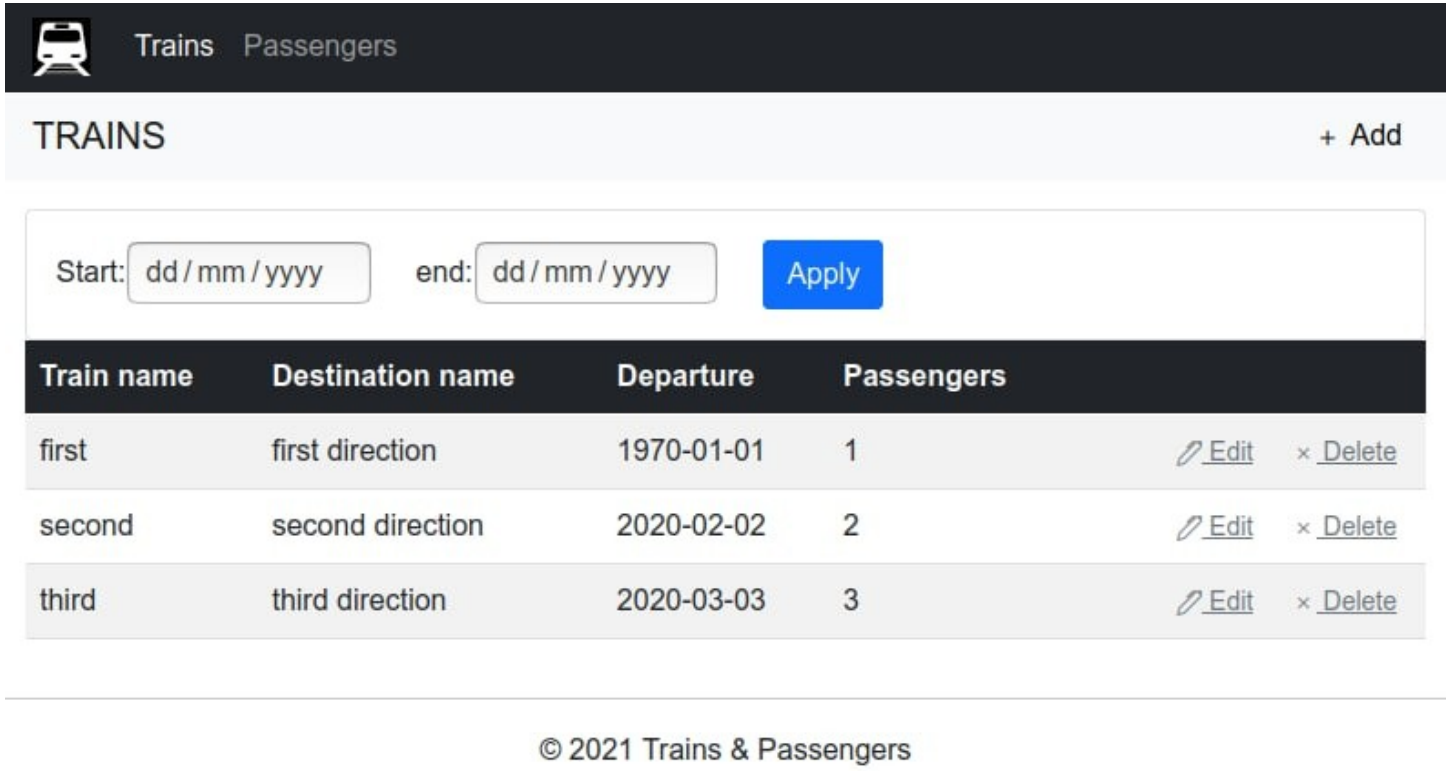
# Trains & passengers

**This sample application was designed to store trains and passengers in the database.**

**The application performs the following functions:**

- Viewing and editing the list of trains for the selected period, with the number of passengers.
- Viewing and editing the list of passengers.



