

Dokumen Teknis

Pengembangan Aplikasi Booking Tiket Tomok Ajibata berbasis Website

Menggunakan Arsitektur Microservice

Tugas Proyek Akhir Semester

Mata Kuliah: Pengembangan Aplikasi Terdistribusi

Dipersiapkan oleh:

11422002	Benyamin Sibarani
11422010	David Kristian Silalahi
11422025	Desrico Siallagan
11422049	Alvina Siallagan
11422050	Listra Imelda Sidabutar
11422058	Yessi Sipahutar

Untuk:

**Institut Teknologi Del
2024**



**PROYEK
PENGEMBANGAN APLIKASI
TERDISTRIBUSI INSTITUT TEKNOLOGI
DEL 2024**

DAFTAR ISI

1	Pendahuluan	1
1.1	Deskripsi Umum Aplikasi	1
1.2	Karakteristik Pengguna Aplikasi	2
1.3	Fungsi pada Aplikasi	3
2	Desain Rancangan Aplikasi	4
2.1	Use Case Diagram	4
2.2	Business Process Modeling Notation	4
2.2.1	Proses Bisnis Registrasi	5
2.2.2	Proses Bisnis Login	5
2.2.3	Proses Bisnis Mengelola Produk	5
2.2.4	Proses Bisnis Melakukan Order Produk	6
2.2.5	Proses Bisnis Melakukan Update Profile	6
2.2.6	Proses Bisnis Logout	7
3	Tampilan Aplikasi	8
4	Pengujian Aplikasi	14

DAFTAR GAMBAR

Gambar 1	4
Gambar 2	5
Gambar 3	6
Gambar 4	6
Gambar 5	7
Gambar 6	8
Gambar 7	9
Gambar 8	9
Gambar 9	10

Pendahuluan

Pada bab 1 berisi tentang deskripsi umum aplikasi, karakteristik pengguna aplikasi, dan fungsi pada aplikasi.

1.1 Deskripsi Umum aplikasi

Berdasarkan sistem yang ada saat ini, kami akan mengembangkan sebuah sistem yang dapat memenuhi kebutuhan klien kami dengan fokus pada informasi pelabuhan Tomok - Ajibata melalui sebuah website. Website ini ditujukan untuk petugas pelabuhan dalam mengelola informasi terkait kapal, jadwal keberangkatan, dan pemesanan tiket kapal. Tujuan utama dari pengembangan aplikasi berbasis web ini adalah untuk membantu dalam pengelolaan data agar lebih terstruktur dan terintegrasi dengan baik, menyediakan pemberitahuan jadwal keberangkatan yang pasti, serta mempermudah pengguna dalam memesan tiket dan melakukan booking kapal.

Beberapa fungsi dan fitur yang akan dikembangkan meliputi login dan Register, mengelola rute, mengelola kapal, mengelola nahkoda, mengelola jadwal dan memesan tiket. Selain itu, aplikasi ini juga akan menyediakan fitur dimana pelanggan dapat mengisi data kendaraan secara online untuk mempermudah proses pemesanan tiket. Pengguna dalam aplikasi ini akan berperan sebagai pemesan tiket dan pengguna informasi kapal.

1.2 Fungsi pada Aplikasi

Terdapat 7 fungsi service yang akan dikembangkan yaitu :

1. Fungsi register pengguna

Fungsi register digunakan user penumpang untuk mendaftarkan akun nya, dan untuk akun pemilik kapal, ditambahkan oleh admin.

2. Fungsi Login pengguna

Fungsi login digunakan user pengguna untuk dapat masuk dan mengakses fitur fitur yang terdapat pada aplikasi

3. Fungsi mengelola rute

Fungsi mengelola rute digunakan untuk menambahkan rute dan mengubah rute. Rute ini berguna sebagai salah satu informasi dari keberangkatan kapal pada suatu jadwal

4. Fungsi mengelola kapal

Fungsi mengelola kapal digunakan sebagai detail yang mengatur dan mengolah informasi kapal.

