# Airport Information System

## Software Requirements Specification

## Purpose

- The goal of this project is to create a multi-functional system designed to manage tasks within the airport.

- The system differs from the usual system in that passengers, in addition to online ticket purchases, will be able to check in online for a flight on the airport's website, airline employees will be able to register their flights themselves without agreement based on the schedule of departures and arrivals. It will be able to serve several users and several airlines simultaneously.

- Information will be presented on the board, allowing the airport staff to see the status of each work, as well as manage them interactively.

## System features

❖ **Types of users:**
  - ➢ **Admin**
  - ➢ **User(passenger)**
  - ➢ **User(airline employee)**

❖ **Functionalities for admin:**
  - ➢ **Full access to application control**
  - ➢ **Create and delete**
    - ■ **Users in the database**
    - ■ **Airlines in database**
    - ■ **Flights in database**

❖ **Functionalities for users(passengers):**
  - ➢ **Online check in**
  - ➢ **Ticket purchases**

- The application allows you to buy a ticket online at the airport website

❖ **Functionalities for users(airlines):**
  ➢ **Flight registration**
    - **The application allows employees of the airline to**
      - **Check your flights on a schedule**
      - **Select the date and time of departure and arrival**
      - **Choose terminal and gate**
      - **Plan the placement of passengers on the number of seats in the plane**

❖ **Functionalities for passengers and airlines:**
  ➢ **View departure and arrival information**

# System restrictions

❖ **Possession of**
  ➢ **Each passenger can have only one ticket in his name for one flight**
  ➢ **Each passenger on a flight can have only one passenger seat.**

❖ **Choise**
  ➢ **A passenger cannot choose a seat already occupied by another passenger in the plane**
  ➢ **When purchasing online, a customer cannot purchase a ticket already purchased by another customer.**

❖ **Visibility of projects**
  ➢ **Passengers do not have access to the passenger list of the flight.**
  ➢ **Employees of airlines have access to the list of passengers of the flight**

# Workflow (Use Cases)

❖ **Use Case 1: Creation of a flight for passengers**

**Condition: Employee of the airline is registered and successfully logged in the system. A panel will appear.**

1.  **Employee clicks on the link "create flight".**

2.  **Employee correctly fills required fields with information about the flight and clicks save.**

    2.1.  **Enter data: airline name.**

    2.2.  **Place, date and time of departure and arrival, time of travel.**

    2.3.  **Aircraft number.**

    2.4.  **Number of passenger seats**

3.  **System saves the flight information, sets the flight creator to be its owner and redirects user to a flight's page.**

4.  **The employee chooses the airport terminal by searching in the search window located on the main flight page and after that chooses the free gate in the terminal.**

5.  **The system adds selected parameters to the flight.**

## ❖ Use Case 2: Online ticket purchase by user

**Condition: User is registered and successfully logged in the system. A panel will appear.**

1.  **The user clicks on the link "buy". The user correctly fills the necessary fields with information about himself and saves clicks.**

    1.1.  **The user specifies the place of departure and arrival**

    1.2.  **The date of departure**

    1.3.  **The number of passengers.**

2.  **The system saves the user's information**

## ❖ Use Case 3: Online check-in

**Condition: At least before departure**

1. A passenger enters his ticket number
2. The system checks the availability of a ticket in the database by its number, if it exists, then sends it to a new page with a form, if not, it displays an error.
3. The passenger chooses a seat in the plane and clicks on the "select" button
4. The system checks if this place is free, then saves the data and displays the message about registration success to the client, if not, then writes that the place is already taken.

# Instructions "How to install the application"

- Download the project, then in the "server" folder
- Open the xml file in the ide, after opening and loading all the settings with the environment, press the "start" in the upper corner while it is being built
- Launch the frontend part through the desktop and start the frontend and in 1-1.5 minutes
- The application will be installed and launched.

# Used technologies

❖ Frontend
  ➢ JavaScript
  ➢ Css
❖ Backend
  ➢ Java
  ➢ Spring Boot Framework
❖ Database

# EAR diagram

## TERMINAL
+ id: int<PK>
+ flight_id : int<FK>

## PLANE_SEAT
+ number: int
+ plane_id: int<FK>
+ free : boolean

## FLIGHT
+ id : int<PK>
+ plane_id: int<FK>
+ airline_id: int<FK>
+ terminal_id: int<FK>
+ department_date: datetime
+ arrivals_date: datetime
+ department: String
+ arrival: String
+ distance: int

## TICKET
+ id: int<PK>
+ passenger_id: int<FK>
+ flights_id : id<FK>
+ price : numeric
+ seat_number : int<FK>

## PLANE
+ id: int<PK>
+ airline_id: int<FK>

## AIRLINE
+ id : int<PK>
+ name: String
+ email : String
+ phone_number : int

## PASSENGER
+ name: String
+ surname: String

## USER
+ id : int<PK>
+ email : String
+ password : String
+ phone_number : int

## EMPLOYEE
+ airlineid: int<FK>