin	Hapus	idBooked	idUser	idPenumpang	idKereta	idStasiunAsal	idStasiunTujuan	departureTime	arrivalTime	tanggalKeberangkatan	harga	waktuBooking
<input type="checkbox"/>				1	001	1	01	001	003	06:00:00	08:15:00	2025-01-08	55000	2025-01-02 16:44:00
<input type="checkbox"/>				2	001	2	01	001	003	06:00:00	08:15:00	2025-01-08	55000	2025-01-02 16:44:00
<input type="checkbox"/>				3	001	3	01	001	003	06:00:00	08:15:00	2025-01-08	55000	2025-01-02 16:44:00
<input type="checkbox"/>				4	006	4	01	001	004	06:00:00	09:00:00	2025-01-06	75000	2025-01-02 21:07:44
<input type="checkbox"/>				5	006	5	01	001	004	06:00:00	09:00:00	2025-01-06	75000	2025-01-02 21:07:44
<input type="checkbox"/>				6	006	6	01	001	004	06:00:00	09:00:00	2025-01-06	75000	2025-01-02 21:07:44

Implementasi Basis Data

Contoh query melihat tiket yang telah dipesan:

Mencari tiket yang telah dipesan dari pengguna yang memiliki id == 001

```
386 • SELECT
387     tb.idBooked, tb.idUser, u.nama AS namaPembeli, tb.idPenumpang, p.namaPenumpang,
388     tb.idKereta, k.nama AS namaKereta, tb.idStasiunAsal, s1.nama AS stasiunAsal,
389     tb.idStasiunTujuan, s2.nama AS stasiunTujuan, tb.departureTime, tb.arrivalTime,
390     tb.tanggalKeberangkatan, tb.harga, tb.waktuBooking
391     FROM TicketBooked tb
392     JOIN User u ON tb.idUser = u.idUser
393     JOIN Penumpang p ON tb.idPenumpang = p.idPenumpang
394     JOIN Kereta k ON tb.idKereta = k.idKereta
395     JOIN Stasiun s1 ON tb.idStasiunAsal = s1.idStasiun
396     JOIN Stasiun s2 ON tb.idStasiunTujuan = s2.idStasiun
397     WHERE tb.idUser = '001'
398     ORDER BY tb.waktuBooking DESC;
399
```

Hasil pencarian tiket yang telah dipesan

	idBooked	idUser	namaPembeli	idPenumpang	namaPenumpang	idKereta	namaKereta	idStasiunAsal	stasiunAsal	idStasiunTujuan	stasiunTujuan	departureTime	arrivalTime	tanggalKeberangkatan	harga	waktuBooking
▶	2	001	Aff Kurniawan Supriyadi	2	Aydin Shidqi Athalla Karyanto	01	Pasundan	001	Bandung	003	Ciamis	06:00:00	08:15:00	2025-01-08	55000	2025-01-02 16:44:00
	3	001	Aff Kurniawan Supriyadi	3	Reza Ilham Maulana	01	Pasundan	001	Bandung	003	Ciamis	06:00:00	08:15:00	2025-01-08	55000	2025-01-02 16:44:00
	1	001	Aff Kurniawan Supriyadi	1	Aff Kurniawan Supriyadi	01	Pasundan	001	Bandung	003	Ciamis	06:00:00	08:15:00	2025-01-08	55000	2025-01-02 16:44:00

Halaman myticket pada website untuk menampilkan tiket yang telah dipesan

localhost:8080/TrainTicketBooking/myticket

My Tickets

#	Passenger Name	Train	From	To	Departure	Arrival	Booking Date	Price
1	Aydin Shidqi Athalla Karyanto	Pasundan	Bandung	Ciamis	06:00:00	08:15:00	2025-01-02T16:44	Rp 55000
2	Aff Kurniawan Supriyadi	Pasundan	Bandung	Ciamis	06:00:00	08:15:00	2025-01-02T16:44	Rp 55000
3	Reza Ilham Maulana	Pasundan	Bandung	Ciamis	06:00:00	08:15:00	2025-01-02T16:44	Rp 55000

