

# 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 user penumpang**

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

#### **2. Fungsi register admin pemilik kapal**

Fungsi register digunakan admin pemilik kapal untuk mendaftarkan akunnya

#### **3. Fungsi Login pengguna**

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

#### **4. Fungsi mengedit rute**

Fungsi mengedit rute digunakan untuk mengedit rute yang sudah di tambahkan.

#### **5. Fungsi menambah rute**

Fungsi menambah rute digunakan untuk menambah rute.

6. Fungsi mengedit nahkoda

Fungsi mengedit nahkoda digunakan untuk mengedit nahkoda yang sudah di tambahkan.

7. Fungsi menambah nahkoda

fungsi menambah nahkoda digunakan untuk menambahkan nahkoda kapal.

8. Fungsi menambah jadwal

Fungsi menambah jadwal digunakan untuk menambah nahkoda kadal

9. Fungsi menambah kapal

Fungsi menambah kapal digunakan untuk menambahkan kapal baru

10. Fungsi mengedit kapal

Fungsi mengedit kapal digunakan untuk mengedit deskripsi kapal

11. Fungsi melakukan pemesanan

Fungsi melakukan pemesanan dilakukan oleh user penumpang

## 2 Bahasa Pemrograman

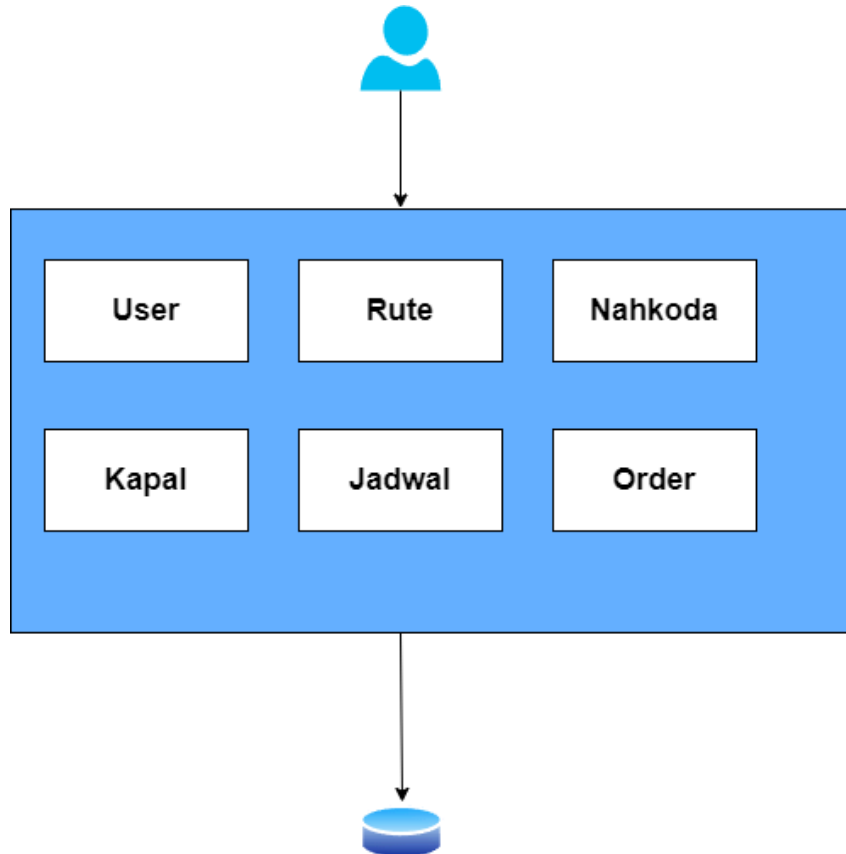
Arsitektur *microservice* yang digunakan pada Pengembangan Aplikasi Booking Tiket Tomok Ajibata berbasis Website. REST API digunakan sebagai penghubung antara *web ui* dengan *service* yang tersedia sehingga memungkinkan *service* dapat dikonsumsi oleh *web ui*. Hal inilah yang menyebabkan *service* 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).