5. Fungsi mengelola nahkoda

Fungsi mengelola nahkoda digunakan untuk menambahkan nahkoda dan melakukan edit pada informasi nahkoda

6. Fungsi mengelola jadwal

Fungsi mengelola jadwal digunakan untuk menambah jadwal, yang mempunyai informasi berupa rute, kapal yang bertugas, nahkoda yang mengendarai, waktu keberangkatan, stok tiket, dan harga per tiket

7. Memesan tiket

Fungsi memesan tiket digunakan user penumpang untuk memesan tiket jadwal kapal yang sudah tersedia.

2 Bahasa Pemrograman

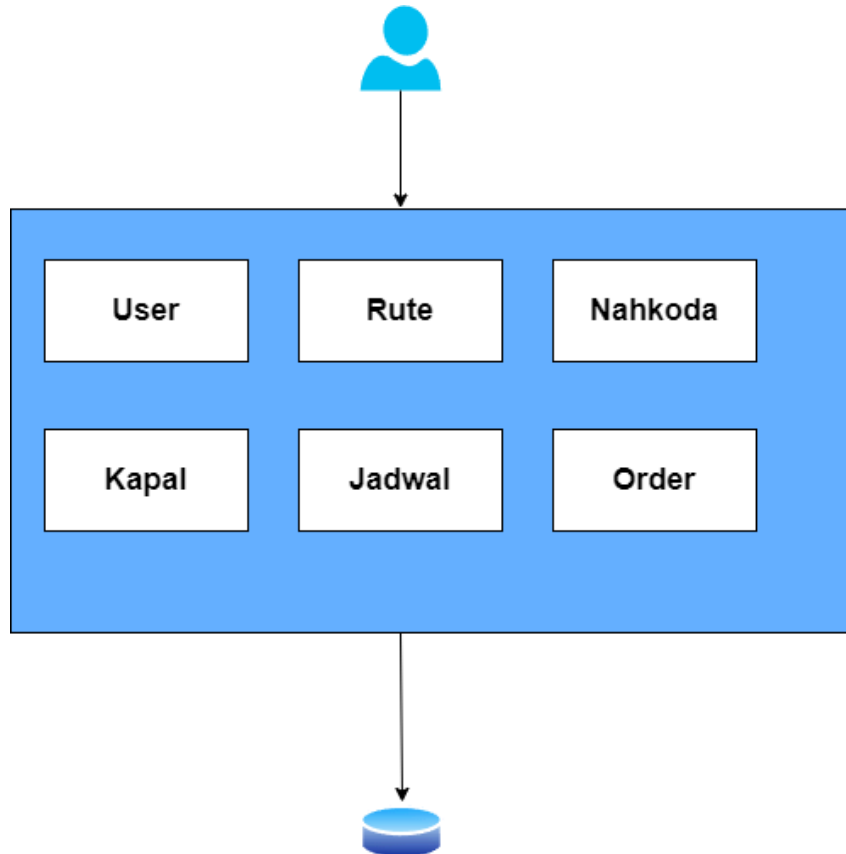
Arsitektur *microservice* yang digunakan pada Pengembangan Aplikasi Booking Tiket Tomok Ajibata berbasis Website. REST API digunakan sebagai penghubung antara *web ui* dengan *service* yang tersedia sehingga memungkinkan *service* dapat dikonsumsi oleh *web ui*. Hal inilah yang menyebabkan *service* dapat diakses oleh beberapa *platform* tanpa harus terbatas pada satu bahasa pemrograman saja. Sama halnya dengan pembangunan aplikasi booking tiket tomok ajibata ini. Bahasa yang digunakan pada bagian *front-end* adalah bahasa PHP (*framework Laravel*) dan pada bagian *back-end* menggunakan bahasa Go. Meskipun memiliki bahasa yang berbeda, tetapi aplikasi akan tetap dapat berjalan dengan adanya REST API.

Pada bagian *back-end*, setiap *service* memiliki *database*-nya masing-masing. Pada aplikasi web ini terdapat enam *service* beserta masing-masing *database*-nya. Setiap *service* dirancang memiliki *port* yang berbeda sehingga *port* tidak akan bertabrakan. Tetapi untuk port semua database, kami menggunakan port yang sama, yaitu 3306.

3 Arsitektur Monolith dan Microservices

Pada bab 3 berisi tentang perbedaan arsitektur monolith dan microservices dalam SYP (second year project).