Back to Homepage



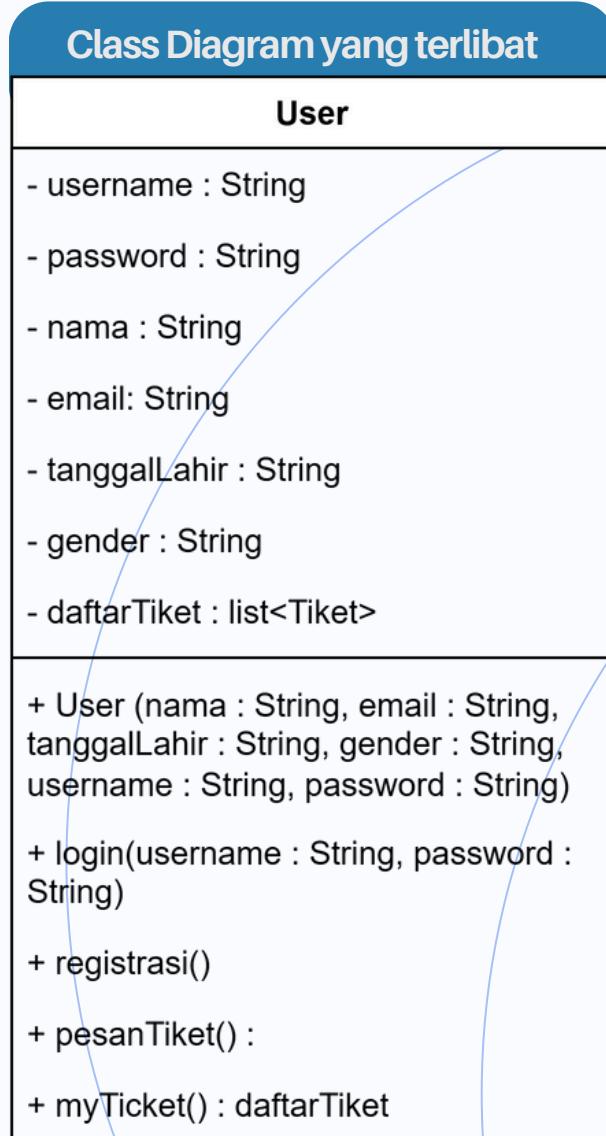
Implementasi Antarmuka dan Proses

List Fitur

- Login register - Daffa Ananta R
- Searching - Cristopher Daeng K
- Booking - Afif Kurniawan S
- MyTicket: view, cancel - M Rafli Adhan S
- Profile: view, edit - Aydin Shidqi Athalla K
- Daftar penumpang: view dan edit - M Ribhy Rizwanul I

Implementasi Antarmuka dan Proses

MODUL: REGISTER



- Untuk mengakses website, pengguna harus **memiliki akun**
- Jika user belum memiliki akun, pengguna dapat melakukan **registrasi akun** dengan mengklik **Daftar segera**.
- Pengguna akan diarahkan ke halaman **registrasi** untuk membuat akun

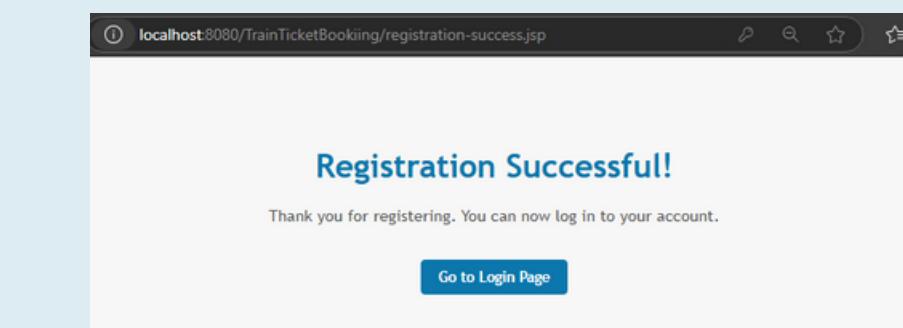
- Isi form registrasi dengan **Nama Lengkap, Email, Tanggal Lahir, Jenis Kelamin, Username, dan Password**. Jika semua data sudah dimasukkan kemudian klik **kirim**

Daftar Akun

Nama Lengkap	Email
Amer Hadafi Utomo	dafi@gmail.com
Tanggal Lahir	Jenis Kelamin
12 / 12 / 2024	Laki-Laki
Username	Password
dafi

Kirim

- Jika registrasi **berhasil**, akan tampilkan tulisan **registration successful** dan meminta user untuk **kembali ke halaman login**



Implementasi Antarmuka dan Proses

MODUL: LOGIN

Antarmuka Website

Selamat Datang di
Website Pemesanan Tiket Kereta

Masuk

Username

Password

Log In

Belum ada akun? [Daftar segera](#)

- Masukkan **username** dan **password** yang telah dibuat

NOTE:

Data diri pengguna yang telah disimpan dalam database akan diambil dan akan diubah menjadi **Objek** atau **Model User**, sebagai **representasi Data User**, yang merupakan hasil instansiasi **Class User** yang telah dirancang pada **Class Diagram**

- Jika username dan password **tidak valid**, akan ada pesan error bahwa **username atau password invalid**

Daftar segera'."/>

Masuk

Invalid username or password. Please try again.

