

Flight Reservation System

FlightReservationSystemApplication

```
package com.example.FlightReservationSystem;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class FlightReservationSystemApplication {

    public static void main(String[] args) {

        SpringApplication.run(FlightReservationSystemApplication.class, args);

    }

}
```

Flight class

```
package com.example.FlightReservationSystem.entity;

import java.time.LocalDateTime;
import java.util.ArrayList;
import java.util.List;
import jakarta.persistence.CascadeType;
import jakarta.persistence.Column;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.OneToOne;
```

@Entity

```
public class Flight {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    @Column(unique = true)
```

```
    private String flightNumber;
```

```
    private String origin;
```

```
    private String destination;
```

```
    private LocalDateTime departureTime;
```

```
    private int seatsAvailable;
```

```
    @OneToMany(mappedBy = "flight", cascade = CascadeType.ALL, orphanRemoval = true)
```

```
    private List<Reservation> reservations = new ArrayList<>();
```

```
        public Long getId() {
```

```
            return id;
```

```
        }
```

```
        public void setId(Long id) {
```

```
            this.id = id;
```

```
        }
```

```
        public String getFlightNumber() {
```

```
            return flightNumber;
```

```
        }
```

```
        public void setFlightNumber(String flightNumber) {
```

```
            this.flightNumber = flightNumber;
```

```
        }
```

```
        public String getOrigin() {
```

```
        return origin;
    }

    public void setOrigin(String origin) {
        this.origin = origin;
    }

    public String getDestination() {
        return destination;
    }

    public void setDestination(String destination) {
        this.destination = destination;
    }

    public LocalDateTime getDepartureTime() {
        return departureTime;
    }

    public void setDepartureTime(LocalDateTime departureTime) {
        this.departureTime = departureTime;
    }

    public int getSeatsAvailable() {
        return seatsAvailable;
    }

    public void setSeatsAvailable(int seatsAvailable) {
        this.seatsAvailable = seatsAvailable;
    }

    public List<Reservation> getReservations() {
        return reservations;
    }

    public void setReservations(List<Reservation> reservations) {
```

```
        this.reservations = reservations;
    }
}
```

Reservation class

```
package com.example.FlightReservationSystem.entity;

import java.time.LocalDateTime;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.JoinColumn;
import jakarta.persistence.ManyToOne;

@Entity
public class Reservation {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String passengerName;

    private String passengerEmail;

    private int seatsBooked;

    private LocalDateTime reservedAt;

    @ManyToOne
    @JoinColumn(name = "flight_id")
    private Flight flight;

    public Long getId() {
        return id;
    }
}
```

```
public void setId(Long id) {  
    this.id = id;  
}  
  
public String getPassengerName() {  
    return passengerName;  
}  
  
public void setPassengerName(String passengerName) {  
    this.passengerName = passengerName;  
}  
  
public String getPassengerEmail() {  
    return passengerEmail;  
}  
  
public void setPassengerEmail(String passengerEmail) {  
    this.passengerEmail = passengerEmail;  
}  
  
public int getSeatsBooked() {  
    return seatsBooked;  
}  
  
public void setSeatsBooked(int seatsBooked) {  
    this.seatsBooked = seatsBooked;  
}  
  
public LocalDateTime getReservedAt() {  
    return reservedAt;  
}  
  
public void setReservedAt(LocalDateTime reservedAt) {  
    this.reservedAt = reservedAt;  
}
```

```

        public Flight getFlight() {
            return flight;
        }

        public void setFlight(Flight flight) {
            this.flight = flight;
        }
    }
}

```

FlightController class

```

package com.example.FlightReservationSystem.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.PutMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.example.FlightReservationSystem.entity.Flight;
import com.example.FlightReservationSystem.service.FlightService;

@RestController
@RequestMapping("/api/flights")

public class FlightController {

    @Autowired private FlightService flightService;

    @PostMapping public ResponseEntity<Flight> add(@RequestBody Flight flight) {

```

```

        return ResponseEntity.ok(flightService.addFlight(flight)); }

@GetMapping
public List<Flight> all() {
    return flightService.getAllFlights();
}

@GetMapping("/{id}")
public Flight get(@PathVariable Long id) {
    return flightService.getFlight(id);
}

@PutMapping("/{id}")
public Flight update(@PathVariable Long id, @RequestBody Flight f) {
    return flightService.updateFlight(id, f);
}

@DeleteMapping("/{id}")
public void delete(@PathVariable Long id) {
    flightService.deleteFlight(id);
}
}

```

ReservationController class

```

package com.example.FlightReservationSystem.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;

```

```
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.RestController;
import com.example.FlightReservationSystem.entity.Reservation;
import com.example.FlightReservationSystem.service.ReservationService;

@RestController

@RequestMapping("/api/reservations")

public class ReservationController {

    @Autowired private ReservationService resService;

    @PostMapping

    public Reservation reserve(@RequestParam Long flightId, @RequestBody Reservation
res) {

        return resService.makeReservation(flightId, res);

    }

    @GetMapping

    public List<Reservation> all() {

        return resService.getAllReservations();

    }

    @GetMapping("/flight/{flightId}")

    public List<Reservation> byFlight(@PathVariable Long flightId) {

        return resService.getReservationsByFlight(flightId);

    }

    @DeleteMapping("/{id}")

    public void cancel(@PathVariable Long id) {

        resService.cancelReservation(id);

    }

}
```


