

# Java 1

## Documentation

Team: MMRA  
project name: CarHub360

### TEAM MEMBERS

name	specialization direction
Rudolf Minz	Medieninformatik
Mohammad Tahiba	ingenieurinformatik
Marc Praus	ingenieurinformatik

# Introduction

Welcome to CarHub360, a comprehensive management system designed specifically for car sellers and renters. CarHub360 aims to streamline and simplify the workflow for employees, making day-to-day operations more efficient and manageable.

CarHub360 offers a user-friendly interface and a wide range of robust features to support the management of vehicles, maintenance operations, contracts, and customer interactions. This system ensures that employees can handle their tasks with ease and accuracy, leading to enhanced productivity and customer satisfaction.

The Vehicle Module allows employees to create, update, delete, and retrieve details of vehicles in the system, ensuring that all vehicle information is current and easily accessible. The Maintenance Module tracks and manages maintenance activities, helping to keep vehicles in optimal condition and reducing downtime.

For rental operations, the Rent Vehicle Module provides tools to manage vehicles available for rent, including setting daily prices and handling rental agreements. The Contract Module facilitates the creation and management of both rental and purchase contracts, ensuring smooth transactions and clear record-keeping.

Managing customer information is effortless with the Customer Module, which handles customer details and ensures data integrity. The Customer Address Module further aids in maintaining accurate and up-to-date address information for all customers.

The Payment Module processes and manages customer payments, ensuring that all financial transactions are handled securely and efficiently. For vehicles available for sale, the Sale Vehicle Module provides comprehensive details and management capabilities.

Finally, the Customer History Module maintains a detailed history of actions related to customers, including reviews and descriptions of interactions, helping to build a complete profile of customer relationships and preferences.

Overall, CarHub360 is designed to be a powerful and intuitive tool for car sellers and renters, providing all the features and functionality needed to manage vehicles, operations, contracts, and customer information efficiently. We hope that CarHub360 will be an invaluable resource for your business, making your work easier and more effective.

# Data Model and Logic

## Customer Module:

The customer module is responsible for managing customer information, including attributes such as customer ID, first name, last name, email, birthdate, gender, deletion status, and address.

Create Customer: Users can add new customers by providing necessary details. The system ensures that no duplicate customers are created by checking the customer ID and email.

Delete Customer: Users can remove a customer from the system by their unique customer ID. The system ensures the correct customer is deleted.

Get Customer Details: Users can retrieve detailed information about a specific customer, including their personal details and address.

## Customer Address Module:

The customer address module manages customer addresses, including attributes such as customer ID, city, postal code, street, and street number.

Update Customer Address: Users can update the address details for an existing customer. The system ensures that the address data is valid and updates the records accordingly.

Get Customer Address Details: Users can retrieve the address details of a specific customer using their ID. This ensures that users have access to accurate and up-to-date address information.

## Customer History Module:

The customer history module maintains the history of actions related to customers, including attributes such as customer history ID, customer, vehicle involved, review, description, action date, and indication of whether it's for a rental car.

Create Customer History: Users can create a new customer history record by providing customer history ID, customer, vehicle involved, review, description, action date, and indication of whether it's for a rental car. The system ensures that the history record is valid and updates or adds the record accordingly.

**Get Customer History:** Users can retrieve a customer history record associated with a specific ID. This helps in tracking the history of actions related to customers.

**Get Customer Final Review:** Users can retrieve the final review associated with a specific customer history ID. This helps in understanding the overall experience and feedback of the customer.

## **Vehicle Module:**

The vehicle module allows users to create, update, delete, and retrieve details of vehicles in the system. Users can manage various attributes such as vehicle ID, name, brand, kilometer count, construction year, and type.

**Create Vehicle:** Users can add new vehicles with valid attributes. The system validates inputs like ID, construction year, and kilometer count to ensure data integrity.

**Update Vehicle:** Existing vehicle records can be updated with new information, provided the vehicle exists in the system.

**Delete Vehicle:** Vehicles can be removed from the system if they are no longer in use.

**Get Vehicle Details:** Detailed information about a specific vehicle can be retrieved using its ID.

## **Sale Vehicle Module:**

The sale vehicle module manages vehicles available for sale, including attributes such as vehicle ID, sale price, and whether the vehicle is new or used.

**Get Sale Vehicle Details:** Users can retrieve comprehensive details of a sale vehicle, including ID, sale price, and whether the vehicle is new or used. This helps in managing sales operations efficiently.

