

**DPPL-xx**

## **DESKRIPSI PERANCANGAN PERANGKAT LUNAK**

### **MNote**

untuk:

Restoran D'Happy

Dipersiapkan oleh:

Kelompok 2

Gagah Aji Gunadi - 1301204093

Laurentius Yudhistira A. - 1301180303

Renaldi Mahardika Putra B. - 1301200531


Sydney Salma Nur H. - 1301202545

Syahdan Naufal Nur I. - 1301204110

Program Studi S1 Informatika

Fakultas Informatika, Universitas Telkom

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

	<b>Prodi S1- Informatika Universitas Telkom</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<b><i>DPPL-xx</i></b> <xx:no grp>		<#>/<jml #
		<b>Revisi</b>	<nomor revisi>	Tgl: <isi tanggal>

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	
<b>G</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

## Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

# Daftar Isi

Daftar Isi	3
Daftar Tabel	6
Daftar Gambar	7
1. Pendahuluan	7
1.1 Tujuan Penulisan Dokumen	8
1.2 Lingkup Masalah	8
1.3 Definisi dan Istilah	8
1.4 Referensi	8
2 Perancangan Global	9
2.1 Rencana Lingkungan Implementasi	9
2.2 Deskripsi Arsitektur Perangkat Lunak	9
3 Perancangan Rinci	10
3.1 Realisasi Use Case	10
3.1.1 Use Case #1: Register	10
3.1.1.1 Use Case Scenario #1 Register	10
3.1.1.2 UI Design dan Deskripsi Objek UI #1 Register	11
3.1.1.3 Identifikasi Object dan Tipe nya #1 Register	11
3.1.1.4 Sequence Diagram #1 Register	12
3.1.2 Use Case #2 : Login	12
3.1.2.1 Use Case Scenario #2: Login	12
3.1.2.2 UI Design dan Deskripsi Objek UI #2: Login	13
3.1.2.3 Identifikasi Object Baru & Tipe Kelas #2 Login	13
3.1.2.4 Sequence Diagram #2 Login	13
3.1.3 Use Case #3 : Search Order	14
3.1.3.1 Use Case Scenario #3: Search Order	14
3.1.3.2 UI Design dan Deskripsi Objek UI #3: Search Order	15
3.1.3.3 Identifikasi Object Baru & Tipe Kelas #3 Search Order	15
3.1.3.4 Sequence Diagram #3 Search Order	16
3.1.4 Use Case #4 :Dashboard	16
3.1.4.1 Use Case Scenario #4: Dashboard	16
3.1.4.2 UI Design dan Deskripsi Objek UI #4: Dashboard	17
3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 Dashboard	17
3.1.4.4 Sequence Diagram #4 Dashboard	18
3.1.5 Use Case #5 : Summary	18
3.1.5.1 Use Case Scenario #5 : Summary	18
3.1.5.2 UI Design dan Deskripsi Objek UI #2: Summary	19
3.1.5.3 Identifikasi Object Baru & Tipe Kelas #5 : Summary	19
3.1.5.4 Sequence Diagram #5 : Summary	20

3.1.6 Use Case #6 :Setting	20
3.1.6.1 Use Case Scenario #6 :Setting	20
3.1.6.2 UI Design dan Deskripsi Objek UI #6 :Setting	21
3.1.6.3 Identifikasi Object Baru & Tipe Kelas #6 :Setting	21
3.1.6.4 Sequence Diagram #6 :Setting	22
3.1.7 Use Case #7 : Help	22
3.1.7.1 Use Case Scenario #7 : Help	22
3.1.7.2 UI Design dan Deskripsi Objek UI #7 : Help	23
3.1.7.3 Identifikasi Object Baru & Tipe Kelas #7 : Help	23
3.1.7.4 Sequence Diagram #7 : Help	24
3.1.8 Use Case #8 : Add Order	24
3.1.8.1 Use Case Scenario #8 : Add Order	24
3.1.8.2 UI Design dan Deskripsi Objek UI #8 : Add Order	25
3.1.8.3 Identifikasi Object Baru & Tipe Kelas #8 : Add Order	25
3.1.8.4 Sequence Diagram #8 : Add Order	26
3.1.9 Use Case #9 : Delete Order	26
3.1.9.1 Use Case Scenario #9 : Delete Order	26
3.1.9.2 UI Design dan Deskripsi Objek UI #9 : Delete Order	27
3.1.9.3 Identifikasi Object Baru & Tipe Kelas #9 : Delete Order	27
3.1.9.4 Sequence Diagram #9 : Delete Order	28
3.2 Diagram Kelas Keseluruhan	29
3.3 Perancangan Data / Basis Data	30
3.4 Perancangan Algoritma dan/atau Query	30
3.4.1 Algoritma/Query Auth	30
3.4.2 Algoritma/Query Order	30
3.4.3 Algoritma/Query Help	31
3.4.4 Algoritma/Query User	31
4. Matriks Keruntutan (Requirement Traceability Matrix)	31

## Daftar Gambar

<b>Daftar Isi</b>	<b>3</b>
<b>Daftar Gambar</b>	<b>6</b>
Gambar 1 Register	11
Gambar 2 Login	13
Gambar 3 Search Order	15
Gambar 4 Dashboard	17
Gambar 5 Summary	19
Gambar 6 Settings	21
Gambar 7 Help	23
Gambar 8 Add Order	25
Gambar 9 Delete Order	27

# 1. Pendahuluan

## 1.1 Tujuan Penulisan Dokumen

Tujuan dari penulisan dokumen ini adalah sebagai dokumen spesifikasi kebutuhan sistem perangkat lunak dari *software* yang telah kami rancang dengan nama **MNote**, yaitu sebuah aplikasi untuk mencatat segala bentuk order pada Restoran D'Happy. Dalam dokumen ini akan dijelaskan secara rinci dan bertahap mengenai aplikasi **MNote**.

## 1.2 Lingkup Masalah

**MNote** adalah aplikasi berbasis website dengan basis utama pada PC Windows yang berisi layanan pencatatan order secara keseluruhan beserta riwayat order pada masa sebelumnya. Aplikasi **MNote** bertujuan untuk membantu pemilik Restoran D'Happy dalam mencatat order dengan dibantu oleh sistem untuk meminimalisir terjadinya kesalahan dalam proses pencatatan.

## 1.3 Definisi dan Istilah

Dalam dokumen ini terdapat beberapa istilah yang digunakan sebagai nama fungsionalitas pada sistem. Definisi dari istilah tersebut dijelaskan pada tabel berikut.

Register	Proses untuk membuat akun baru agar dapat mengakses sistem
Login	Proses untuk mengakses sistem
User	Aktor yang mengoperasikan sistem
Database	Pusat penyimpanan data dari aplikasi

## 1.4 Referensi

1. Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) MNote.

## 2 Perancangan Global

### 2.1 Rencana Lingkungan Implementasi

Lingkungan implementasi untuk client, aplikasi ini kami desain untuk dapat beroperasi pada berbagai browser pada desktop seperti Edge, Chrome, dan browser lainnya.

Untuk bagian server dapat dijalankan melalui localhost pada perangkat keras dari pengguna, karena aplikasi ini didesain khusus untuk restoran D'Happy saja, jadi tidak perlu dijalankan pada web-server.

Teknologi yang digunakan pada aplikasi ini yaitu sebagai berikut

Front-end : ReactJS

Back-end : NodeJS, ExpressJS

Database : MongoDB

### 2.2 Deskripsi Arsitektur Perangkat Lunak

Diisi dengan gambar **komponen diagram** atau **daftar modul**, dapat dituliskan dalam bentuk tabel berikut:

No	Nama Komponen	Keterangan
	Login	Menu utama untuk masuk ke aplikasi MNote
	Register	Menu pendaftaran akun untuk user
	Dashboard	Halaman utama dari MNote yang menampilkan data order
	Summary	Halaman yang menampilkan ringkasan order pada waktu tertentu
	Search	Halaman yang menampilkan data order berdasarkan nama
	Settings	Halaman untuk mengubah data user
	Help	Halaman untuk menampilkan bantuan untuk user
	User	Aktor atau pengguna aplikasi MNote



### 3 Perancangan Rinci

#### 3.1 Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut :

No	Nama Use Case	Deskripsi Use Case
#1	Login	Mengakses menu aplikasi dengan akun user
#2	Register	Halaman mendaftarkan akun user
#3	Dashboard	Halaman yang menampilkan seluruh rincian order pada waktu tersebut
#4	Summary	Halaman yang menampilkan data order pada waktu tertentu
#5	Search	Halaman yang menampilkan data order berdasarkan nama item
#6	Settings	Halaman untuk mengubah data user
#7	Help	Halaman untuk menampilkan bantuan
#8	Add Order	Fitur menambahkan data order baru
#9	Delete Order	Fitur menghapus data order

##### 3.1.1 Use Case #1: Register

##### 3.1.1.1 Use Case Scenario #1 Register

Skenario Use Case #1 :

- i. Pre-Condition : User membuka aplikasi
- ii. Use Case Description
  - a. Primary Flow
    - i. user memilih menu register
    - ii. user memasukkan data email, nomor hp, nama, dan password
    - iii. user menekan tombol register
  - b. Alternate Flow
    - i. register gagal karena email tidak valid atau kosong
    - ii. register gagal karena phone tidak valid atau kosong
    - iii. register gagal karena email atau phone sudah terdaftar
    - iv. register gagal karena fullname kosong
    - v. register gagal karena password kosong
- iii. Post-Condition : User diarahkan ke halaman login

### 3.1.1.2 UI Design dan Deskripsi Objek UI #1 Register

**Gambar 1 Register**

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-REG	RegisterPage	Page untuk meminta identitas dan password bagi pengguna baru yang akan mendaftar aplikasi MNote