FlightRepository interface

```
package com.example.FlightReservationSystem.repository;

import java.util.Optional;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.example.FlightReservationSystem.entity.Flight;

@Repository

public interface FlightRepository extends JpaRepository<Flight, Long> {

    Optional<Flight> findByFlightNumber(String flightNumber);

}
```

ReservationRepository interface

```
package com.example.FlightReservationSystem.repository;

import java.util.List;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.example.FlightReservationSystem.entity.Reservation;

@Repository

public interface ReservationRepository extends JpaRepository<Reservation, Long> {

    List<Reservation> findByFlightId(Long flightId);

}
```

FlightService class

```
package com.example.FlightReservationSystem.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.example.FlightReservationSystem.entity.Flight;
```

```
import com.example.FlightReservationSystem.repository.FlightRepository;

@Service

public class FlightService {

    @Autowired

    private FlightRepository flightRepo;

    public Flight addFlight(Flight flight) {

        return flightRepo.save(flight);

    }

    public List<Flight> getAllFlights() {

        return flightRepo.findAll();

    }

    public Flight getFlight(Long id) {

        return flightRepo.findById(id).orElseThrow(() -> new RuntimeException("Flight not found"));

    }

    public Flight updateFlight(Long id, Flight updated) {

        Flight flight = getFlight(id);

        flight.setOrigin(updated.getOrigin());

        flight.setDestination(updated.getDestination());

        flight.setDepartureTime(updated.getDepartureTime());

        flight.setSeatsAvailable(updated.getSeatsAvailable());

        return flightRepo.save(flight);

    }

    public void deleteFlight(Long id) {

        flightRepo.deleteById(id);

    }

}
```

ReservationService class

```
package com.example.FlightReservationSystem.service;

import java.time.LocalDateTime;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.example.FlightReservationSystem.entity.Flight;
import com.example.FlightReservationSystem.entity.Reservation;
import com.example.FlightReservationSystem.repository.FlightRepository;
import com.example.FlightReservationSystem.repository.ReservationRepository;

@Service

public class ReservationService {

    @Autowired

    private ReservationRepository resRepo;

    @Autowired

    private FlightRepository flightRepo;

    public Reservation makeReservation(Long flightId, Reservation res) {

        Flight flight = flightRepo.findById(flightId).orElseThrow(() -> new
        RuntimeException("Flight not found"));

        if (res.getSeatsBooked() > flight.getSeatsAvailable())

            throw new RuntimeException("Not enough seats");

        flight.setSeatsAvailable(flight.getSeatsAvailable() - res.getSeatsBooked());

        res.setFlight(flight);

        res.setReservedAt(LocalDateTime.now());

        flightRepo.save(flight);

        return resRepo.save(res);

    }

}
```

```

public List<Reservation> getAllReservations() {
    return resRepo.findAll();
}

public List<Reservation> getReservationsByFlight(Long flightId) {
    return resRepo.findByFlightId(flightId);
}

public void cancelReservation(Long id) {
    Reservation res = resRepo.findById(id).orElseThrow(() -> new RuntimeException("Not found"));

    Flight flight = res.getFlight();

    flight.setSeatsAvailable(flight.getSeatsAvailable() + res.getSeatsBooked());

    resRepo.delete(res);

    flightRepo.save(flight);
}
}

```

Application.properties

```

spring.application.name=FlightReservationSystem
spring.datasource.url=jdbc:h2:mem:flightdb
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.username=sa
spring.datasource.password=
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=update
spring.h2.console.enabled=true
spring.h2.console.path=/h2-console
springdoc.api-docs.path=/api-docs

```

springdoc.swagger-ui.path=/swagger-ui.html

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <parent>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-parent</artifactId>

    <version>3.5.4</version>

    <relativePath/> <!-- lookup parent from repository -->

  </parent>

  <groupId>com.example</groupId>

  <artifactId>FlightReservationSystem</artifactId>

  <version>0.0.1-SNAPSHOT</version>

  <name>FlightReservationSystem</name>

  <description>Demo project for FileReservationSystem</description>

  <url/>

  <licenses>

    <license/>

  </licenses>

  <developers>

    <developer/>

  </developers>

  <scm>

    <connection/>
```

```
<developerConnection/>

<tag/>

<url/>

</scm>

<properties>

  <java.version>21</java.version>

</properties>

<dependencies>

  <dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-data-jpa</artifactId>

  </dependency>

  <dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-web</artifactId>

  </dependency>


  <dependency>

    <groupId>com.h2database</groupId>

    <artifactId>h2</artifactId>

    <scope>runtime</scope>

  </dependency>

  <dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-test</artifactId>

    <scope>test</scope>

  </dependency>

</dependencies>
```

```
        <dependency>
<groupId>org.springdoc</groupId>
<artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>
<version>2.3.0</version> <!-- or latest stable -->
</dependency>
    </dependencies>

    <build>
        <plugins>
            <plugin>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-maven-plugin</artifactId>
            </plugin>
        </plugins>
    </build>

</project>
```