allane mobility group

Allane SE

Code-Challenge
Fullstack-Developer

17.02.2022

1.1.1 Allane SE – Code Challenge

1.2 Business Requirements

We have the need to implement a leasing application to be able to administrate leasing contracts.

1.2.1 Models

1.2.1.1 Leasing Contract

A leasing contract consists of the following properties:

- Contract Number
- Monthly Rate

1.2.2 Customer

A leasing contract is connected to one customer and a customer could have several leasing contracts.
A customer consists of the following properties:

- First name
- Last name
- Birthdate

1.2.3 Vehicle

A leasing contract is connected to one specific vehicle and a vehicle could only be connected to one contract at a time.
A vehicle consists of the following properties:

- Brand
- Model
- Model year
- Vehicle identification number when the vehicle is produced. During creation of contract it could be empty.
- Price

1.3 Frontend

The application for maintaining the contracts should look like at least as follows. If you see potential to improve things feel free to do so.

© Allane SE 2021. All rights reserved.
Confidential and Proprietary

1.3.1 Contract Overview

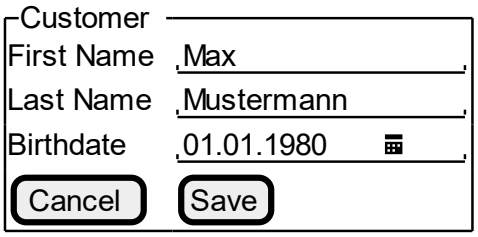
Contract Overview

Contract No	Customer	Vehicle	VIN	Monthy Rate	Vehicle Price	Details
123567	Max Mustermann	BMW X3 (2022)	X123456	350,00 €	45.350,00 €	
123568	Maria Musterfrau	BMW 330i (2022)	-	365,00 €	47.350,00 €	

There is planned to show a contract overview with some columns:

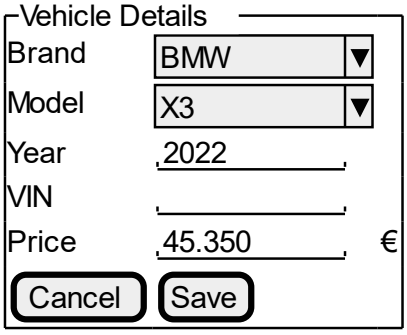
- Contract No
- Summary of the Customer
- Summary of the Vehicle
  - Brand
  - Model
  - In braces: Model year
- VIN if available or "-" if empty
- Monthly Rate
- Vehicle Price
- Link to the Contract Details

1.3.2 Customer Details



Edit mask to create/edit a customer.

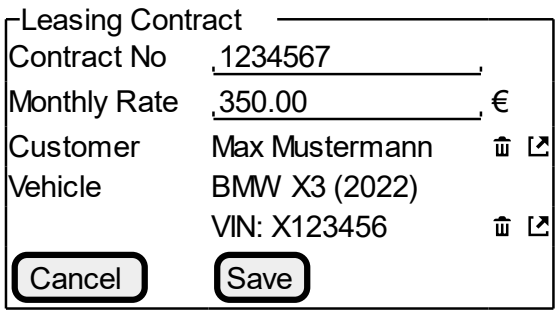
1.3.3 Vehicle



Edit mask to create/edit a vehicle.

© Allane SE 2021. All rights reserved.
Confidential and Proprietary

1.3.4 Leasing Contract



Edit mask to create/edit a leasing contract.

1.4 Task

- Implement a REST service to fulfil the requirements and has to be used by the frontend.
- The data should be persisted into a database.
- Model the entities for persistence.
- Implement the frontend to fulfil the requirements derived from the wireframes.
- Write unit test if appropriate.

1.5 Technical Constraints

The following technologies should be used:

- Java 11
- Spring Boot >= 2.5
- MySQL/MariaDB Database (locally dockerized)
- Gradle
- Flyway database migration to create the initial schema
- Optional: OpenAPI specification to generate server/client.
- Angular >= 13

1.6 Result

- Provide your solution as git repo to be able to see how the progress was.
- Document in a README
    - Preconditions needed to run the application
    - How to start the application
    - Reason of chosen solution

© Allane SE 2021. All rights reserved.
Confidential and Proprietary