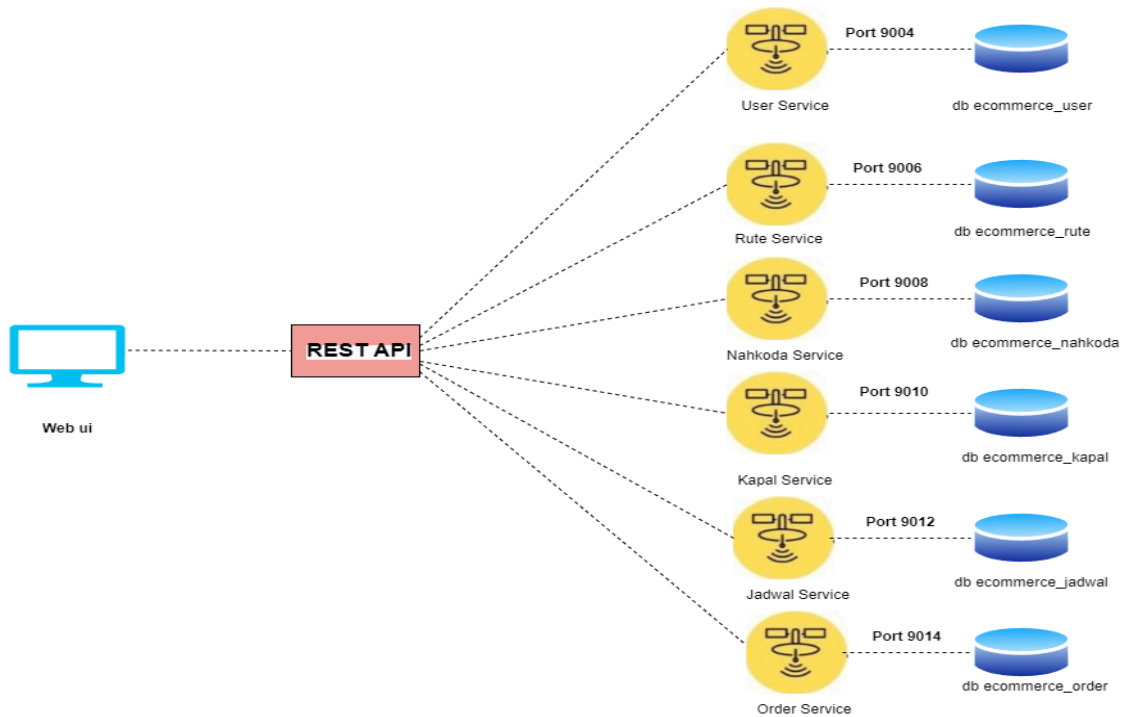
- Arsitektur monolith adalah pendekatan di mana seluruh fungsi dan layanan dari sebuah aplikasi disatukan dalam satu kode dasar (codebase) yang tunggal. Pendekatan ini menawarkan kesederhanaan dan kemudahan pengembangan, namun bisa menjadi rumit dan sulit untuk dipertahankan seiring perkembangan aplikasi. Seperti tabel User, Rute, Nahkoda, Kapal, Jadwal, dan Order hanya berada pada satu database saja.



Gambar 1

- Arsitektur microservices adalah pendekatan di mana aplikasi dibagi menjadi sejumlah layanan kecil yang berdiri sendiri dan dapat dikembangkan, diimplementasikan, dan

diskalakan secara independen. Setiap layanan biasanya bertanggung jawab atas fungsi tertentu dalam aplikasi.