Page Register

Id_Objek	JENIS	LABEL*	Keterangan**
input-name	InputText	Name	field form untuk user memasukan nama
input-phone	InputText	Phone Number	field form untuk user memasukan PhoneNumber
input-email	InputText	Email	field form untuk user memasukan Email
input-password	InputText	Password	field form untuk user memasukan Password
btn-register	Button	Register	Jika di klik akan mendaftarkan data user
btn-signin	Link	Sign In	Jika diklik akan mengarahkan ke halaman login

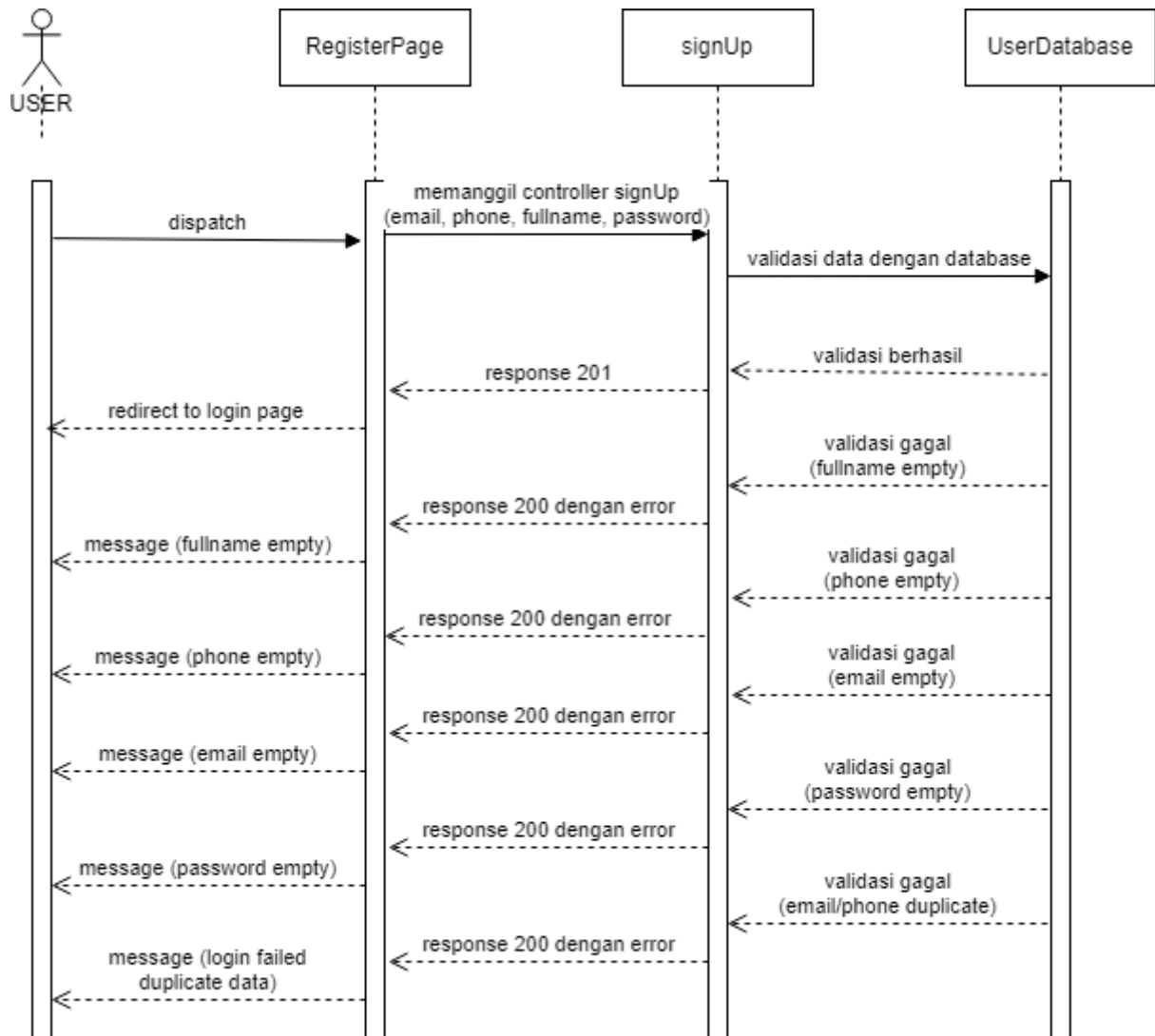
### 3.1.1.3 Identifikasi Object dan Tipe nya #1 Register

#### TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	register-page	Boundary
2	register	Entity
3	phone	Entity
4	email	Entity
5	password	Entity
6	UserController	Controller

\*Tipe kelas diisi dengan **Boundary**(Interface), **Entity**(Database), **Controller**

### 3.1.1.4 Sequence Diagram #1 Register



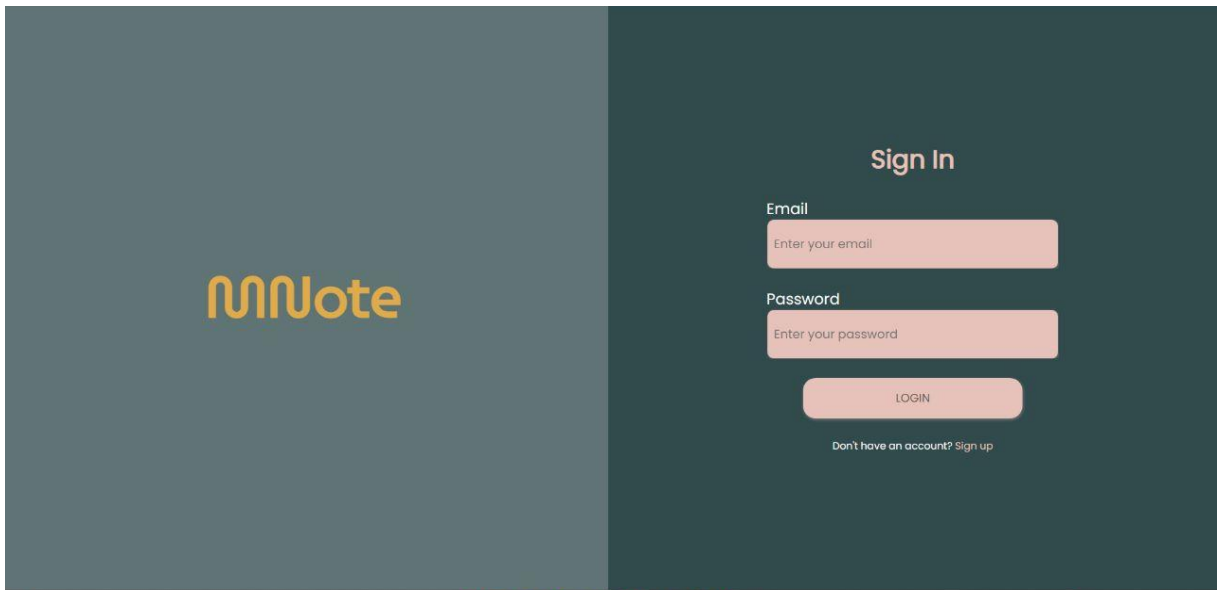
### 3.1.2 Use Case #2 : Login

#### 3.1.2.1 Use Case Scenario #2: Login

Skenario Use Case #2 :

- i. Pre-Condition : User memasuki aplikasi MNote
- ii. Use Case Description
  - a. Primary Flow
    - i. user menginputkan email dan password
    - ii. user menekan tombol login
  - b. Alternative Flow
    - i. error email / password kosong
    - ii. error data user tidak ditemukan
    - iii. error password salah
- iii. Post-Condition : User memasuki halaman dashboard ( login berhasil )

### 3.1.2.2 UI Design dan Deskripsi Objek UI #2: Login



**Gambar 2 Login**

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
<i>MN-Login</i>	<i>LoginPage</i>	<i>Menampilkan halaman Login</i>

Page login

Id_Objek	JENIS	LABEL*	Keterangan**
<i>input-email</i>	<i>InputText</i>	<i>Email</i>	<i>field form untuk user memasukan Email</i>
<i>input-password</i>	<i>InputText</i>	<i>Password</i>	<i>field form untuk user memasukan Password</i>
<i>btn-login</i>	<i>Button</i>	<i>Login</i>	<i>Jika diklik akan login dan redirect ke halaman dashboard</i>
<i>btn-signup</i>	<i>Link</i>	<i>Sign Up</i>	<i>Jiika klik button ini akan menuju halanan register</i>