Username

afifks

Password

.....

Log In

Belum ada akun? [Daftar segera](#)

- Jika username dan password **valid**, user akan diarahkan ke halaman **homepage**

Kereta Kuy

My Ticket

Pemesanan Tiket Kereta

Stasiun Asal

Cari Stasiun Asal...

Stasiun Tujuan

Cari Stasiun Tujuan...

Tanggal Keberangkatan

Pilih Tanggal

Jumlah Penumpang

1 Penumpang

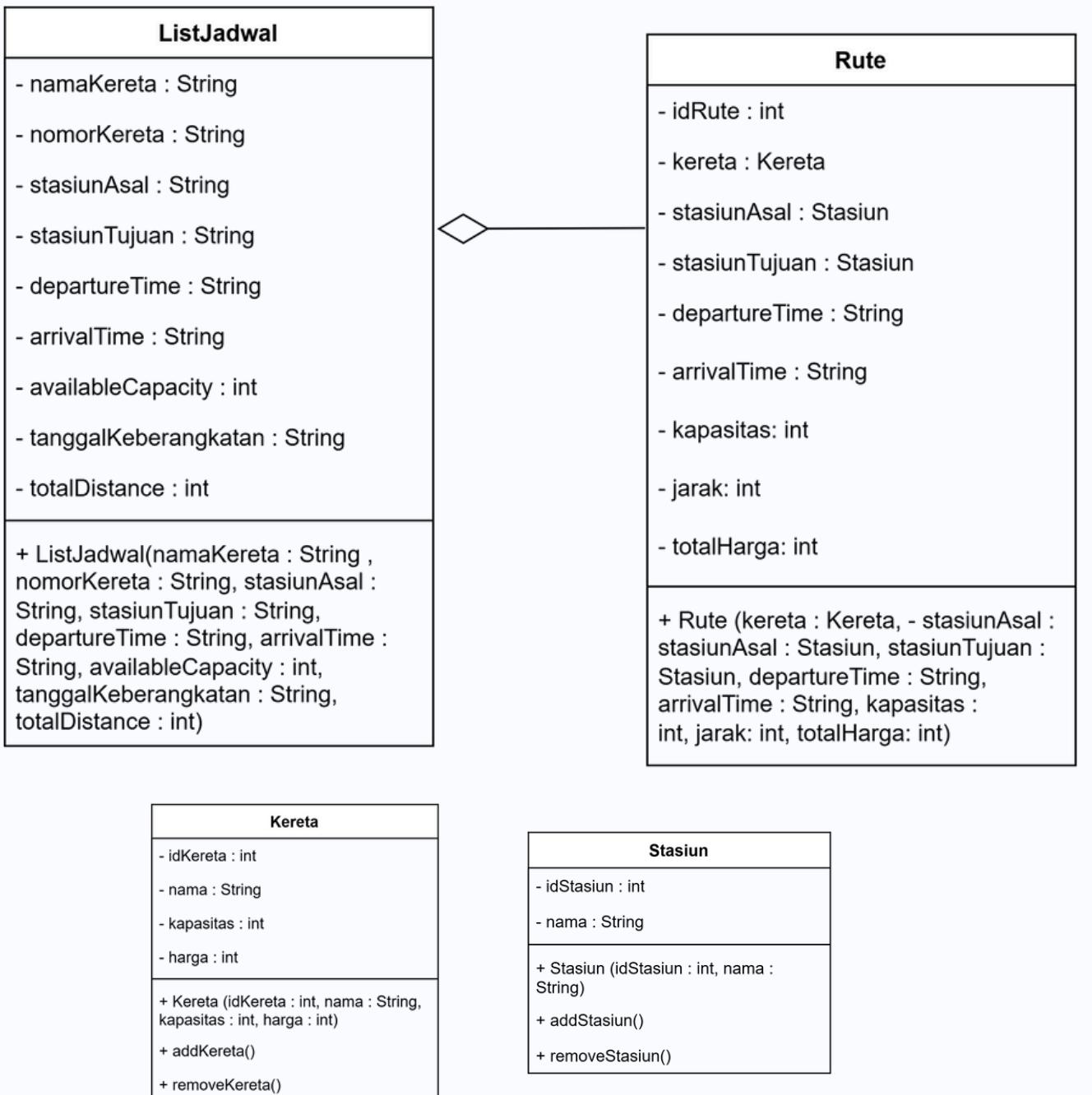
Cari Kereta

Implementasi Antarmuka dan Proses

MODUL: SEARCHING



Class Diagram yang terlibat



Antarmuka Website

Kereta Kuy

My Ticket

Pemesanan Tiket Kereta

Stasiun Asal Cari Stasiun Asal...

