

API med Spring Boot

Bakgrund

Wigellkoncernen har beslutat att bygga ett nytt bokningssystem för sin biluthyrning. Två grupper jobbar med projektet där en grupp ansvarar för användargränssnittet och en annan för det API som användargränssnittet kommunicerar med.

Du arbetar med API:et som ska utvecklas med Java Spring Boot

API:et ska lanseras måndag 8/5 kl 00:01 så samtliga projekt måste vara inlämnade då!

Kravspecifikation

Funktionella krav

Systemet ska ha funktionalitet för både kunder och administratörer.

Kund

Systemet ska kunna hantera kunder, bilar, bokningar.

Bilar ska ha pris per dag, fabrikat, modell, registreringsnummer och uppgift om den är bokad eller inte.

Kunder ska ha unika id-nummer, användarnamn och kunduppgifter som namn, adress, e-post och telefonnummer.

Bokningar ska inkludera datum för bokningen.

Kunderna ska kunna göra ett antal aktiviteter med följande endpoints:

- Lista tillgängliga bilar
GET /api/v1/cars
- Beställa hyrbil
POST /api/v1/ordercar
- Avboka
PUT /api/v1/cancelorder
- Se tidigare och aktiva bokningar
GET /api/v1/myorders

Admin

Administratörer ska kunna göra ett antal aktiviteter med följande endpoints:

- Lista kunder
GET /api/v1/customers
- Lägga till fordon
POST /api/v1/addcar
- Ta bort fordon
DELETE /api/v1/deletocar
- Uppdatera fordon
PUT /api/v1/updatecar

- Lägga till kund
POST /api/v1/addcustomer
- Uppdatera kund
PUT /api/v1/updatecustomer
- Ta bort kund
DELETE /api/v1/deletcustomer
- Lista samtliga bilar
GET /api/v1/allcars
- Lista bokningar
GET /api/v1/orders
- Ta bort bokning
DELETE /api/v1/deleteorder

Viktigt att alla endpoints följer specifikationen eftersom de ska fungera med användargränssnittet.

Loggning

Applikationen ska logga info i en loggfil när något skapas, ändras eller tas bort. Skriv ett meddelande som talar om vad som händer t.ex. "admin created car abc",

Tekniska krav

Skapa de klasser som behövs för att lösa uppgiften och är vettiga designmässigt från informationen i kraven.

Strukturera applikationen med lämpliga package så som controller, service, entity osv.

Data ska lagras i en H2 databas. Databasen ska heta twrental.

Verifiera med postman att alla endpoints går att använda enligt specifikation.

Övrigt

Ingen frontend ska implementeras. All kommunikation med API:et ska ske via Postman

API:et ska nås på port 9090

Java 11 ska användas

Databasen ska initieras med fem kunder, fem bilar och tre bokningar vid start av applikationen

Log4J ska användas