

Final Project of Backend Developer Bootcamp

Eda Özyurt





Contents

1	. Pur	pose of Project	. 3
		Project Summary	
	1.2	Technologies	. 3
	1.3	Requirements of Project	. 3
2	. Serv	rice Documentation and JSON Examples for API Endpoints	. 4
	2.1	Endpoints	. 4
	2.2 DT	Os (Data Transfer Objects)	. 5
	2.2	JSON Examples	. 7

1. Purpose of Project

Project Name: Airline Ticket Booking System

1.1 Project Summary

The "Airline Ticket Booking System Backend Service Application" project addresses the needs of modern airline operations by providing a comprehensive backend solution.

1.2 Technologies

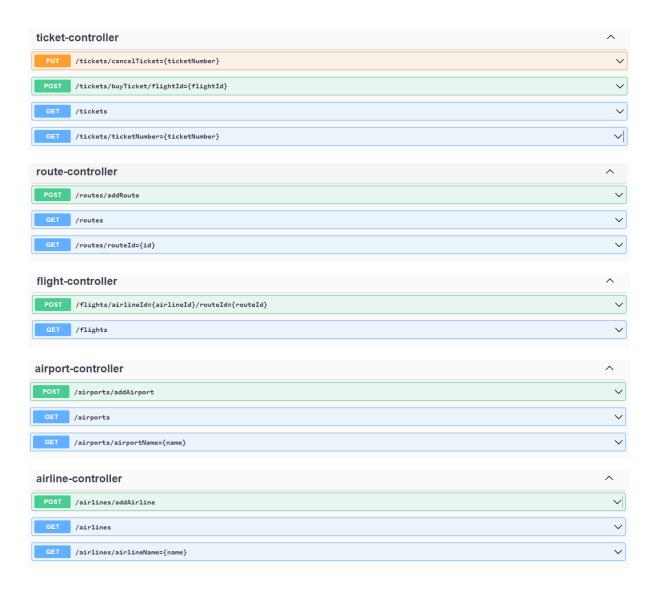
Technology	Version
Java	17
Spring Boot	3
MySQL	
Database Restful Web Service (JSON)	
modelmapper	3.1.1
openapi	2.1.0
lombok	
IntelliJ IDEA	

1.3 Requirements of Project

- Streamlined Operations: The project aims to facilitate efficient airline management through functionalities like adding, searching, and defining airline companies, airports, and flight routes. This ensures smoother operations for both airlines and passengers.
- Secure Ticket Purchasing: With a strong emphasis on security, the project implements a secure ticket purchasing process. Sensitive credit card information is masked during the transaction, accommodating various card number formats to ensure a secure and hassle-free experience.
- Enhanced Ticket Management: The system enables users to easily search for tickets using ticket numbers and provides the capability to cancel tickets.
- Error Management and Consistent Responses: The project establishes a robust global exception handling structure to handle errors effectively. API responses are structured uniformly, providing clarity and consistency to users interacting with the system.

2. Service Documentation and JSON Examples for API Endpoints

2.1 Endpoints



2.2 DTOs (Data Transfer Objects)

```
TicketRequestDTO > {
    price
    price
    passengerName string
    seatNumber string
    c
    string
}
 AirlineResponseDTO • {
    name string country string }
price number($double)
passengerName string
cc string
cc string
 FlightResponseDTO → {
    route RouteResponseDTO V {
    departureAirport string arrivalAirport string
}
    departureAirport arrivalAirport string arrivalAirport string

flightNumber airline String AirlineResponseDTO 

name country string string string arrivalTime string capacity integer($int32)
 RouteResponseDTO > {
    departureAirport string arrivalAirport string
```

```
AirlineRequestDTO V {
    airlineId integer($int64)
    name string
    country string
RouteResponseDTO 

departureAirport string
arrivalAirport string
}
string
                                       flightNumber
airline
                                                                string

AirlineResponseDTO 

name string country string 
} string 
string 
integer(Sint32)
```

departureTime arrivalTime capacity }
price | number(\$double) |
passengerName | string |
seatNumber | string |