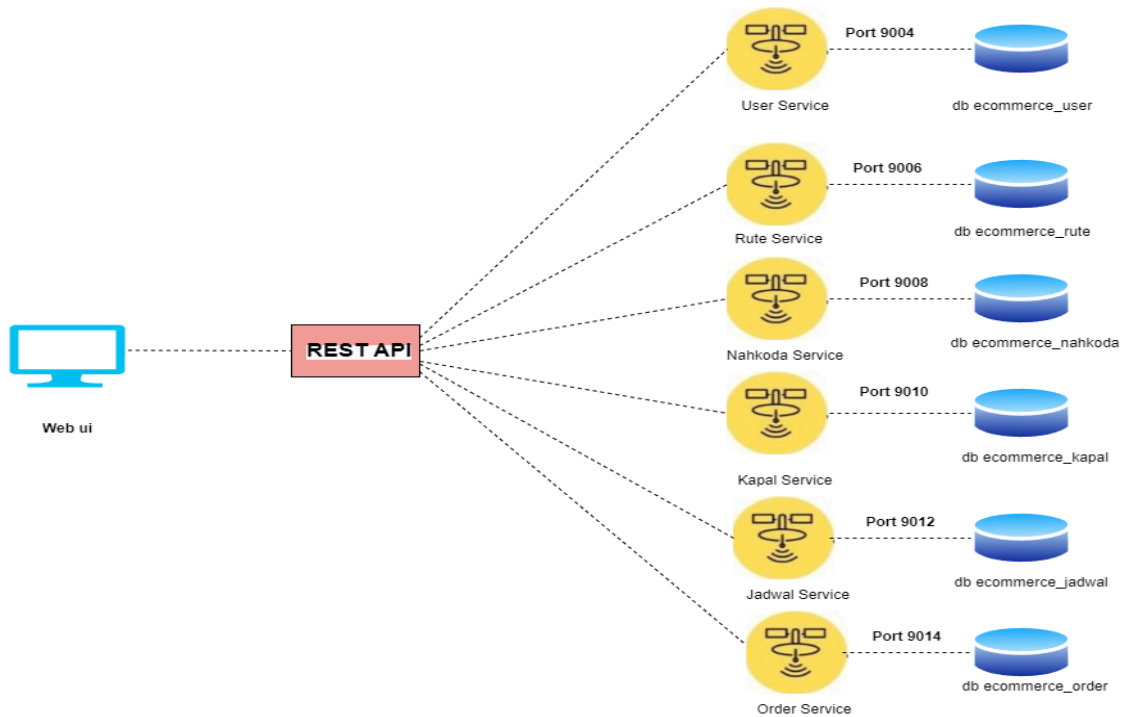
- Arsitektur monolith adalah pendekatan di mana seluruh fungsi dan layanan dari sebuah aplikasi disatukan dalam satu kode dasar (codebase) yang tunggal. Pendekatan ini menawarkan kesederhanaan dan kemudahan pengembangan, namun bisa menjadi rumit dan sulit untuk dipertahankan seiring perkembangan aplikasi. 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.

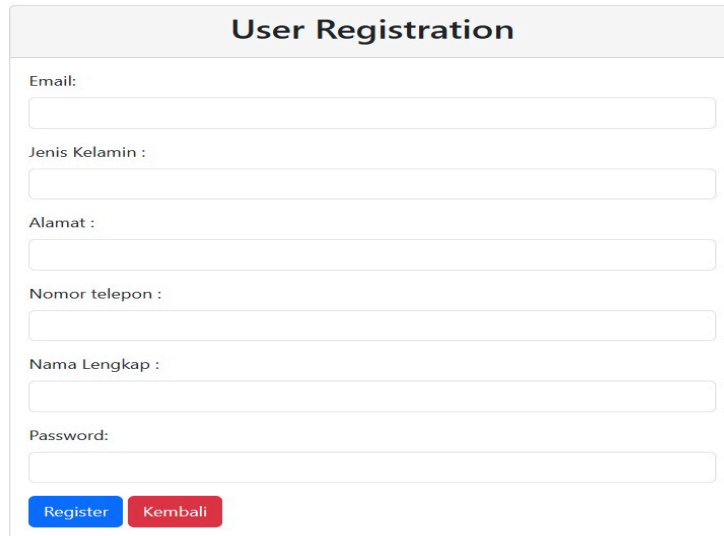
### 4.1 Antarmuka Pengguna

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

Hal ini mengacu pada bagian aplikasi yang berinteraksi langsung dengan pengguna. Pada arsitektur microservice, antarmuka pengguna ini biasanya adalah aplikasi web atau aplikasi seluler yang memanggil berbagai layanan microservice untuk mendapatkan data dan melakukan operasi yang diperlukan. Frontend (antarmuka pengguna) tidak terikat langsung dengan backend tertentu. Antarmuka hanya mengetahui endpoint API yang disediakan oleh berbagai layanan microservice (Decoupled Frontend and Backend). UI dapat memuat konten secara dinamis dari berbagai layanan microservice, misalnya, memuat jadwal kapal dari service jadwal, atau menampilkan informasi user dari service user.