Stasiun Tujuan Cari Stasiun Tujuan...

Tanggal Keberangkatan Pilih Tanggal

Jumlah Penumpang 3 Penumpang

Cari Kereta

- Homepage website akan menampilkan **searching form** yang digunakan sebagai **filter pencarian** jadwal kereta.

Implementasi Antarmuka dan Proses

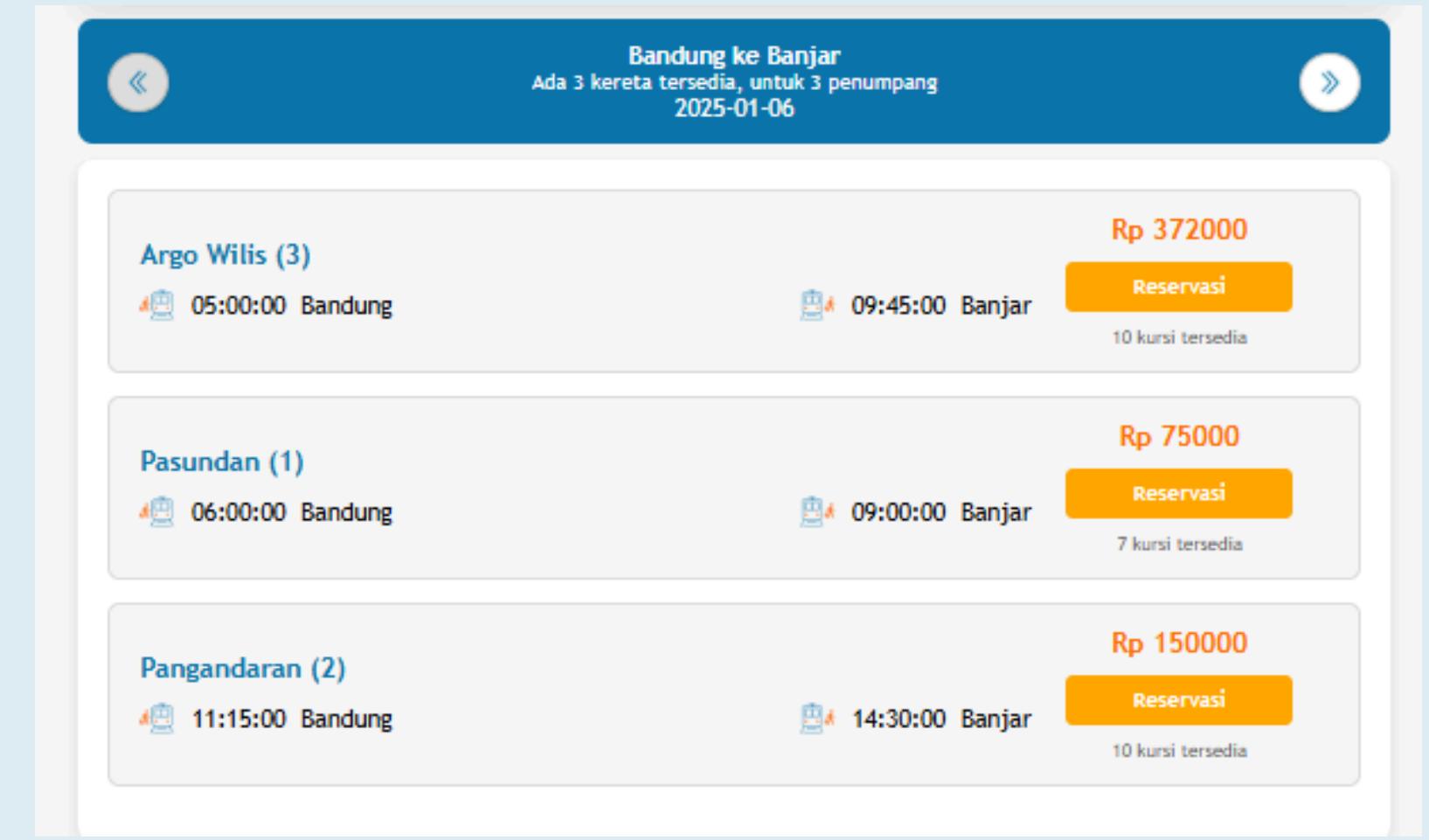
MODUL: SEARCHING

- Searching form akan berisi **parameter input** yang digunakan untuk pencarian jadwal kereta, seperti: **Stasiun Asal, Stasiun Tujuan, Tanggal Keberangkatan, dan Jumlah Penumpang**
- Sebagai contoh, misalkan akan dilakukan pencarian jadwal kereta untuk:
 - Stasiun Asal: **Bandung**
 - Stasiun Tujuan: **Banjar**
 - Tanggal Keberangkatan: **06 Januari 2025**
 - Jumlah Penumpang: **3 Penumpang**