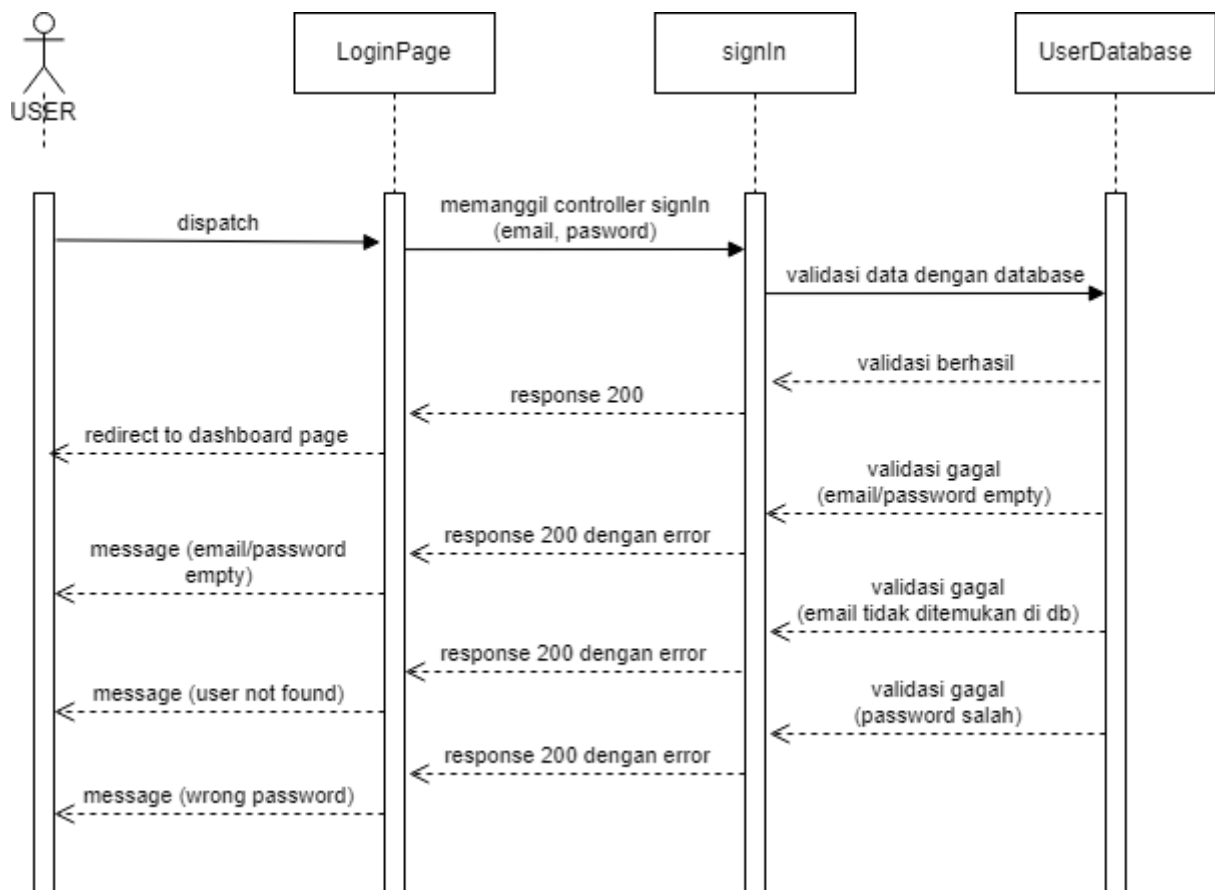
### 3.1.2.3 Identifikasi Object Baru & Tipe Kelas #2 Login

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	<i>AuthController</i>	Controller
2	<i>LoginPage</i>	Boundary
3	<i>User</i>	Entity

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

### 3.1.2.4 Sequence Diagram #2 Login



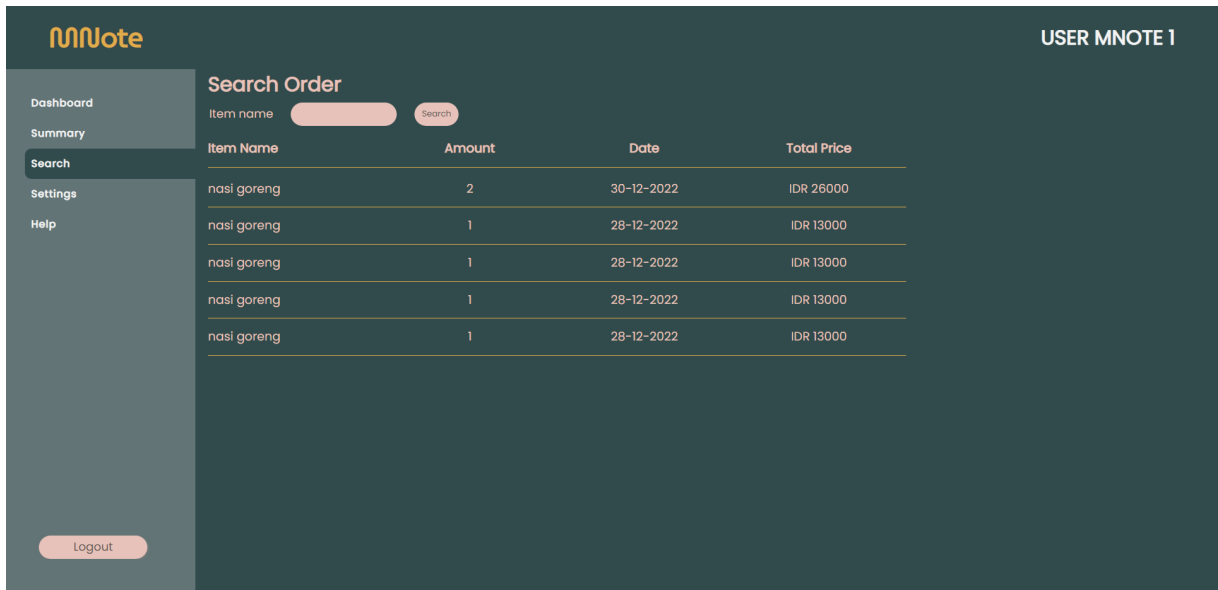
### 3.1.3 Use Case #3 : Search Order

#### 3.1.3.1 Use Case Scenario #3: Search Order

Skenario Use Case #2 :

- iv. Pre-Condition : User memilih menu search
- v. Use Case Description
  - a. Primary Flow
    - i. user memasukkan nama order yang akan dicari
    - ii. user menekan tombol search
    - iii. sistem menampilkan hasil pencarian
  - b. Alternative Flow
    - i. error id user tidak bisa di konversi ke ObjectId yang diperlukan MongoDB
    - ii. error id user tidak terisi atau tidak didapatkan atau berbentuk string
    - iii. error nama user tidak terisi atau tidak didapatkan
    - iv. error data user tidak terdapat atau kosong
    - v. error data order tidak bisa didapatkan
- vi. Post-Condition : Sistem menampilkan data order berdasarkan hasil pencarian pada database

### 3.1.3.2 UI Design dan Deskripsi Objek UI #3: Search Order



Gambar 3 Search Order

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-SR C	SearchPage	Menampilkan halaman search

Page SearchOrder

Id_Objek	JENIS	LABEL*	Keterangan**
input-itemName	InputText	Item name	field form untuk user memasukan nama item
btn-submit	Button	search	button untuk mencari order-order sesuai nama item yang tertulis di input-itemName

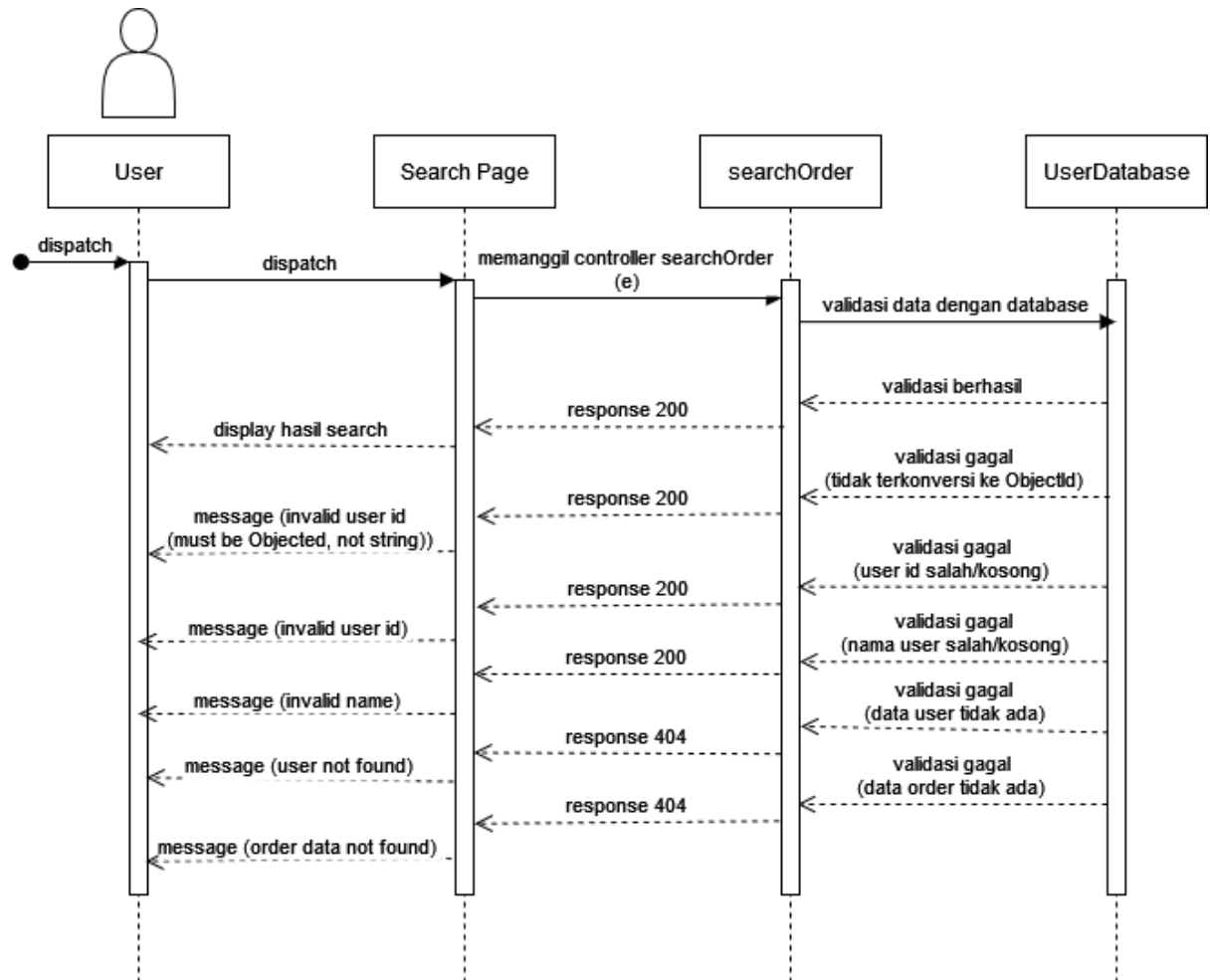
### 3.1.3.3 Identifikasi Object Baru & Tipe Kelas #3 Search Order