Gambar 2

4 Antarmuka Pengguna

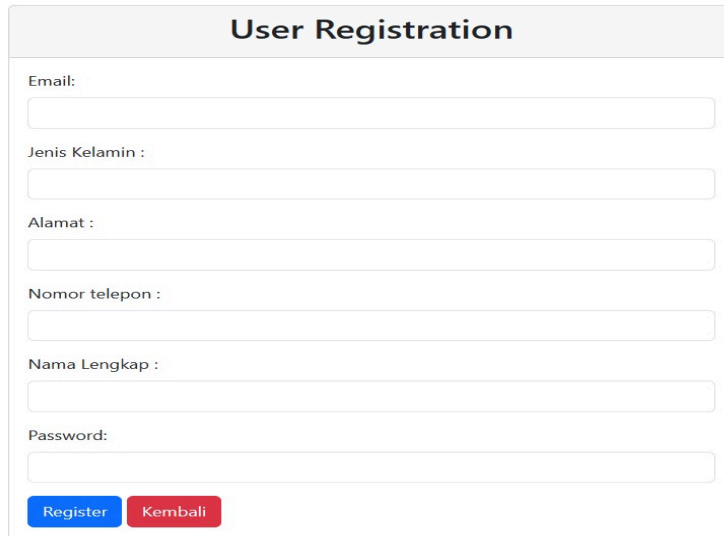
Pada bab 4 berisi tentang arsitektur layanan mikro yang terdiri dari beberapa prosesor independen, masing-masing mencakup komponen umum dari aplikasi perusahaan seperti antarmuka pengguna, basis data, server, serta model tarik dan dorong.

4.1 Antarmuka Pengguna

Bagian yang berinteraksi langsung dengan pengguna. Dalam layanan mikro, UI adalah layanan terpisah yang dikembangkan dan dikelola secara independen dari backend.

1. Tampilan Halaman Registrasi

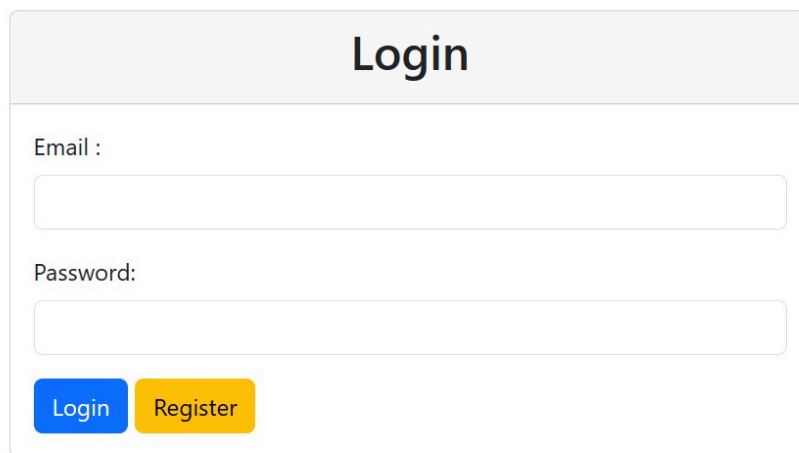
Tampilan halaman registrasi ini digunakan untuk mendaftarkan akun user



The 'User Registration' form is a vertical layout with a light gray header. It contains six text input fields, each preceded by a label: 'Email:', 'Jenis Kelamin:', 'Alamat:', 'Nomor telepon:', 'Nama Lengkap:', and 'Password:'. At the bottom, there are two buttons: a blue 'Register' button and a red 'Kembali' button.

Gambar 3

2. Tampilan Halaman Login
Tampilan halaman login ini digunakan untuk masuk dan dapat mengakses halaman yang lebih banyak lagi



The 'Login' form has a light gray header with the title 'Login'. Below the header, there are two text input fields labeled 'Email :' and 'Password:'. At the bottom, there are two buttons: a blue 'Login' button and a yellow 'Register' button.

Gambar 4

3. Tampilan Halaman Mengelola Rute
Halaman mengelola rute digunakan untuk menambahkan dan mengedit rute kapal

AjibataTomok

Dashboard

KAPAL

Pemilik kapal

Nakhoda

KELOLA JADWAL

Rute

Jadwal

ANOTHER

Pages

Dashboards

Logout

#	Lokasi Berangkat	Lokasi Tujuan	Action
1	Ajibata	Tomok2	<div>Edit</div>
2	Tomok	Medan	<div>Edit</div>
3	Medan	Bekasi	<div>Edit</div>
4	Hinalang	Medan	<div>Edit</div>

Form Edit Rute

Lokasi Berangkat :

Lokasi tujuan :

Update

Tambah Rute

Lokasi Berangkat:

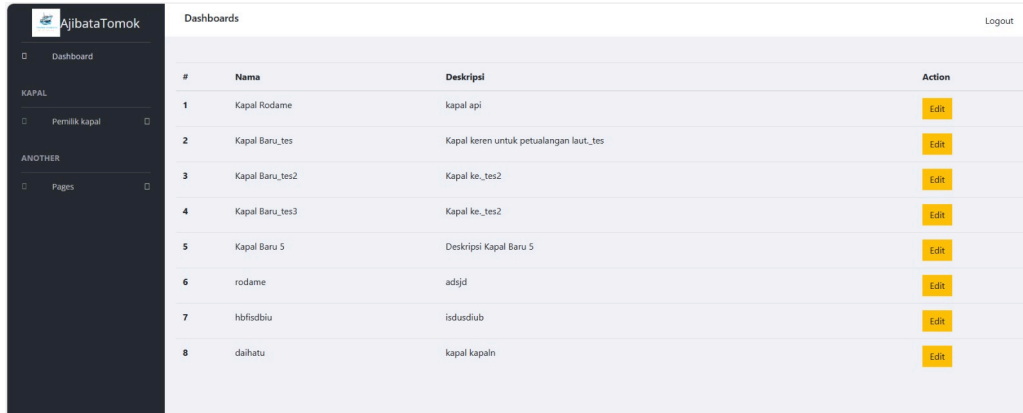
Lokasi Tujuan:

Tambah Rute

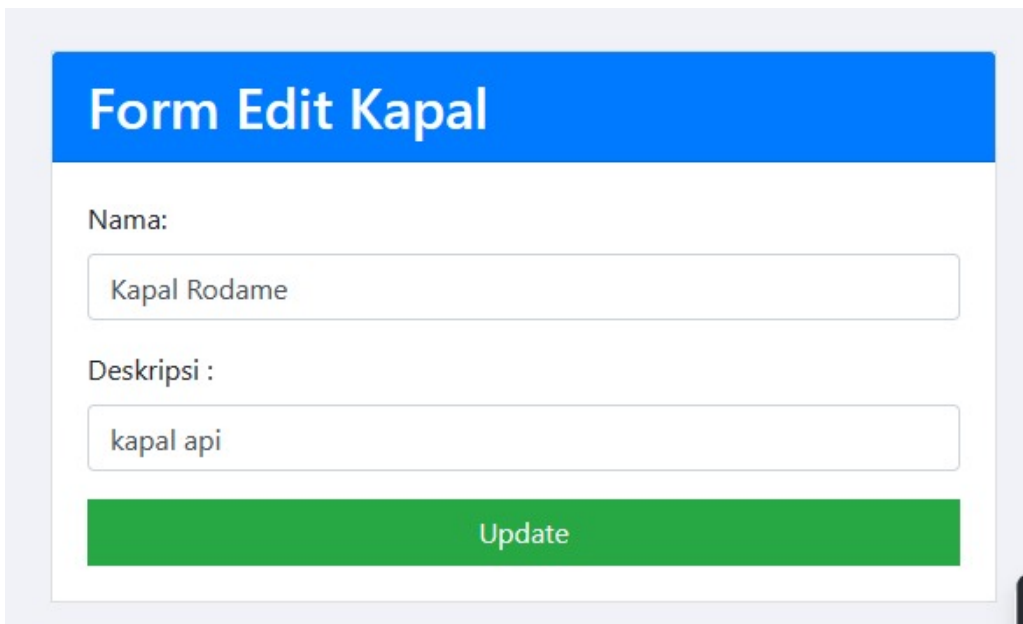
Gambar 5

4. Tampilan Halaman Mengelola Kapal

Tampilan halaman mengelola kapal digunakan untuk menambahkan dan mengedit kapal



#	Nama	Deskripsi	Action
1	Kapal Rodame	kapal api	Edit
2	Kapal Baru_tes	Kapal keren untuk petualangan laut_tes	Edit
3	Kapal Baru_tes2	Kapal ke_tes2	Edit
4	Kapal Baru_tes3	Kapal ke_tes2	Edit
5	Kapal Baru 5	Deskripsi Kapal Baru 5	Edit
6	rodame	adjsjd	Edit
7	hbfsdibu	isdudub	Edit
8	daihatu	kapal kapaln	Edit



Form Edit Kapal

Nama:

Kapal Rodame

Deskripsi :