NOTE:

- **Hasil** modul searching adalah menampilkan **daftar pilihan kereta atau jadwal** yang tersedia. **Jadwal** ini ditampilkan dalam bentuk **Model ListJadwal**, yang merupakan **hasil instansiasi objek** dari **Class ListJadwal**, sesuai rancangan pada **Class Diagram**

- Klik **cari kereta** untuk mencari kereta. Jika **terdapat** jadwal kereta yang tersedia, hasil pencarian berupa **list jadwal kereta** akan ditampilkan.



Implementasi Antarmuka dan Proses

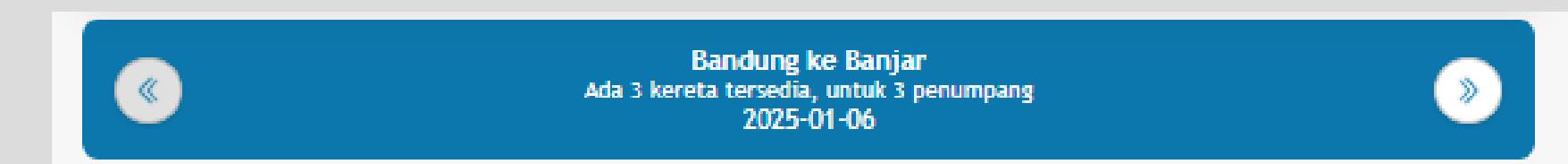
MODUL: SEARCHING

- Jika **tidak terdapat** jadwal kereta yang tersedia, akan ditampilkan pesan bahwa **tidak ada kereta yang ditemukan**

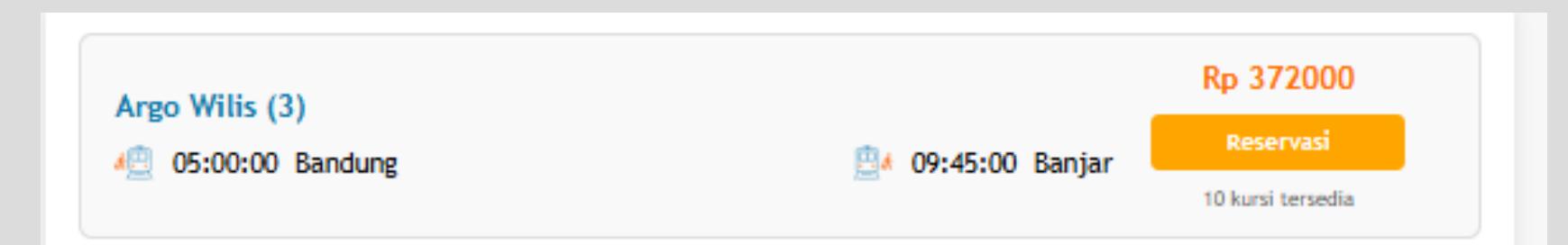
The screenshot shows a search form titled "Pemesanan Tiket Kereta". It includes fields for "Stasiun Asal" (Bandung), "Stasiun Tujuan" (Bekasi), "Tanggal Keberangkatan" (Pilih Tanggal), and "Jumlah Penumpang" (1 Penumpang). A "Cari Kereta" button is present. Below the form, a blue bar displays "Bandung ke Bekasi" with the message "Ada 0 kereta tersedia, untuk 1 penumpang". At the bottom, a red message states "Tidak ada kereta yang ditemukan."

Detail hasil pencarian kereta:

- Date navigation:** user dapat melakukan pencarian tiket menggunakan button **arrow** pada sisi kiri dan kanan navigation, untuk mencari tiket berdasarkan **tanggal**, dengan **catatan** bahwa data stasiun asal, stasiun tujuan, dan jumlah penumpang **sama seperti** yang dimasukkan user pada **searching form**.



- Searching result:** baris atau list jadwal kereta yang tersedia. Satu jadwal kereta akan berisi informasi: **Nama dan ID kereta, Jam Keberangkatan, Jam Tiba, Stasiun Asal, Stasiun Tujuan, Harga Tiket, dan Jumlah kursi yang tersedia**



Klik **Reservasi** untuk melanjutkan ke proses booking.