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	OrderController	Controller
2	SearchPage	Boundary
3	Order	Entity

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

### 3.1.3.4 Sequence Diagram #3 Search Order



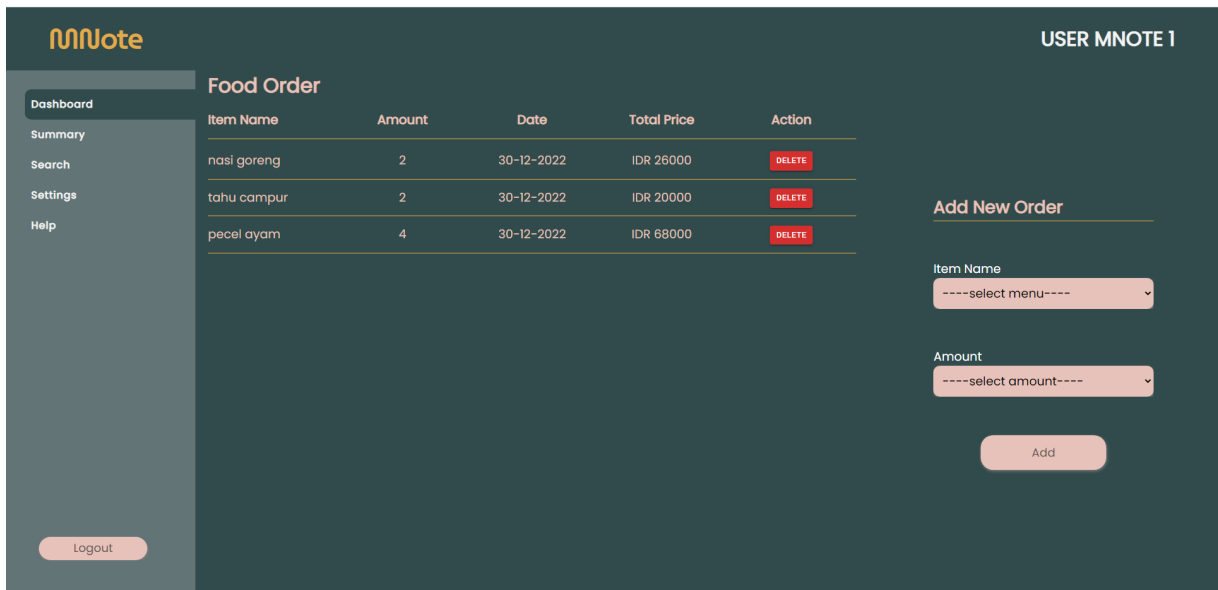
### 3.1.4 Use Case #4 :Dashboard

#### 3.1.4.1 Use Case Scenario #4: Dashboard

Skenario Use Case #2 :

- vii. Pre-Condition : User memasuki menu dashboard
- viii. Use Case Description
  - a. Primary Flow
    - i. user melakukan login dan berhasil
    - ii. user memasuki menu dashboard
    - iii. sistem menampilkan data order
  - b. Alternative Flow
    - i. Sistem menampilkan data kosong karena data order dengan id user tertentu tidak ditemukan
    - ii. Sistem menampilkan data kosong karena user id invalid
- ix. Post-Condition : Sistem menampilkan data order

### 3.1.4.2 UI Design dan Deskripsi Objek UI #4: *Dashboard*



**Gambar 4 Dashboard**

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-DSH	DashboardPage	Halaman dashboard yang menampilkan data order

Page Dashboard

Id_Objek	JENIS	LABEL*	Keterangan**
btn-delete	Button	Delete	Button untuk delete order
input-name	Input	item name	Input nama item order
input-amount	Input	amount	input jumlah item order
btn-add	Button	Add	Button tambah data order
btn-logout	Button	Logout	Button untuk logout akun

### 3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 Dashboard

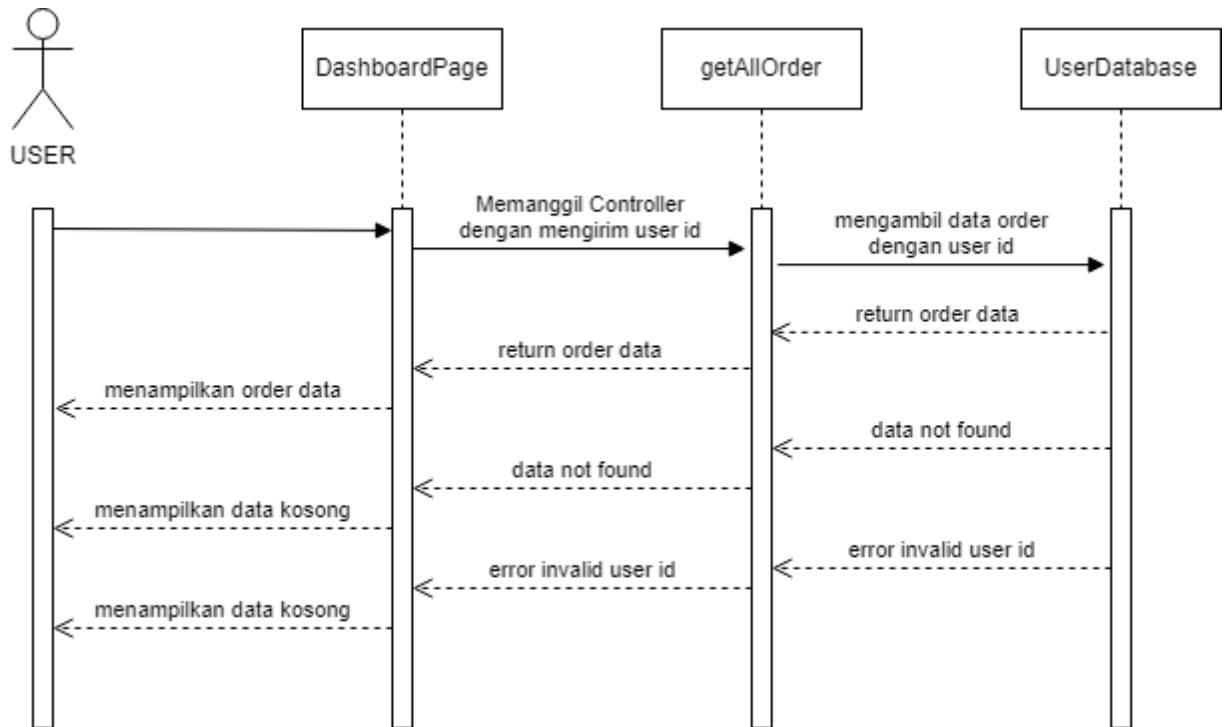
TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Dashboard Page	Boundary
2	Order	Entity
3	OrderController	Controller

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller



### 3.1.4.4 Sequence Diagram #4 Dashboard



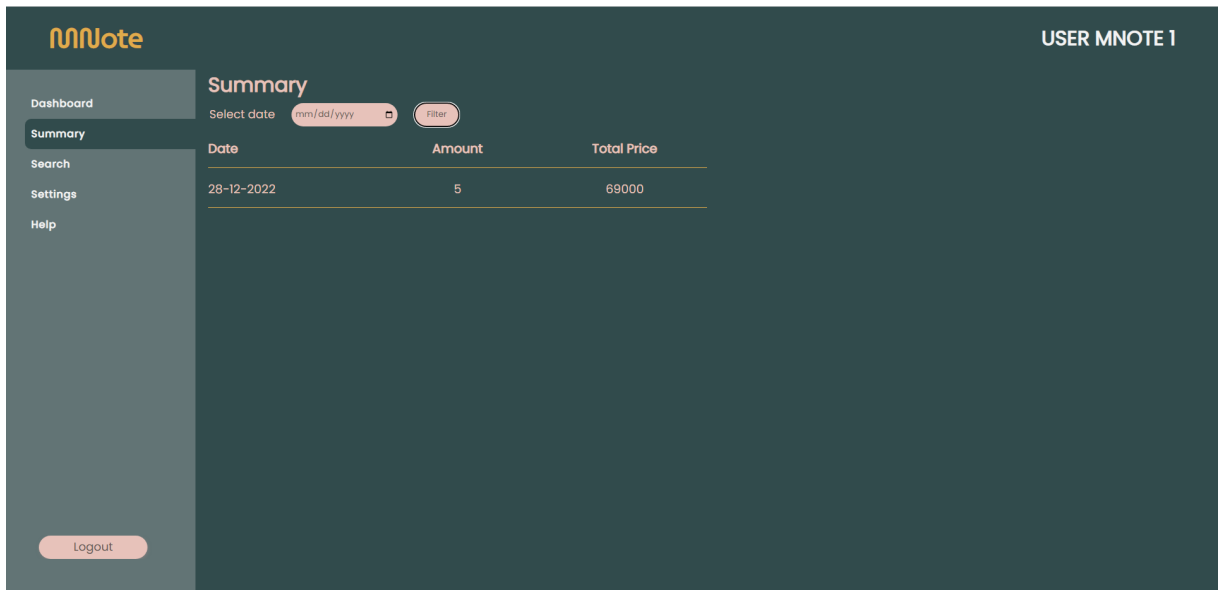
### 3.1.5 Use Case #5 : Summary

#### 3.1.5.1 Use Case Scenario #5 : Summary

Skenario Use Case #2 :