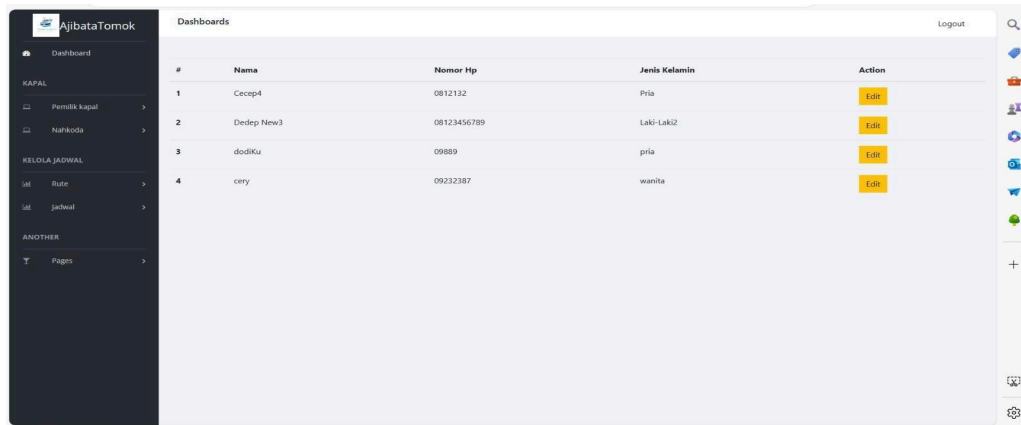
kapal api

Update

Gambar 6

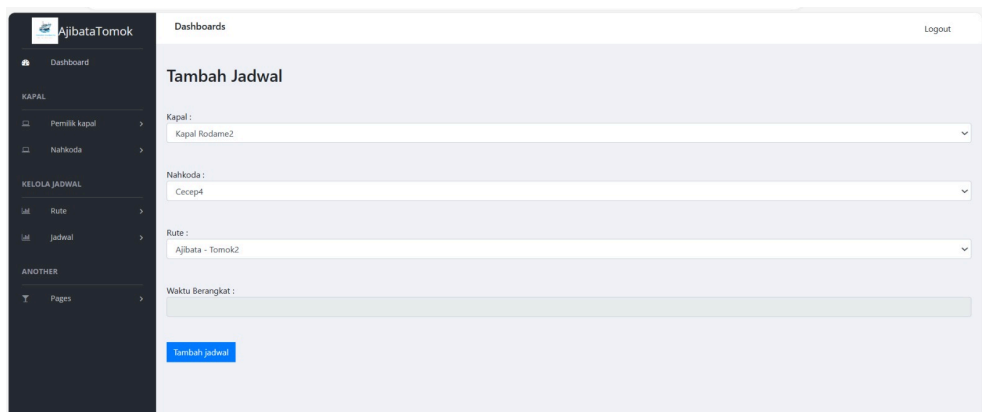
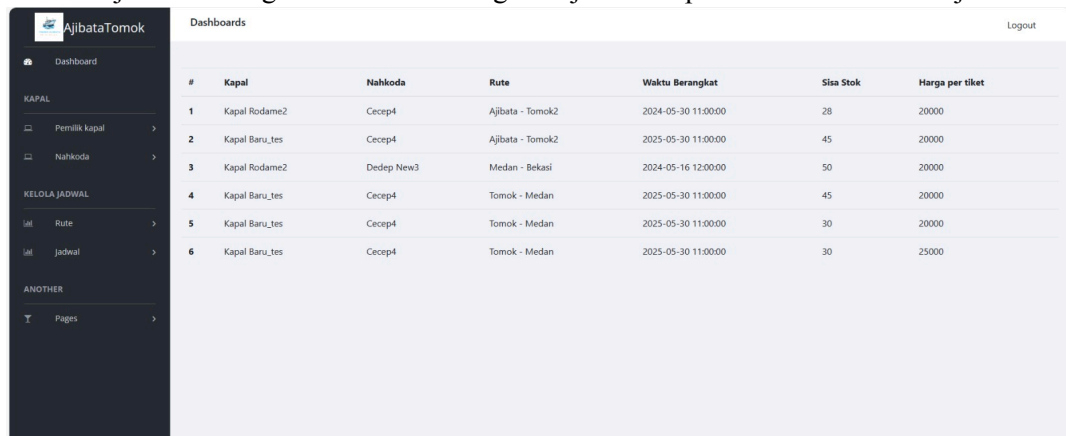
5. Tampilan Halaman Mengelola Nahkoda