# Technologies used:

- **Spring MVC** - framework providing hooks for extension and customization for web applications and RESTful (representational state transfer) Web services.
- **Spring boot -** solution for creating stand-alone Spring-based Applications
- **Bootstrap** - framework for responsive, mobile-first front-end web development.
- **Thymeleaf** - server-side Java template engine for both web and standalone environments.
- **Tomcat -** servlet container.

# Development tools:

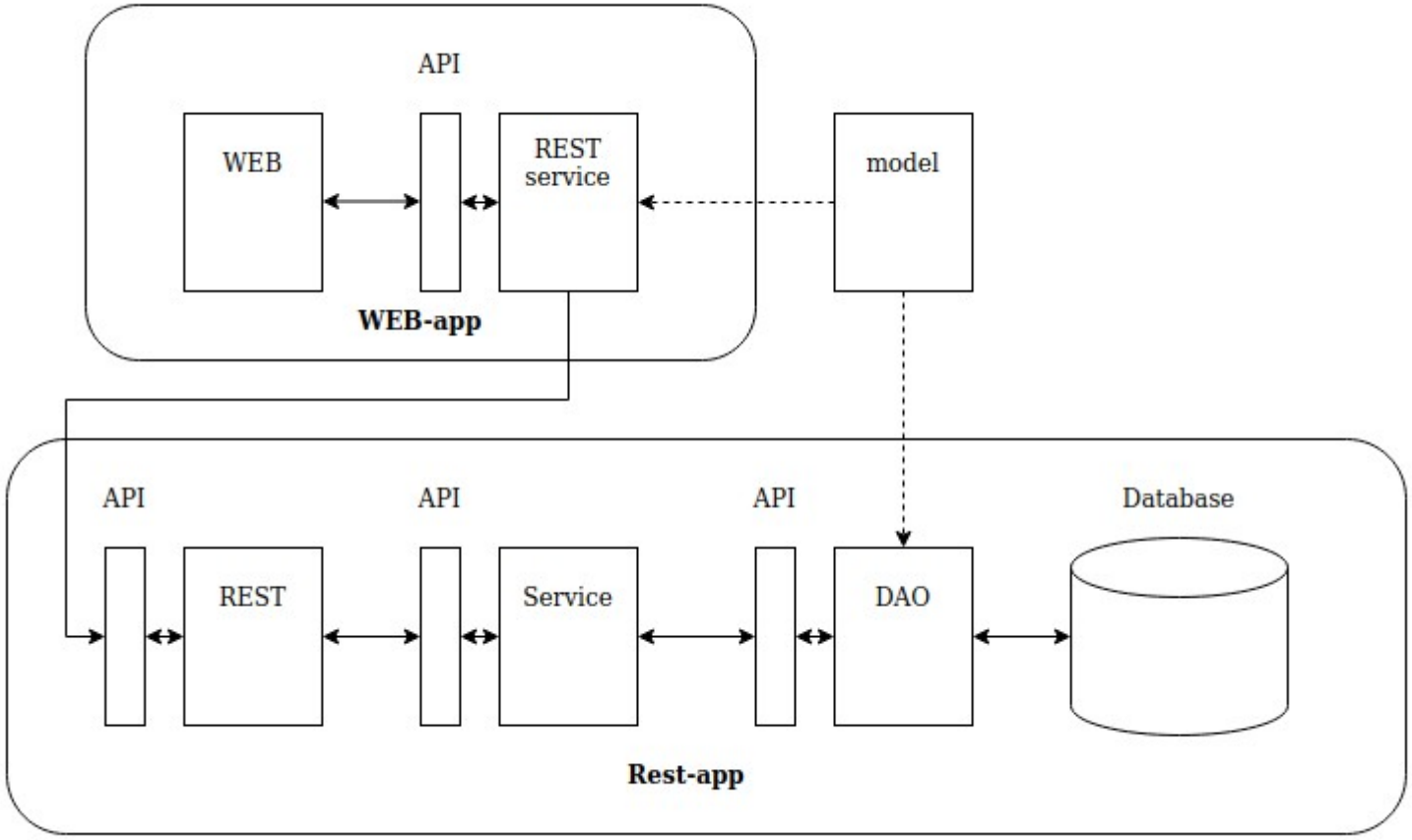
- **Java** is the programming language.
- **JUnit** - unit testing framework.
- **Git** - software for version control and tracking changes.
- **IntelliJ IDEA** - integrated development environment.
- **Maven** - build automation tool.