- x. Pre-Condition : User memasuki menu summary
- xi. Use Case Description
  - a. Primary Flow
    - i. user memilih date berupa tanggal, bulan, dan tahun
    - ii. user menekan tombol filter
    - iii. sistem menampilkan data order
  - b. Alternative Flow
    - i. aplikasi menampilkan daftar kosong karena data dengan date yang dicari tidak ditemukan
    - ii. aplikasi menampilkan data kosong karena id dari user yang login tidak ditemukan
    - iii. aplikasi menampilkan data kosong karena format date yang dimasukkan tidak valid
    - iv. aplikasi menampilkan data kosong karena id dari user yang login tidak valid
    - v. aplikasi menampilkan data kosong karena tipe dari id user yang login tidak valid
- xii. Post-Condition : Sistem menampilkan data order berdasarkan tanggal, bulan, dan tahun

### 3.1.5.2 UI Design dan Deskripsi Objek UI #5: *Summary*



Gambar 5 Summary

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-SUM	Summary	Laman di mana menampilkan jumlah pesanan dan total harga pada tanggal yang ditentukan

Page Summary

Id_Objek	JENIS	LABEL*	Keterangan**
input-date	Input	Date	Input tanggal untuk filter data order
btn-filter	Button	Filter	Jika di klik akan menampilkan result jumlah pesanan dan total harga pada hari yang ditentukan

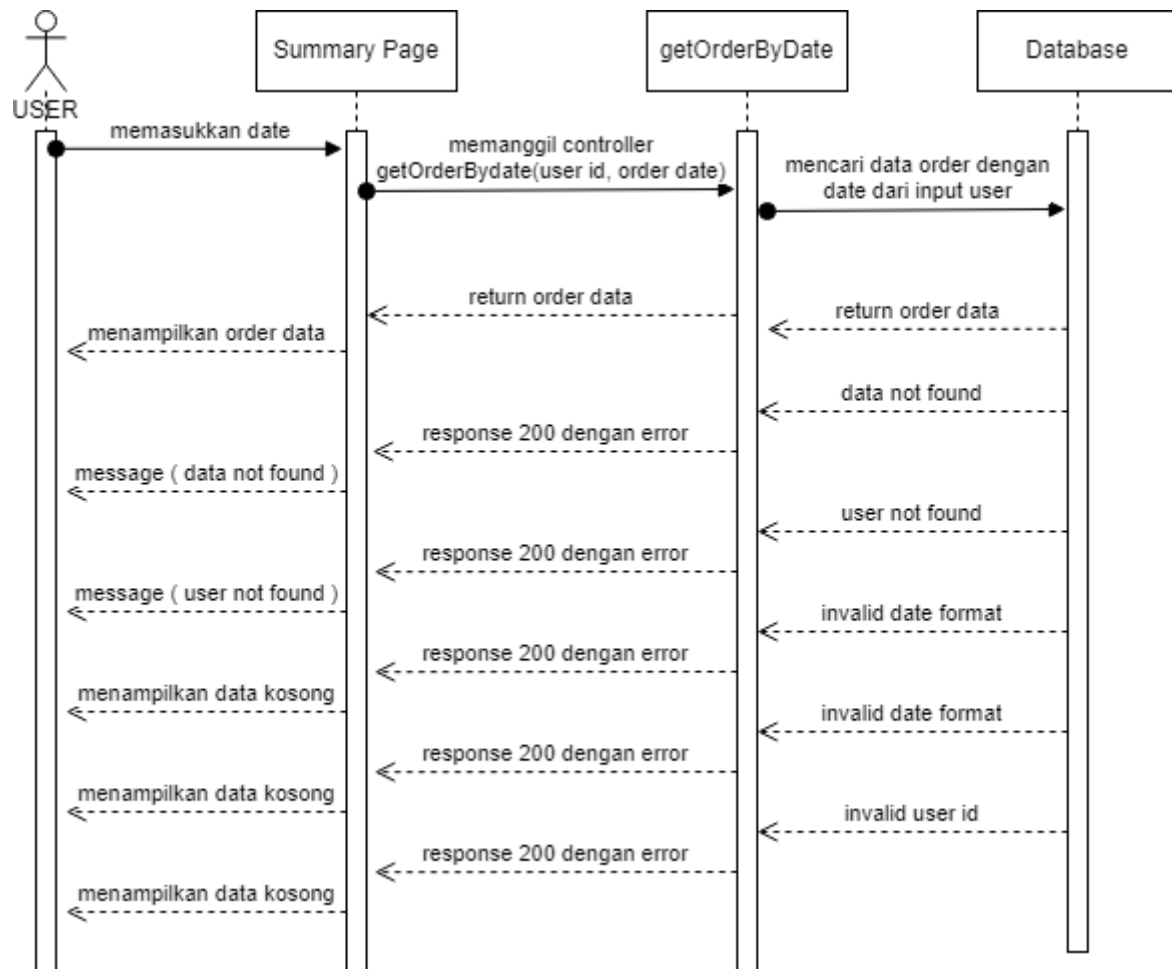
### 3.1.5.3 Identifikasi Object Baru & Tipe Kelas #5 : *Summary*

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	SummaryPage	Boundary
2	Order	Entity
3	OrderController	Controller

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

### 3.1.5.4 Sequence Diagram #5 : Summary



### 3.1.6 Use Case #6 :Setting

#### 3.1.6.1 Use Case Scenario #6 :Setting

Skenario Use Case #2 :

xiii. Pre-Condition : User memasuki halaman dashboard

xiv. Use Case Description

a. Primary Flow

- i. user memilih menu settings
- ii. user memasukkan data baru berupa nama, phone, email, atau password
- iii. user menekan tombol update

b. Alternative Flow

- i. update gagal karena email kosong
- ii. update gagal karena phone kosong
- iii. update gagal karena fullname kosong
- iv. update gagal karena password kosong

xv. Post-Condition : Sistem memperbarui data user berdasarkan data terbaru

### 3.1.6.2 UI Design dan Deskripsi Objek UI #6 :Setting

Gambar 6 Settings

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-AC S	Account setting	Menampilkan halaman account setting

Page Settings

Id_Objek	JENIS	LABEL*	Keterangan**
<i>input-fullname</i>	<i>Input</i>	<i>Fullname</i>	Field form untuk username sebagai data pengguna
<i>input-email</i>	<i>Input</i>	<i>Email</i>	Field form untuk email sebagai data pengguna
<i>input-phone number</i>	<i>Input</i>	<i>PhoneNumber</i>	Field form untuk phone number sebagai data pengguna
<i>input-password</i>	<i>Input</i>	<i>Password</i>	Field form untuk password sebagai data pengguna
<i>btn-Update</i>	Button	<i>Update</i>	Jika di klik akan mengupdate account setting pengguna

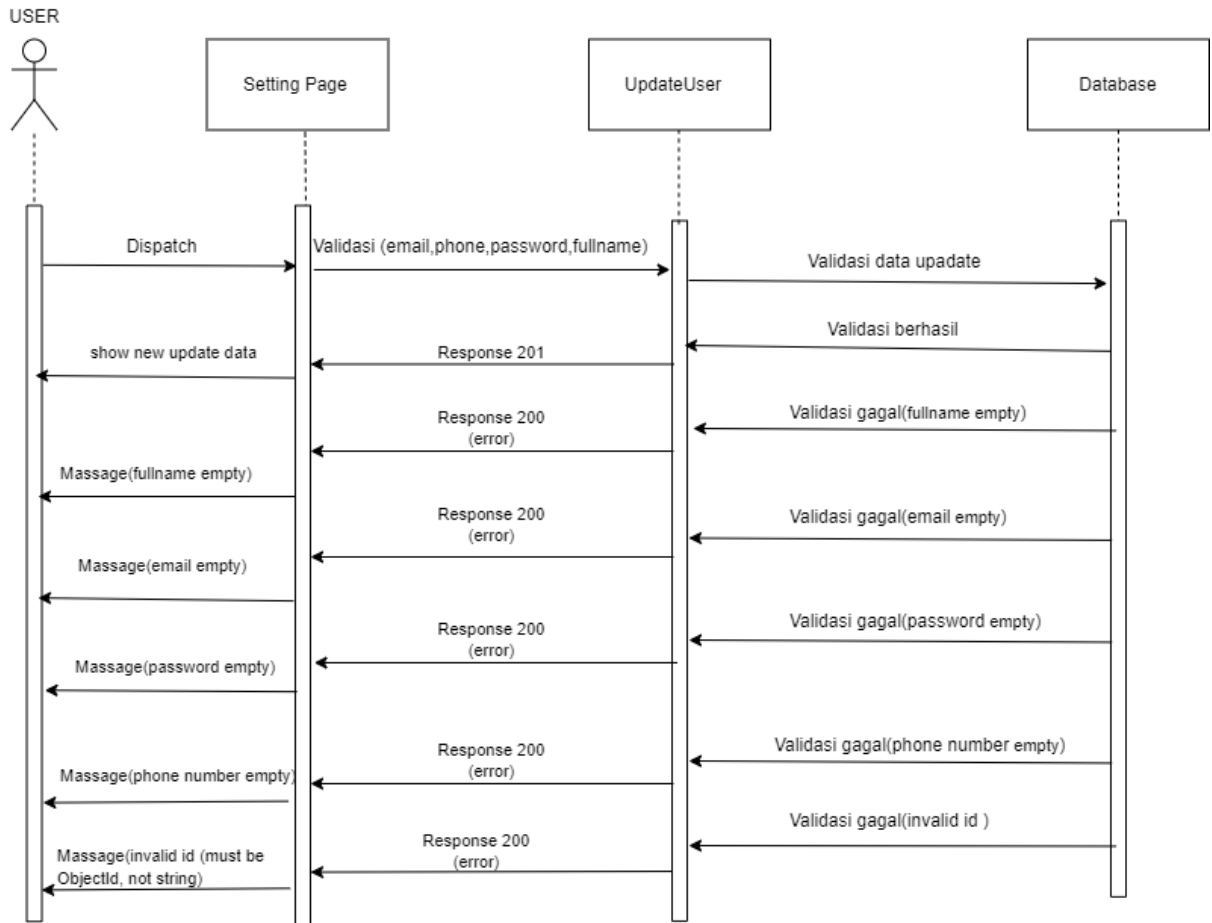
### 3.1.6.3 Identifikasi Object Baru & Tipe Kelas #6 :Setting