#### 1. Tampilan Halaman Registrasi

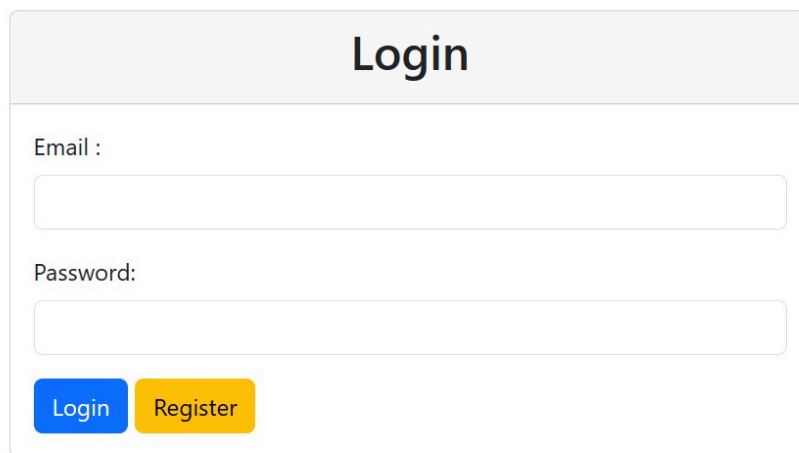
Tampilan halaman registrasi ini digunakan untuk mendaftarkan akun user



The form is titled "User Registration" in a grey header. It contains several input fields: "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 form is titled "Login" in a grey header. It contains two input fields: "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



- Dashboard
- KAPAL
  - Pemilik kapal
  - Nakhoda
- KELOLA JADWAL
  - Rute
  - Jadwal
- ANOTHER
  - Pages

Logout

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

## Form Edit Rute

Lokasi Berangkat :

Lokasi tujuan :

Update

## Tambah Rute

Lokasi Berangkat:

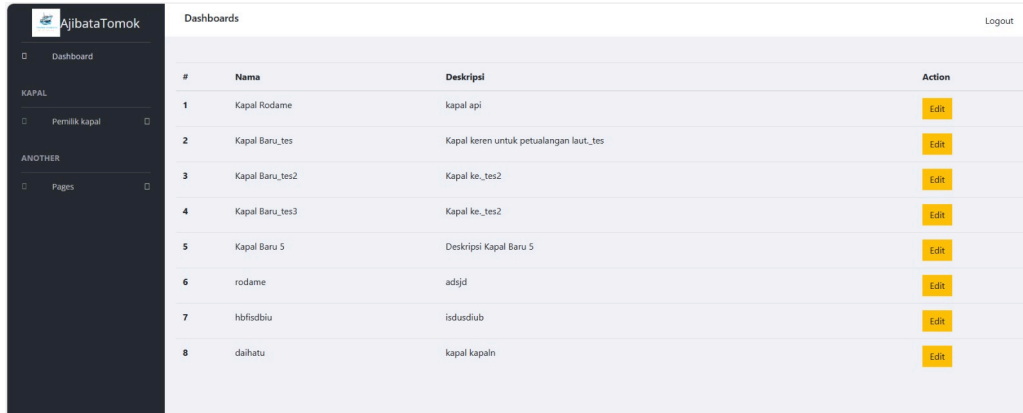
Lokasi Tujuan:

Tambah Rute

Gambar 5

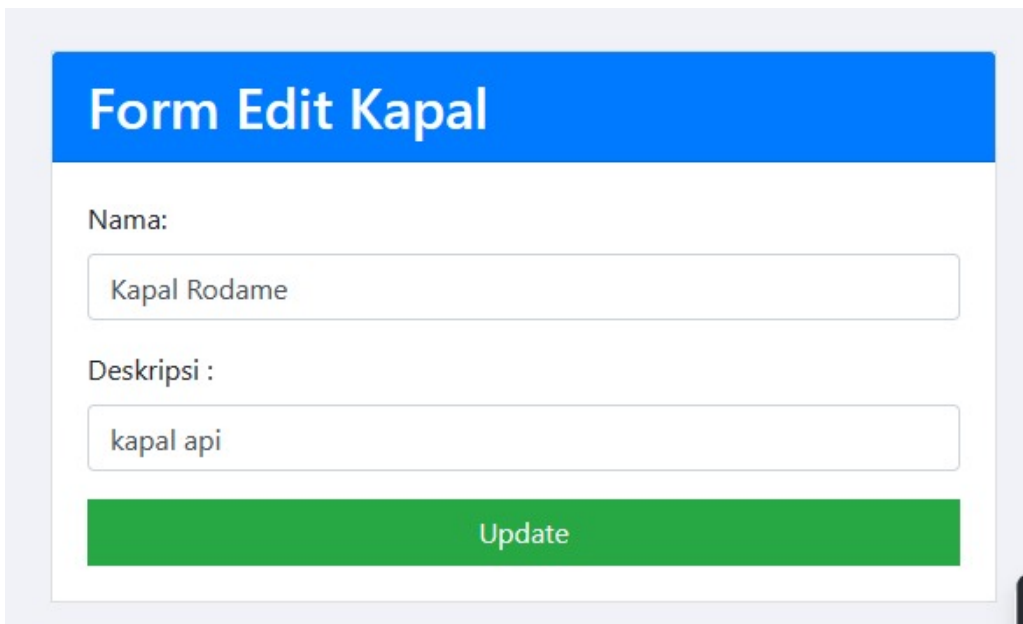
#### 4. Tampilan Halaman Mengelola Kapal

Tampilan halaman mengelola kapal digunakan untuk menambahkan dan mengedit kapal



The screenshot shows a web application interface for 'AjibataTomok'. On the left is a dark sidebar with navigation links: 'Dashboard', 'KAPAL', 'Pemilik kapal', and 'ANOTHER' with a 'Pages' link. The main content area is titled 'Dashboards' and contains a table with 8 rows of ship data. Each row has an 'Action' column with an 'Edit' button. The table columns are '#', 'Nama', 'Deskripsi', and 'Action'.

#	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	adjsd	Edit
7	hbfsdbiu	isdudub	Edit
8	daihatu	kapal kapaln	Edit



The screenshot shows a form titled 'Form Edit Kapal'. It has two input fields: 'Nama:' with the value 'Kapal Rodame' and 'Deskripsi :' with the value 'kapal api'. Below the fields is a large green button labeled 'Update'.

Gambar 6

#### 5. Tampilan Halaman Mengelola Nahkoda

Halaman mengelolah nahkoda digunakan untuk menambah dan mengedit nahkoda

#	Nama	Nomor Hp	Jenis Kelamin	Action
1	Cecep4	0812132	Pria	Edit
2	Dedep New3	08123456789	Laki-Laki2	Edit
3	dodiku	09889	pria	Edit
4	cery	09232387	wanita	Edit

Gambar 7

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

#	Kapal	Nahkoda	Rute	Waktu Berangkat	Sisa Stok	Harga per tiket
1	Kapal Rodame2	Cecep4	Ajibata - Tomok2	2024-05-30 11:00:00	28	20000
2	Kapal Baru_tes	Cecep4	Ajibata - Tomok2	2025-05-30 11:00:00	45	20000
3	Kapal Rodame2	Dedep New3	Medan - Bekasi	2024-05-16 12:00:00	50	20000
4	Kapal Baru_tes	Cecep4	Tomok - Medan	2025-05-30 11:00:00	45	20000
5	Kapal Baru_tes	Cecep4	Tomok - Medan	2025-05-30 11:00:00	30	20000
6	Kapal Baru_tes	Cecep4	Tomok - Medan	2025-05-30 11:00:00	30	25000

**Tambah Jadwal**

Kapal :  
Kapal Rodame2

Nahkoda :  
Cecep4

Rute :  
Ajibata - Tomok2

Waktu Berangkat :

[Tambah jadwal](#)

Gambar 8

## 7. Tampilan Halaman Memesan Tiket

The image displays two screenshots of the AjibataTomok website. The top screenshot shows the 'Pesanan Tiket Jadwal' (Ticket Schedule Orders) page, which lists six available routes for booking. Each route includes the departure date and time, route details, ship name, captain name, and remaining stock. The bottom screenshot shows the 'Dashboards' page for ship owners, featuring a 'Ship Owner Registration' form with fields for email, phone number, and password, along with a sidebar menu for managing ship and schedule data.