Halaman mengelolah nahkoda digunakan untuk menambah dan mengedit nahkoda



Gambar 7

6. Tampilan Halaman Mengelolah Jadwal
Halaman jadwal ini digunakan untuk mengelola jadwal kapal dan menambahkan jadwal kapal



Gambar 8

7. Tampilan Halaman Memesan Tiket

Cek info terbaru Kapal Tomok Tour tiket Kapal dan jadwal Keberangkatan kapal Tomok Tour Ajibata.

Home Pesan Tiket Jadwal Orderan LOGOUT

Keberangkatan : 2024-05-30 11:00:00
Rute : Ajibata - Tomok2
Kapal : Kapal Rodame2
Nahkoda : Cecep4
Sisa stok : 28

Keberangkatan : 2025-05-30 11:00:00
Rute : Ajibata - Tomok2
Kapal : Kapal Baru_tes
Nahkoda : Cecep4
Sisa stok : 45

Keberangkatan : 2024-05-16 12:00:00
Rute : Medan - Bekasi
Kapal : Kapal Rodame2
Nahkoda : Dedep New3
Sisa stok : 50

Keberangkatan : 2025-05-30 11:00:00
Rute : Tomok - Medan
Kapal : Kapal Baru_tes
Nahkoda : Cecep4
Sisa stok : 30

Keberangkatan : 2025-05-30 11:00:00
Rute : Tomok - Medan
Kapal : Kapal Baru_tes
Nahkoda : Cecep4
Sisa stok : 30

Order ID	Jadwal ID	Quantity	Total Amount
1	1	2	40000
2	2	1	20000
3	2	3	60000
4	1	3	60000
8	1	2	40000
9	2	5	100000

Home Features Pricing FAQs About

AjibataTomok Dashboards Logout

Ship Owner Registration

Email:

Jenis Kelamin:

Alamat:

Nomor telepon:

Nama Lengkap:

Password:

Gambar 9

4.2 Basis Data (DataBase)

Tempat penyimpanan data. Dalam arsitektur microservices, setiap layanan biasanya memiliki basis data sendiri yang terpisah. Berbeda dengan dengan pendekatan monolitik, di mana satu basis data besar digunakan oleh seluruh aplikasi.

Dengan memiliki basis data terpisah, setiap layanan dapat dikelola secara independen. mengurangi ketergantungan dengan layanan lain dan memungkinkan pengembangan yang mandiri, di mana masalah di satu layanan tidak akan secara langsung mempengaruhi layanan lain.

4.3 Server

Tempat eksekusi logika aplikasi. Setiap layanan mikro berjalan pada server terpisah, memungkinkan pengembangan, pengujian, dan penerapan yang independen. Setiap microservice berjalan pada server atau container terpisah, dan memungkinkan setiap layanan untuk di-deploy, diskalakan, dan diperbarui secara independen tanpa harus menghentikan seluruh aplikasi.

Pengembang sistem dapat memperbarui satu layanan tanpa harus khawatir tentang dampak pada layanan lainnya. Jika ada masalah atau bug dalam satu layanan, hanya layanan tersebut yang perlu diperbaiki dan di-deploy ulang, sementara layanan lainnya tetap berjalan tanpa gangguan.

4.4 Model Tarik dan Dorong (Push and Pull Model)

Dalam model pull, setiap layanan mengambil data yang diperlukan dari layanan lainnya saat diminta oleh klien melalui permintaan REST API. Misalnya, ketika pengguna memuat halaman di aplikasi UI dan memerlukan informasi tentang jadwal kapal, layanan UI akan membuat permintaan ke layanan jadwal kapal melalui REST API untuk menarik data jadwal tersebut.

Dalam model push, layanan yang menghasilkan data secara aktif mengirimkan data tersebut ke layanan lain yang memerlukannya tanpa diminta terlebih dahulu oleh klien. Misalnya, jika ada perubahan dalam informasi kapal, layanan kapal akan mengirim pembaruan ke layanan lain yang bergantung padanya, seperti layanan yang menangani pembayaran, tanpa menunggu permintaan dari klien.