# Objectives of the project:

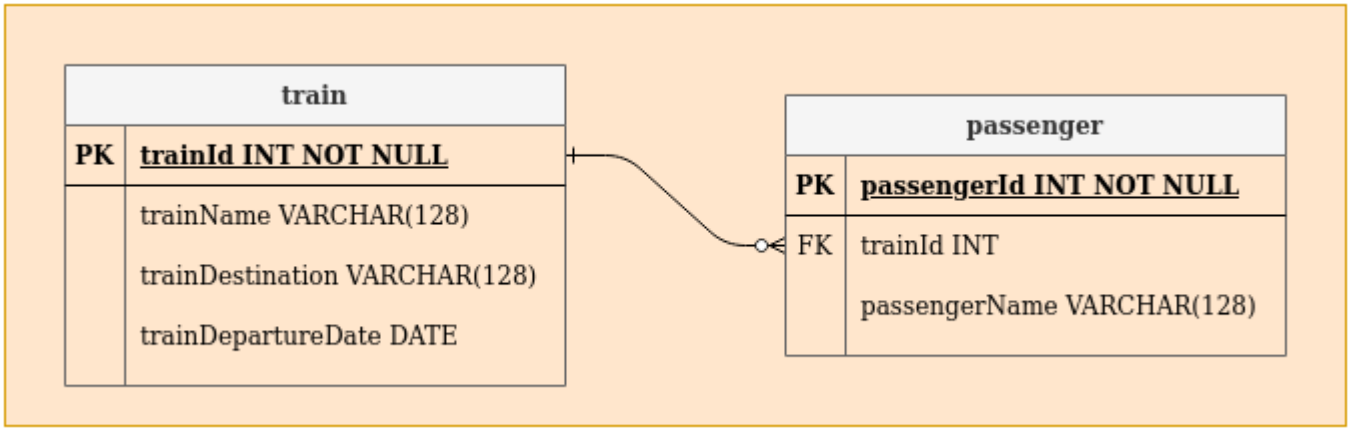
Creation of an application for registering trains and passengers in the database.

Acquiring skills in using various other technologies.

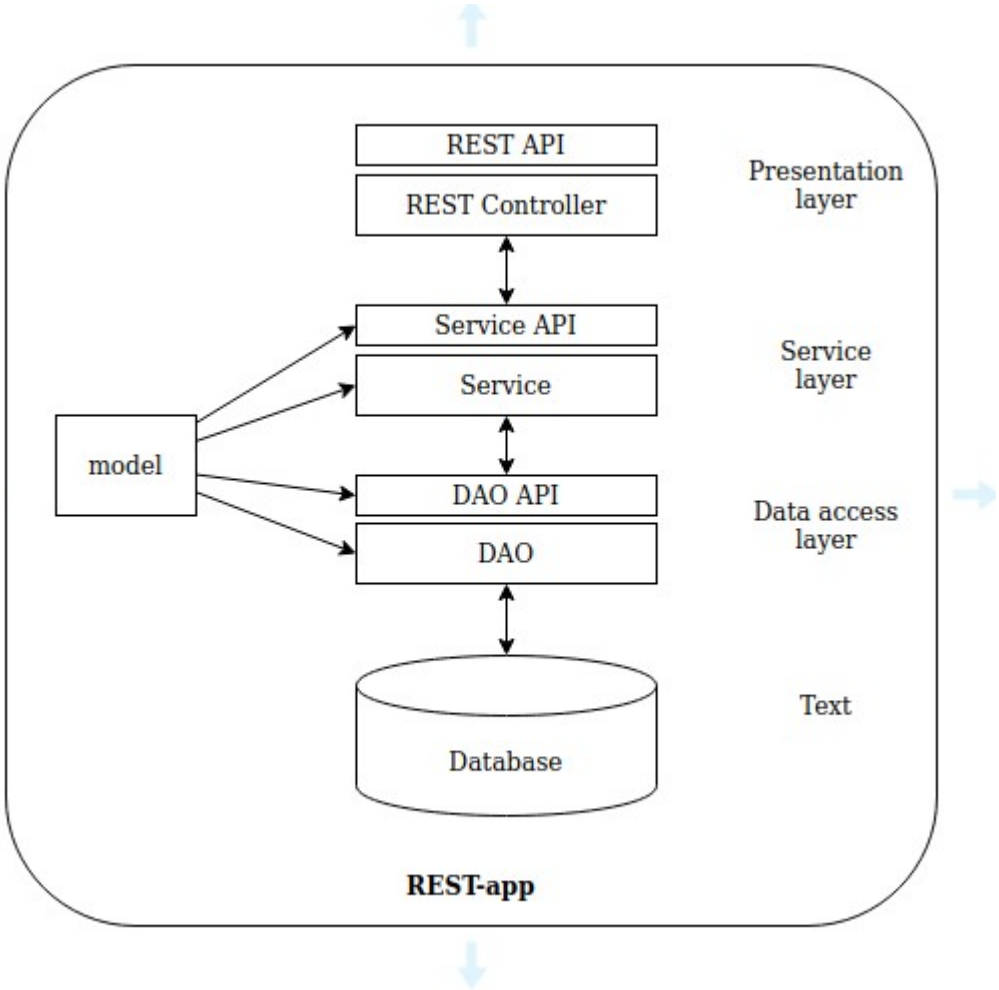
# Project structure:



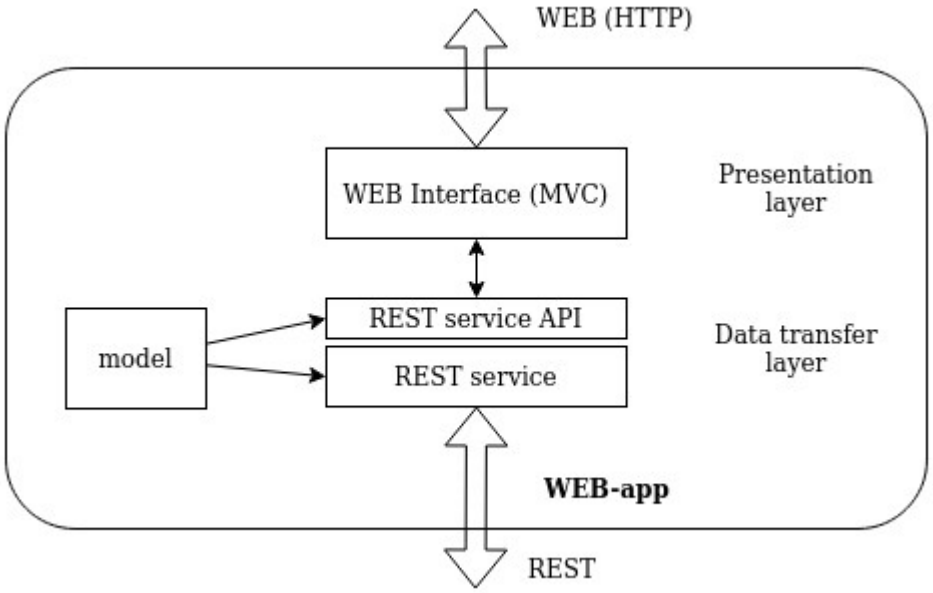
# Database structure:



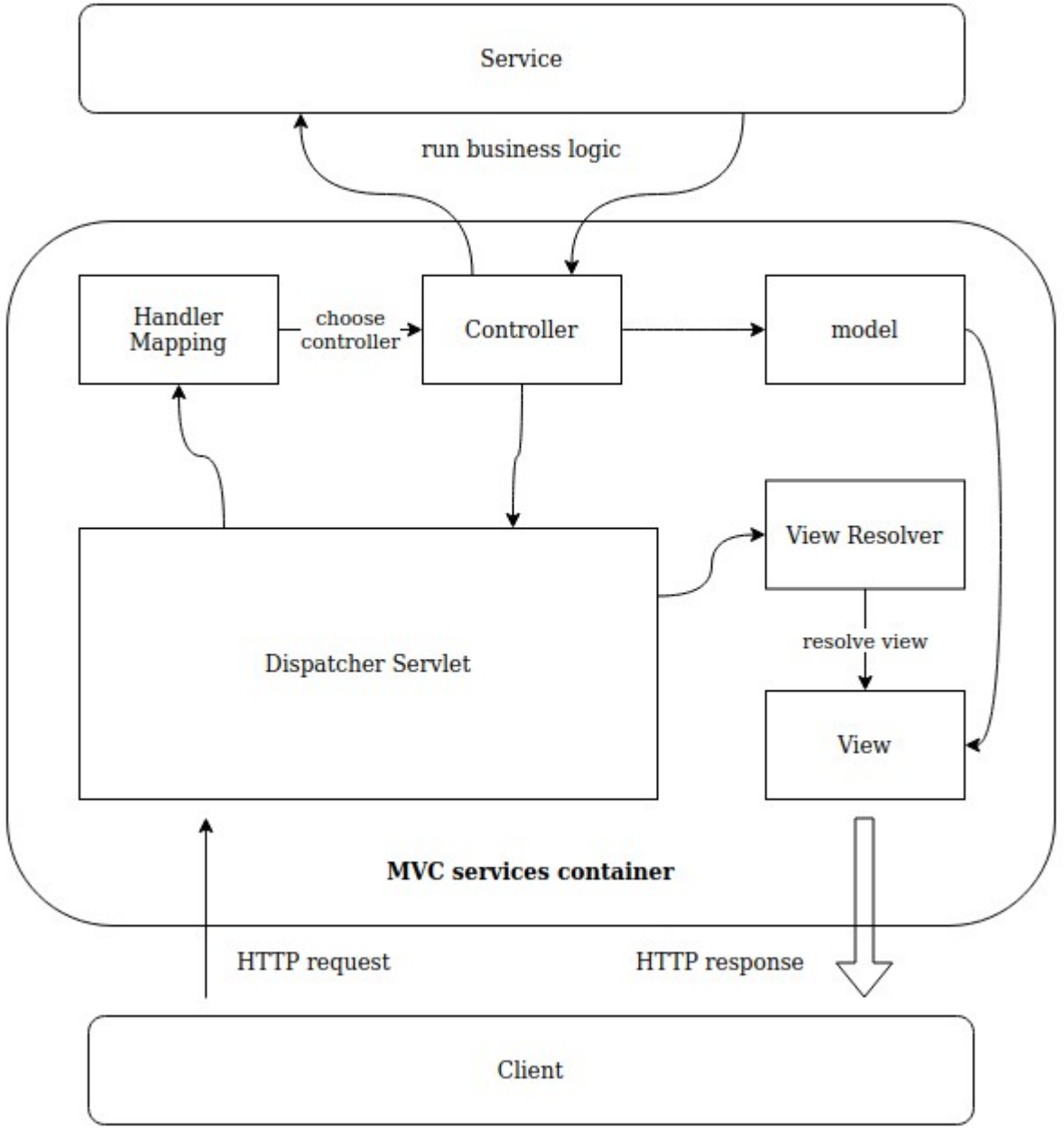
# REST – application:



# WEB – application:



# MVC


Service
run business logic
Handler Mapping
choose controller
Controller
model
View Resolver
Dispatcher Servlet
resolve view
View
MVC services container
HTTP request
HTTP response
Client

# Workflow

The user, using the browser, accesses the web application server.

From the web application, the request goes to REST services.

The REST service refers to the REST controller.

From the REST-controller the request is sent to the services.

There is no business logic so the reuest is sent further to the DAO.

DAO forms SQL query to the database.

The database returns data to DAO.

The data is collected into a collection and returned to services.

Services transfer data to the REST controller.

The REST controller returns data to the REST service.

The web application transfer to browser HTML page.

The user enjoys the result.