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	<i>Setting page</i>	<i>Boundary</i>
2	<i>Database</i>	<i>Entity</i>
3	<i>UserController</i>	<i>Controller</i>

\*Tipe kelas seperti *Boundary*(Interface), *Entity*(Database), *Controller*

### 3.1.6.4 Sequence Diagram #6 :Setting



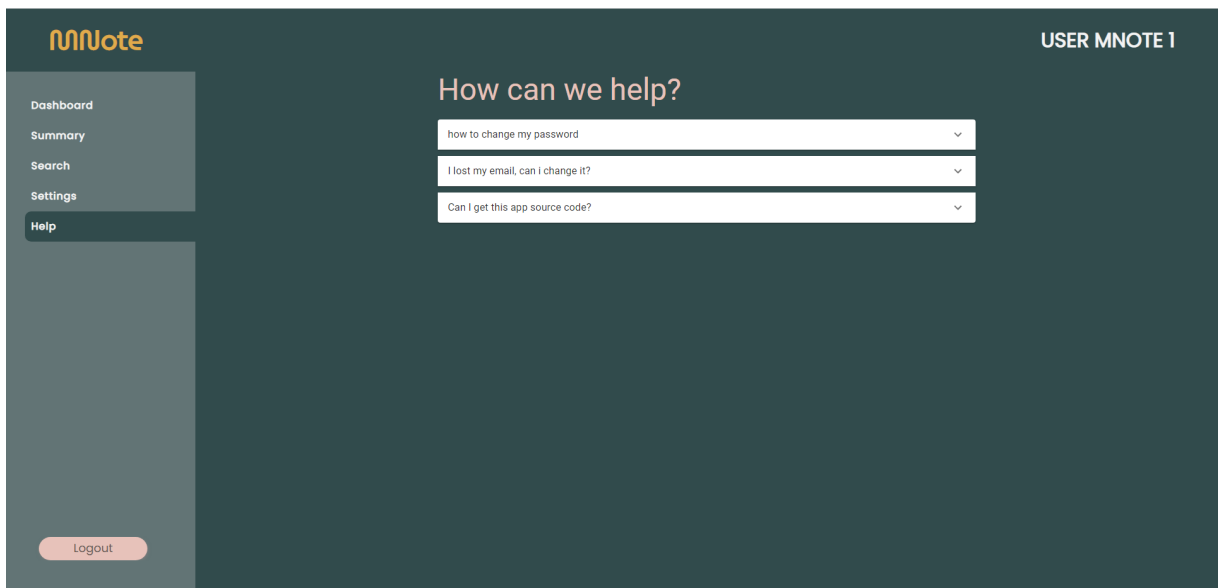
### 3.1.7 Use Case #7 : Help

#### 3.1.7.1 Use Case Scenario #7 : Help

Skenario Use Case #2 :

- xvi. Pre-Condition : Sistem menampilkan beberapa topik
- xvii. Use Case Description
  - a. Primary Flow
    - i. user memilih topik yang tersedia
    - ii. sistem menampilkan informasi atau jawaban berdasarkan topik pilihan user
  - b. Alternative Flow
    - i. sistem menampilkan message: data not found dikarenakan tidak terdapat data pada HelpDatabase
- xviii. Post-Condition : Sistem menampilkan informasi sesuai topik yang dipilih user

### 3.1.7.2 UI Design dan Deskripsi Objek UI #7 : Help



Gambar 7 Help

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-Help	HelpPage	Laman dimana user dapat menemukan informasi bantuan berdasarkan topik yang telah dipilih user.

Page Help

Id_Objek	JENIS	LABEL*	Keterangan**
btn-accordion	Button	v	Jika diklik, akan mengexpand topik yang dipilih dan menampilkan jawaban

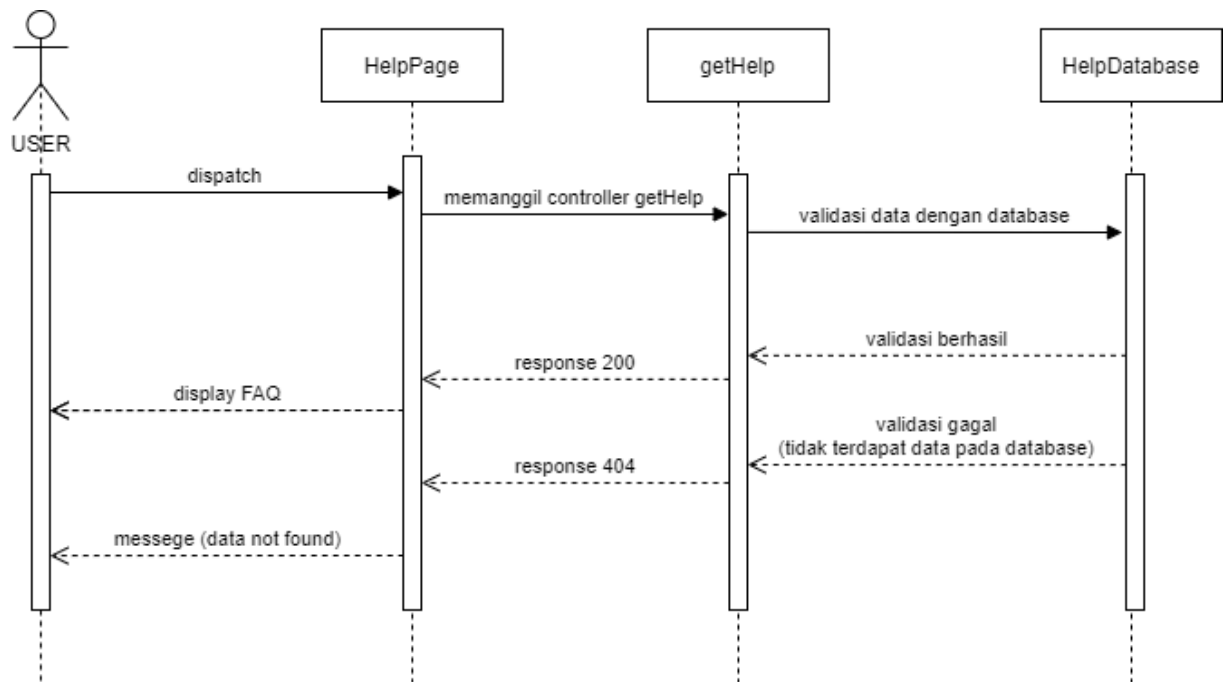
### 3.1.7.3 Identifikasi Object Baru & Tipe Kelas #7 : Help

TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	HelpPage	Boundary
2	Help	Entity
3	HelpController	Controller

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

#### 3.1.7.4 Sequence Diagram #7 : Help



#### 3.1.8 Use Case #8 : Add Order

##### 3.1.8.1 Use Case Scenario #8 : Add Order

Skenario Use Case #2 :

xix. Pre-Condition : Sistem menyediakan form untuk menambah data order

xx. Use Case Description

a. Primary Flow

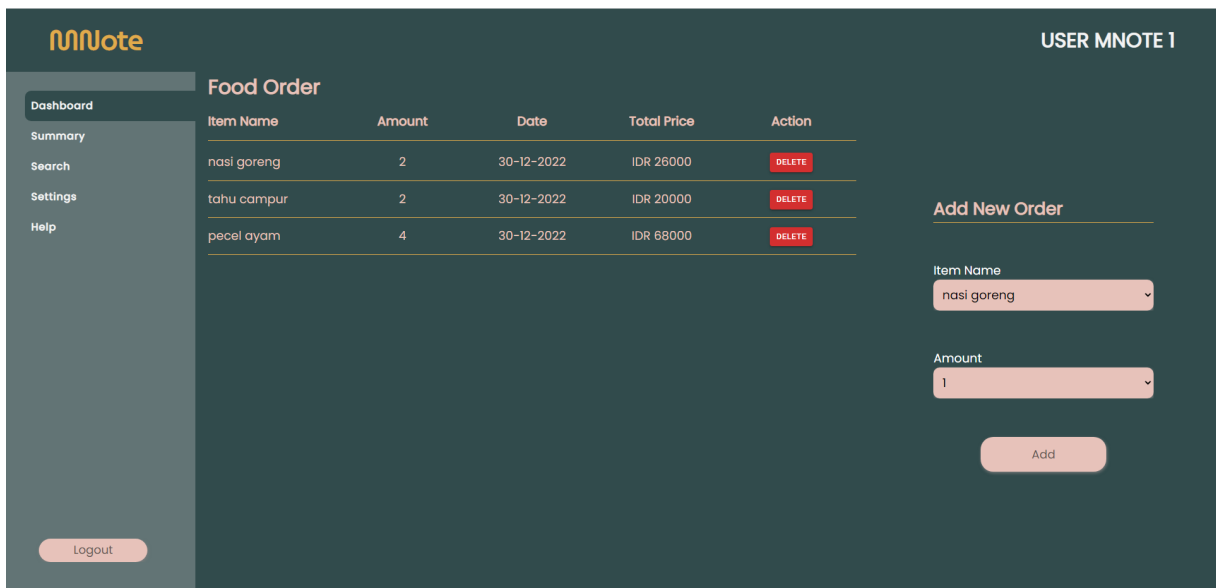
- i. user memasukkan data order baru berupa nama item, dan amount
- ii. user menekan tombol add order
- iii. Sistem menyimpan data order baru ke database

b. Alternative Flow

- i. error name kosong
- ii. error amount kosong
- iii. error userId kosong
- iv. error userId tidak valid

xxi. Post-Condition : Sistem menampilkan data baru order ke dashboard

### 3.1.8.2 UI Design dan Deskripsi Objek UI #8 : Add Order



Gambar 8 Add Order

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-DSH	DashboardPage	Halaman dashboard yang menampilkan data order