Implementasi Antarmuka dan Proses

MODUL: BOOKING

Antarmuka Website

Pangandaran (2)
11:15:00 Bandung → 14:30:00 Banjar

Rincian Biaya
Harga Tiket: Rp 150000
Jumlah Tiket: x3
Total: Rp 450000

Data Penumpang
Data Penumpang 1
Nama Penumpang: Amer Hadafi Utomo
Email: dafi@gmail.com

Data Penumpang 2
Nama Penumpang:
Email:

Data Penumpang 3
Nama Penumpang:
Email:

Pesan

- Halaman booking menampilkan **review jadwal kereta** yang **telah dipilih** dan perhitungan **total harga tiket** sesuai dengan **banyaknya** jumlah penumpang yang dimasukkan.

NOTE:

- Hasil modul booking adalah menyimpan tiket yang telah dipesan oleh pengguna. Tiket disimpan dalam bentuk Model Tiket, yang merupakan **hasil instansiasi objek** dari **Class Tiket**

- Data penumpang ke-1** akan terisi otomatis oleh **data**

Data Penumpang

Data Penumpang 1
Nama Penumpang: Amer Hadafi Utomo
Email: dafi@gmail.com

- Jika pengguna **memasukkan** jumlah penumpang **lebih dari 1**, maka **form** data penumpang **akan menyesuaikan** sesuai banyaknya jumlah penumpang
- Isi data penumpang tersebut **dengan nama dan email**.

Data Penumpang

Data Penumpang 1
Nama Penumpang: Amer Hadafi Utomo
Email: dafi@gmail.com

Data Penumpang 2
Nama Penumpang: Mukhlis Nur Hidayat
Email: muis@gmail.com

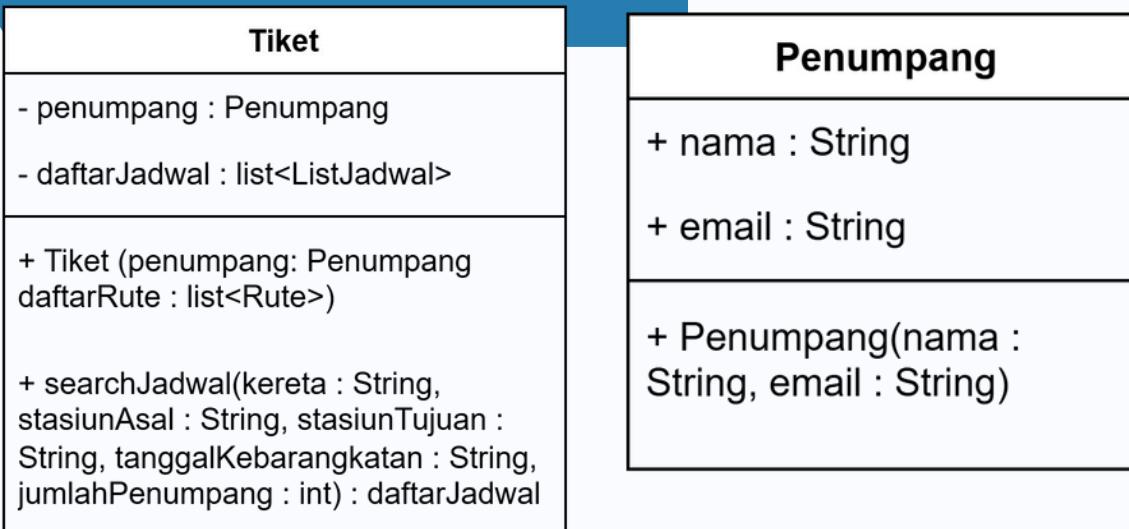
Data Penumpang 3
Nama Penumpang: Ridho Chan
Email: dodang@gmail.com