**Ticket Booking Interface (Top Screenshot):**

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

Navigation: Home, Pesanan Tiket Jadwal, Orderan, LOGOUT

**Ticket Listings:**

- Keberangkatan : 2024-05-30 11:00:00**  
Rute : Ajibata - Tomok2  
Kapal : Kapal Rodame2  
Nahkoda : Cecep4  
Sisa stok : 28  
 mau pesan berapa tiket
- Keberangkatan : 2025-05-30 11:00:00**  
Rute : Ajibata - Tomok2  
Kapal : Kapal Baru\_tes  
Nahkoda : Cecep4  
Sisa stok : 45  
 mau pesan berapa tiket
- Keberangkatan : 2024-05-16 12:00:00**  
Rute : Medan - Bekasi  
Kapal : Kapal Rodame2  
Nahkoda : Dedep New3  
Sisa stok : 50  
 mau pesan berapa tiket
- Keberangkatan : 2025-05-30 11:00:00**  
Rute : Tomok - Medan  
Kapal : Kapal Baru\_tes  
Nahkoda : Cecep4  
Sisa stok : 45  
 mau pesan berapa tiket
- Keberangkatan : 2025-05-30 11:00:00**  
Rute : Tomok - Medan  
Kapal : Kapal Baru\_tes  
Nahkoda : Cecep4  
Sisa stok : 30  
 mau pesan berapa tiket
- Keberangkatan : 2025-05-30 11:00:00**  
Rute : Tomok - Medan  
Kapal : Kapal Baru\_tes  
Nahkoda : Cecep4  
Sisa stok : 30  
 mau pesan berapa tiket

**Order History Table (Bottom Screenshot):**

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

Footer: Home Features Pricing FAQs About

**Ship Owner Registration Form (Bottom Screenshot):**

Header: AjibataTomok, Dashboards, Logout

Form Fields:

- Email:
- Jenis Kelamin:
- Alamat:
- Nomor telepon:
- Nama Lengkap:
- Password:
- 

Gambar 9

## 4.2 Basis Data (DataBase)

Microservice Database adalah pendekatan di mana setiap layanan microservice memiliki database tersendiri. Hal ini mendukung isolasi dan otonomi masing-masing layanan. Dalam konteks aplikasi pemesanan jadwal kapal, Setiap layanan, seperti service user, service kapal, service nahkoda, service rute, service jadwal, dan service order, memiliki database tersendiri. Hal ini meminimalisir dampak jika terjadi masalah pada satu layanan, karena layanan lain tidak akan terpengaruh. Semua layanan dapat menggunakan jenis database yang sama (misalnya MySQL yang berjalan di port 3306), tetapi setiap layanan memiliki skema database yang berbeda.

### **4.3 Server**

Microservice Server adalah infrastruktur yang mendukung pelaksanaan layanan microservice secara independen. Setiap layanan berjalan pada server yang terpisah atau dalam container yang terpisah, dengan port yang berbeda-beda:

Setiap layanan dapat di-deploy, di-skalakan, dan diperbarui secara independen. Misalnya, service user dapat berjalan di server dan port tertentu, sementara service jadwal di server dan port lain.

Jika salah satu server yang menjalankan layanan tertentu mengalami masalah, hanya layanan tersebut yang terpengaruh, sedangkan layanan lain tetap berjalan normal. Misalnya, jika server yang menjalankan service kapal down, service user tetap berfungsi.

### **4.4 Model Tarik dan Dorong (Push and Pull Model)**

Pull Model :

berarti client atau service meminta (menarik) data atau informasi dari server atau service lain secara eksplisit.

Ketika pengguna membuka aplikasi untuk melihat jadwal kapal, antarmuka pengguna (UI) mengirimkan permintaan (request) ke service jadwal untuk menarik data jadwal kapal terbaru.

UI atau layanan microservice lainnya melakukan panggilan API (HTTP GET, POST, dll.) ke endpoint yang disediakan oleh layanan terkait. Misalnya, UI melakukan GET request ke service-jadwal/api/jadwal untuk mendapatkan daftar jadwal. Data hanya diambil saat ada permintaan dari client.

Push Model :

server atau service mendorong (mengirimkan) data atau informasi ke client atau service lain tanpa perlu diminta terlebih dahulu.

Untuk fungsi tentang perubahan informasi jadwal seperti pembaruan stok tiket, service jadwal dapat mendorong informasi langsung ke client.

Untuk implementasi, aplikasi dapat menggunakan message ketika selesai melakukan aksi seperti memesan jadwal, di mana akan ada pesan yang tampil di user interface berupa stok tiket yang terupdate. Ketika ada perubahan stok tiket, service jadwal dapat mendorong update langsung ke UI.