Page AddOrder

Id_Objek	JENIS	LABEL*	Keterangan**
input-itemName	Input	Item name	Input field untuk nama item
input-amount	Input	Amount	Input field untuk jumlah order
btn-add	Button	Add	Button add order yang akan menambahkan data baru order ke database dengan mengakses api addOrder

### 3.1.8.3 Identifikasi Object Baru & Tipe Kelas #8 : Add Order

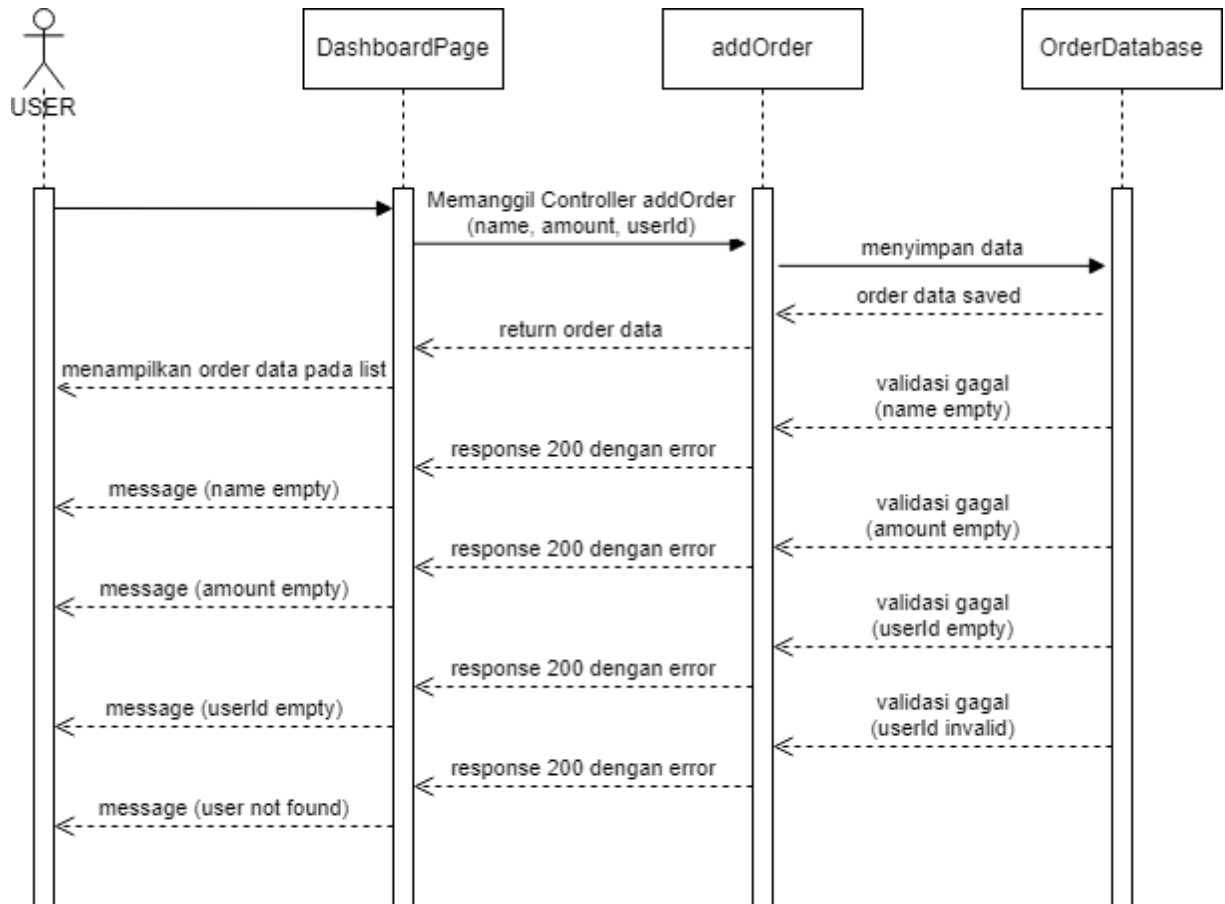
TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
1	Order	Entity
2	DashboardPage	Boundary
3	OrderController	Controller

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller



### 3.1.8.4 Sequence Diagram #8 : Add Order



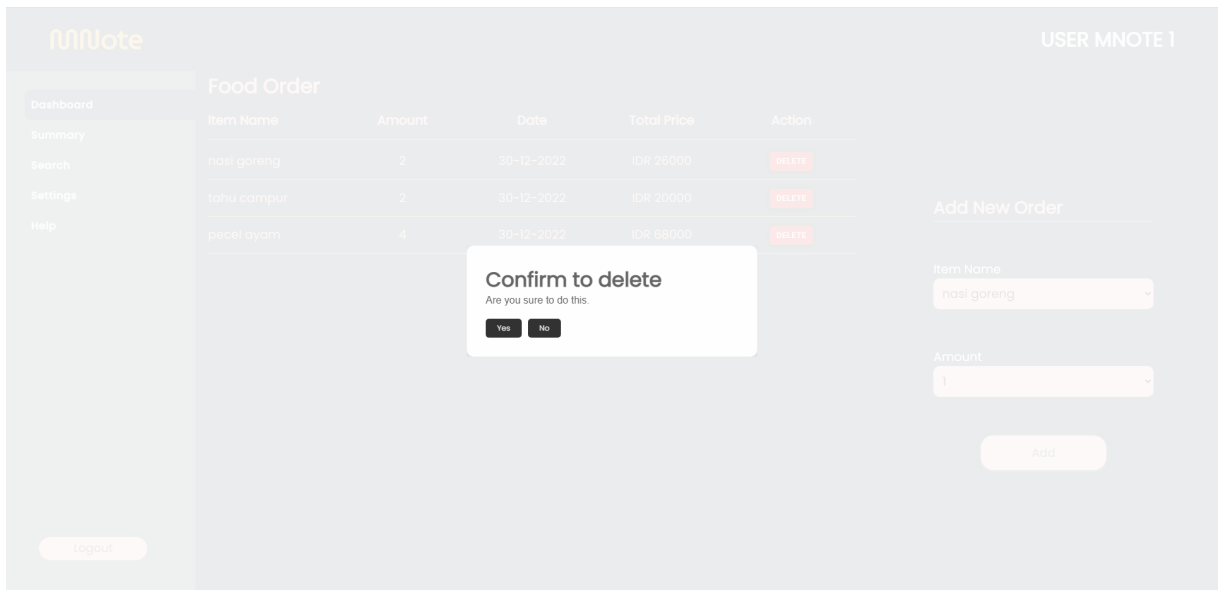
### 3.1.9 Use Case #9 : Delete Order

#### 3.1.9.1 Use Case Scenario #9 : Delete Order

Skenario Use Case #2 :

- xxii. Pre-Condition : User memilih catatan order yang akan dihapus dan system menampilkan opsi untuk *delete catatan order*
- xxiii. Use Case Description
  - a. Primary Flow
    - i. sistem menampilkan tombol delete
    - ii. user menekan tombol delete
    - iii. sistem menampilkan konfirmasi delete
    - iv. user menekan tombol konfirmasi
    - v. sistem menghapus data pada database
  - b. Alternative Flow
    - i. error orderId tidak valid
    - ii. error orderId kosong
    - iii. error tidak ada data order dengan order id yang dipilih
- xxiv. Post-Condition : Data catatan order yang telah dihapus terhapus dalam database dan tidak ditampilkan pada list *food order* pada menu *dashboard*

### 3.1.9.2 UI Design dan Deskripsi Objek UI #9 : Delete Order



Gambar 9 Delete Order

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
MN-DS H	DashboardPage	Halaman dashboard yang menampilkan data order

Page deleteOrder

Id_Objek	JENIS	LABEL*	Keterangan**
btn-delete	Button	Delete	button ditekan akan menghapus data order dengan mengakses api deleteOrder
div-row	Div	Row	row akan mengirim data berupa id order ke api deleteOrder
btn-yes	Button	Yes	Konfirmasi hapus order data
btn-no	Button	No	Konfirmasi batal hapus order data