Pesan

Implementasi Antarmuka dan Proses

MODUL: BOOKING

Class Diagram yang terlibat

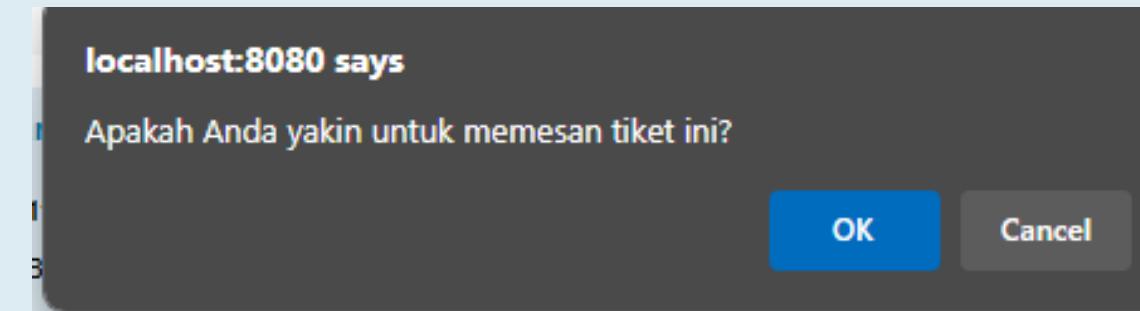


NOTE:

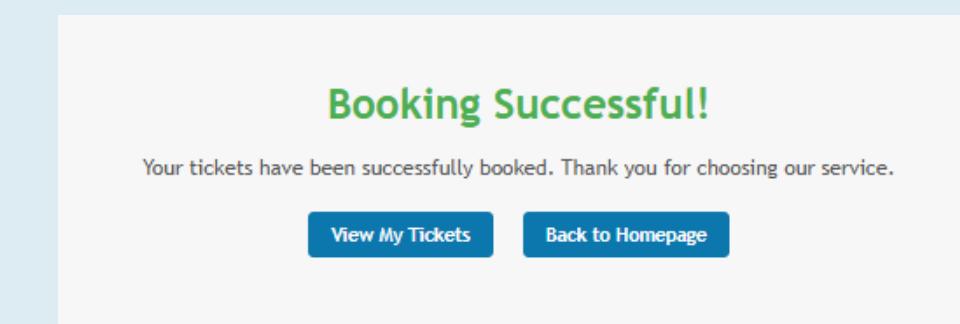
Terdapat 3 logika yang berjalan dibelakang layar, saat button **Pesan** diklik oleh pengguna untuk melakukan proses booking:

1. **[Insert data to Table Penumpang]**: Sistem memasukkan data penumpang ke table penumpang.
2. **[Update Table Rute]**: Sistem akan mengurangi kapasitas kursi yang tersedia pada table rute, menyesuaikan jadwal kereta yang dipilih oleh pengguna.
3. **[Insert data to Table Tiket]**: Sistem memasukkan data tiket yang berhasil dipesan ke table Ticket

- Klik **Pesan** untuk memesan tiket dan klik **OK** pada validasi **alert** yang muncul untuk konfirmasi pemesanan



- Pemesanan tiket berhasil, akan ditampilkan halaman **Booking Success**



- User dapat kembali ke **Homepage** atau melihat data tiket yang telah di pesan dengan klik **View My Tickets**

Implementasi Antarmuka dan Proses

MODUL: PROFILE

Class Diagram yang terlibat

User

- username : String
- password : String
- nama : String
- email: String
- tanggalLahir : String
- gender : String

+ User (nama : String, email : String, tanggalLahir : String, gender : String, username : String, password : String)

Antarmuka Website

User Profile

Informasi Pengguna

Nama: Aydin
Email: aydin@gmail.com
Gender: Laki-laki
Tanggal Lahir: 2004-10-01
Username: aydin

Edit User

Nama:
Aydin

Email:
aydin@gmail.com

Gender:
Laki-laki

Tanggal Lahir:
01 / 10 / 2004

Username:
aydin

Password:
.....

Update

NOTE:

User dapat melihat informasi pengguna sekaligus meng-edit profile nya jika ingin diubah.

Jika User tersebut mengubah **username** tersebut menjadi **username** yang sudah ada di dalam **database**, pesan **error** akan ditampilkan di halaman profile

Edit User

Username sudah ada yang dipakai

Dan ketika User meng-edit lalu menekan tombol **Update** namun membiarkan salah satu **box input** kosong, pesan **warning** akan muncul

Nama:
|

Email:
! Please fill out this field.