2.2 JSON Examples

Add Airpot

```
Parameters

Request body required

Application/json

Tisscholae Havalinana**,

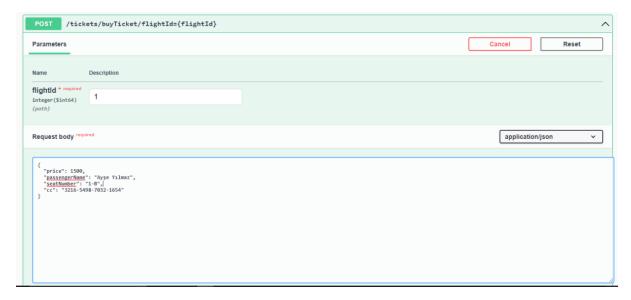
"city: "Ankara*,
"country*: "Illerkive!"

Execute
```

Response Body – Add Airport

Get All Airports

Buy Ticket



Response Body – Buy Ticket

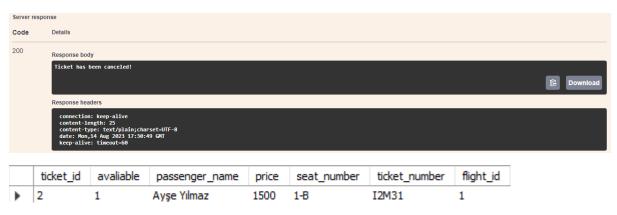
Buy Ticket For Unavaliable Seat with Same Seat Number



Cancel Ticket – Request



Cancel Ticekt – Response Body



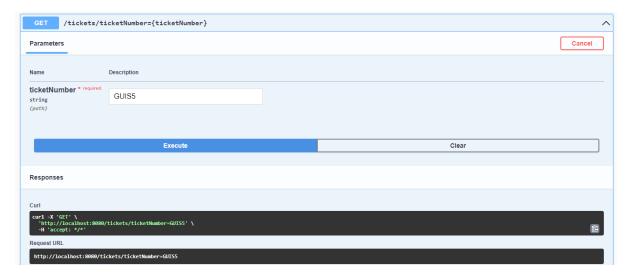
Get All Tickets

```
Code Details

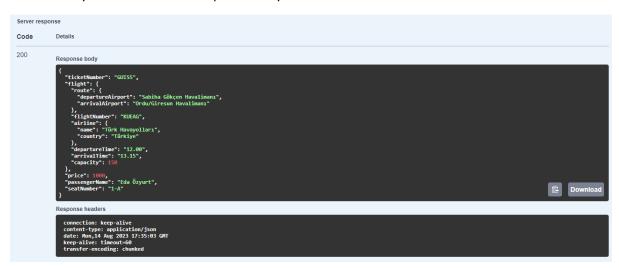
Response body

[
{
    "ticketNumber": "GUISS",
    "light": {
        "rower": "Fight ": "Sabiha Gökçen Havalimana",
        "arrivalAirport": "Sabiha Gökçen Havalimana",
        "arrivalAirport": "GUAG",
        "ainline": {
        "name": Türk Havayolları",
        "country": "Türkiye"
        },
        "departureline": 12.00",
        "arrivalAirport": "33.15",
        "capacity": 130.0,
        "passengerhame": "Eda Özyurt",
        "scathumber": "1.A"
        },
        "ticketHumber": "12/81",
        "fight": "Gody Gökçen Havalimana",
        "arrivalAirport": "Sabiha Gökçen Havalimana",
        "arrivalAirport": "Ordu/Giresun Havaliman
```

Get Ticket By Ticket Number – Request



Get Ticket By Ticket Number – Response Body



Get Ticket By Ticket Number with Unexisting Ticket Number