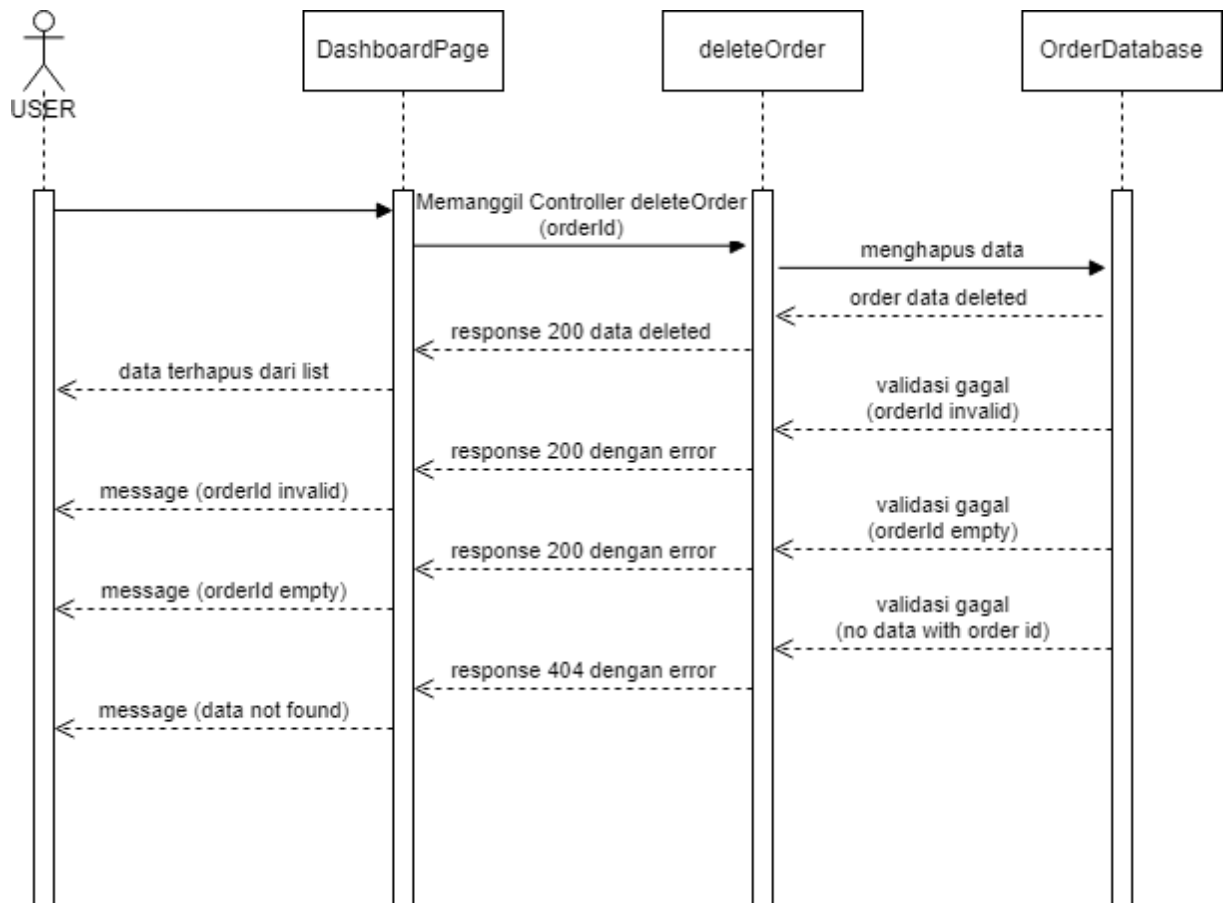
### 3.1.9.3 Identifikasi Object Baru & Tipe Kelas #9 : Delete Order

TABEL OBJECT PERANCANGAN

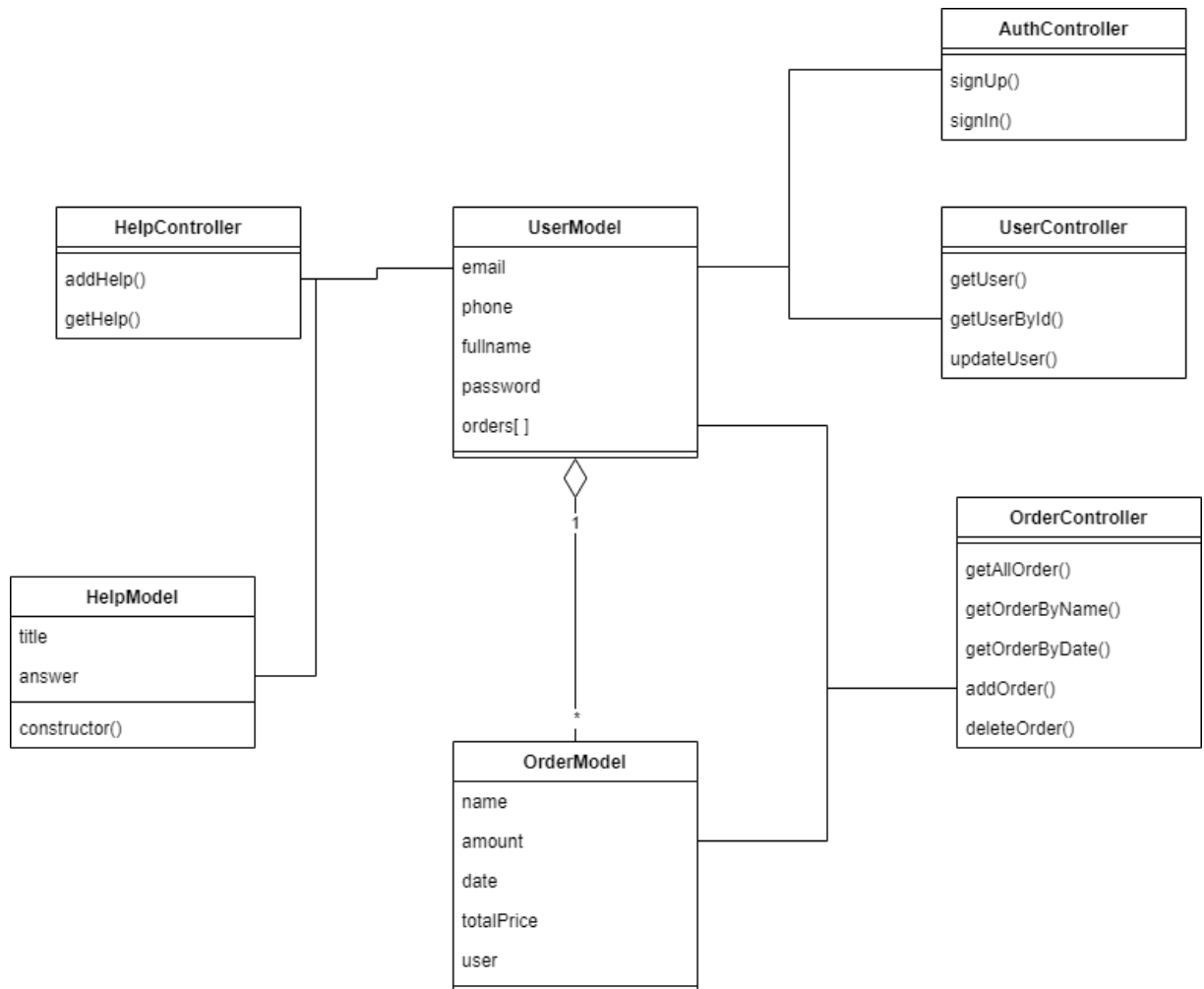
No	Nama Object Baru	Jenis / Tipe Kelas
1	Order	Entity
2	DashboardPage	Boundary
3	OrderController	Controller

\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller

### 3.1.9.4 Sequence Diagram #9 : Delete Order



### 3.2 Diagram Kelas Keseluruhan



**TABEL KELAS :**

ID Kelas	Nama Kelas Perancangan	Atribute (visibility)	Method / Operation
	AuthController	public	signUp, signIn
	UserController	public	getUser, getUserById, updateUser
	OrderController	public	getAllOrder, getOrderByName, getOrderById, addOrder, deleteOrder
	HelpController	public	addHelp, getHelp
	UserModel	public	-
	OrderModel	public	-
	HelpModel	public	-

### 3.3 Perancangan Data / Basis Data

User	
PK	<u>id String NOT NULL</u>
	email String NOT NULL
	phone String NOT NULL
	password String NOT NULL
	fullname String NOT NULL

Order	
PK	<u>id String NOT NULL</u>
	name String NOT NULL
	amount Number NOT NULL
	date String NOT NULL
	totalPrice Number NOT NULL

Help	
PK	<u>id String NOT NULL</u>
	title String NOT NULL
	answer String NOT NULL

### 3.4 Perancangan Algoritma dan/atau Query

Bagian ini hanya diisi untuk kerangka algoritma untuk proses-proses yang dianggap cukup penting. Implementasi skeleton code juga sudah dapat dilakukan untuk kelas-kelas yang terdefinisi pada bahasa pemrograman tertentu. Boleh dibuat subbab per kelas.

#### 3.4.1 Algoritma/Query Auth

Nama Kelas : AuthController  
Nama Operasi : Register akun user  
Algoritma :

```
input(email, phone, fullname, password)
```

```
x : user
```

```
x : await new user.model({email: email, phone: phone, fullname: fullname.toUpperCase(), password: password})  
await x.save()
```

Query pada algoritma tersebut digunakan untuk membuat sebuah data user baru pada database MongoDB dengan data berupa email, phone, fullname, dan password yang dihash.

#### 3.4.2 Algoritma/Query Order

Nama Kelas : OrderController  
Nama Operasi : Menambah data order  
Algoritma :

```
input(name, amount, userId)
```

```
x : order
```

```
x : await new order.model({name: name, amount: amount, date: new Date(), totalPrice: totalPrice *  
amount, user: userId})  
await x.save()
```

```
y: user
```

```
y : await user.model.findOneAndUpdate({userId}, {$push: {orders: x._id}})
```

```
await y.save()
```

Query pada algoritma tersebut digunakan untuk membuat sebuah data order baru dan id dari data tersebut kemudian disimpan pada list orders pada data user

### 3.4.3 Algoritma/Query Help

Nama Kelas : HelpController

Nama Operasi : Melihat data FAQ / Help

Algoritma :

```
x : help
```

```
x : help.model.find()
```

```
return x
```

### 3.4.4 Algoritma/Query User

Nama Kelas : UserController

Nama Operasi : Mencari data user dengan id tertentu

Algoritma :

```
input(userId)
```

```
x : user
```

```
x : user.model.findOne({_id: userId})
```

```
return x
```

## 4. Matriks Kerunutan (Requirement Traceability Matrix)

Mapping Requirement dengan Use Case yang direalisasikan

Kode FR	Nama Functional Requirement	Nama Use Case
FR1	Register	Register
FR2	Login	Login
FR3	Search Order	Search Order
FR4	Dashboard	Dashboard
FR5	Summary	Summary
FR6	Settings	Setings
FR7	Help	Help

FR8	Add Order	Add Order
FR9	Delete Order	Delete Order