Implementasi Antarmuka dan Proses

MODUL: VIEW TIKET

Class Diagram yang terlibat	
User	
- username : String	
- password : String	
- nama : String	
- email: String	
- tanggalLahir : String	
- gender : String	
- daftarTiket : list<Tiket>	
+ User (nama : String, email : String, tanggalLahir : String, gender : String, username : String, password : String)	
+ login(username : String, password : String)	
+ registrasi()	
+ pesanTiket() :	
+ myTicket() : daftarTiket	

 Kereta Kuy

My Tickets

#	Nama Penumpang	Kereta	Stasiun Asal	Stasiun Tujuan	Jam Keberangkatan	Jam Tiba	Tanggal Keberangkatan	Waktu Booking	Harga	Action
1	rafli	Argo Wilis	Bandung	Cipeundeuy	05:00:00	06:30:00	2025-01-08	2025-01-05T20:26:12	Rp 6336000	<button>Cancel Ticket</button>
2	ocang	Argo Wilis	Bandung	Cipeundeuy	05:00:00	06:30:00	2025-01-08	2025-01-05T20:26:12	Rp 6336000	<button>Cancel Ticket</button>
3	ayyub	Argo Wilis	Bandung	Cipeundeuy	05:00:00	06:30:00	2025-01-08	2025-01-05T20:26:12	Rp 6336000	<button>Cancel Ticket</button>

[Back to Homepage](#)

NOTE:

Pengguna dapat melihat data tiket yang telah dipesan melalui **View My Ticket** pada halaman **Booking Success** atau melalui **My Ticket** yang ada pada header di setiap halaman website, user juga dapat melakukan cancel tiket.

Pengujian

- Kami memilih class **SearchingDAO** untuk dilakukan pengujian, yaitu menguji fitur **Searching** untuk mencari jadwal atau tiket kereta yang tersedia
- Data masukkan yang digunakan untuk pengujian mencari tiket atau jadwal kereta adalah:
 - Stasiun Awal: **Bandung**
 - Stasiun Tujuan: **Banjar**
 - Tanggal Keberangkatan: **2025 - 01 - 06**
 - Jumlah Penumpang: **1**

```

1 import dao.SearchingDAO;
2 import model.ListKereta;
3 import org.junit.jupiter.api.BeforeEach;
4 import org.junit.jupiter.api.Test;
5 import java.util.List;
6 import static org.junit.jupiter.api.Assertions.*;
7
8 public class SearchingDAOTest {
9     private SearchingDAO searchingDAO;
10    @BeforeEach
11    public void setUp() throws Exception {
12        // Initialize SearchingDAO
13        searchingDAO = new SearchingDAO();
14    }
15    @Test
16    public void testSearchKeretaValid() {
17        // Test with valid parameters
18        List<ListKereta> results = searchingDAO.searchKereta("Bandung", "Banjar", "2025-01-06", 1);
19        assertNotNull(results, "Results should not be null");
20        assertFalse(results.isEmpty(), "Results should not be empty");
21
22        // Verify the first result
23        ListKereta train = results.get(0);
24        assertEquals("Argo Wilis", train.getNamaKereta(), "Train name should match");
25        assertEquals(310, train.getTotalDistance(), "Total distance should match");
26    }
27    @Test
28    public void testSearchKeretaInvalid() {
29        // Test with invalid parameters
30        List<ListKereta> results = searchingDAO.searchKereta("Bandung", "UnknownStation", "2025-01-06", 1);
31        assertNotNull(results, "Results should not be null");
32        assertTrue(results.isEmpty(), "Results should be empty for invalid search");
33    }
34}

```

- Pengujian berhasil dilakukan dengan hasil testing **100%**, artinya tiket atau jadwal tiket yang dicari berhasil ditemukan/tersedia.

Test Results

com:TrainTicketBooking:war:1.0-SNAPSHOT (Unit)

Tests passed: 100,00 %

Both tests passed. (0,525 s)

Output

Tubes - D:\IF-46-07 - Semester 5\05 - PBO\Tubes x Test (SearchingDAOTest) x

[war]

--- resources:3.3.1:resources (default-resources) @ TrainTicketBooking ---
Copying 1 resource from src\main\resources to target\classes

--- compiler:3.8.1:compile (default-compile) @ TrainTicketBooking ---
Nothing to compile - all classes are up to date

--- resources:3.3.1:testResources (default-testResources) @ TrainTicketBooking ---
Copying 2 resources from src\test\resources to target\test-classes

--- compiler:3.8.1:testCompile (default-testCompile) @ TrainTicketBooking ---
Changes detected - recompiling the module!
Compiling 3 source files to D:\IF-46-07 - Semester 5\05 - PBO\Tubes\TrainTicketBooking\target\test-classes

--- surefire:3.2.5:test (default-cli) @ TrainTicketBooking ---
Using auto detected provider org.apache.maven.surefire.junitplatform.JUnitPlatformProvider

T E S T S

Running SearchingDAOtest
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'.
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.525 s -- in SearchingDAOtest

Results:

Tests run: 2, Failures: 0, Errors: 0, Skipped: 0

BUILD SUCCESS

Total time: 7.140 s
Finished at: 2025-01-03T21:05:52+07:00

Thank You So Much