## **Rent Vehicle Module:**

The rent vehicle module allows users to manage vehicles available for rent, including setting daily prices and license plate information.

**Get Rent Vehicle Details:** Users can retrieve comprehensive details of a rental vehicle, including ID, daily price, and license plate. This helps in managing rental operations efficiently.

## **Maintenance Module:**

The maintenance module is used to track and manage the maintenance activities of vehicles. This includes adding new maintenance records and retrieving maintenance details.

**Add Maintenance:** Users can log maintenance activities for vehicles, including start and end dates, cost, and description of the maintenance work.

**Get Maintenance Details:** Detailed information about a specific maintenance record can be retrieved using its ID. This feature ensures that users can track the maintenance history and costs associated with each vehicle.

## **Contract Module:**

The contract module handles the creation, management, and retrieval of both rental and purchase contracts.

**Create Purchase Contract:** Users can create a new purchase contract by providing contract ID, customer details, and the sale vehicle. The system validates the input data and ensures that the contract does not already exist.

**Create Rental Contract:** Users can create a new rental contract by providing contract ID, customer details, rental vehicle, and rental period. The system validates the input data, including the rental period, and ensures that the contract does not already exist.

**Terminate Rental Contract:** Users can terminate an existing rental contract by setting its end date to the current date and marking the rent vehicle as available.

**Renew Rental Contract:** Users can renew an existing rental contract by extending its end date. The system ensures that the new rental period is valid.

**Get Total Price:** Users can retrieve the total price of a rental contract based on the number of days rented and the daily price of the rent vehicle.

**Get Rental Contract Details:** Users can retrieve detailed information about a specific rental contract, including customer details, rent vehicle details, and the rental period.

**Get Purchase Contract Details:** Users can retrieve detailed information about a specific purchase contract, including customer details and sale vehicle details.

## **Payment Module:**

The payment module handles the processing and management of customer payments, including attributes such as payment ID, customer details, payment method, payment status, and payment amount.

**Process Payment:** Users can process a payment by providing payment ID, customer details, payment method, payment status, and payment amount. The system validates the input data to ensure the payment is processed correctly.

**Get Payment Details:** Users can retrieve detailed information about a specific payment, including payment method, payment status, and payment amount.

Each of these modules is designed to provide a seamless experience for users managing vehicles, maintenance operations, contracts, customer information, payments, sales, and customer history, ensuring that all operations are handled efficiently and accurately.

# Requirements and dependencies

In order to run CarHub360, you will need to meet the following requirements:

1. **Operating system:** CarHub360 is compatible with Windows 10, macOS, and Linux.
2. **Programming language:** CarHub360 is written in Java and requires at least version 8 of the Java Development Kit (JDK) to be installed on your system.
3. **Testing:** CarHub360 includes unit tests that are written using JUnit. In order to run these tests, you will need to have JUnit installed on your system.
4. **Build management:** CarHub360 uses Maven for build management. You will need to have Maven installed on your system in order to build and run CarHub360.
5. **Other dependencies:** CarHub360 does not use any third-party libraries and frameworks.

Please make sure that you meet these requirements before attempting to install and run CarHub360.

If you have any questions or encounter any issues, please consult the troubleshooting section of this documentation or contact us for assistance.

## Installation instructions

To install and set up CarHub360, follow these steps:

1. **Download** or clone the latest version of CarHub360 from our git repository (<https://git.ai.fh-erfurt.de/prgj1-23/mmra>)
2. **Extract** the downloaded file to a folder on your computer.
3. **Install** the Java Development Kit (JDK) (<https://www.oracle.com/java/technologies/downloads/>) if you don't already have it installed. You can download the JDK from the Oracle website.
4. **Install** and **configure** an IDE, such as Eclipse ([https://www.eclipse.org/downloads/download.php?file=/oomph/epp/2023-12/R/eclipse-inst-jre-win64.exe&mirror\\_id=1301](https://www.eclipse.org/downloads/download.php?file=/oomph/epp/2023-12/R/eclipse-inst-jre-win64.exe&mirror_id=1301)) or IntelliJ IDEA (<https://www.jetbrains.com/idea>).
5. **Install** JUnit and Maven if you don't already have them installed. You can find instructions for

installing JUnit and Maven on the JUnit (<https://junit.org/junit5>) and Maven (<https://maven.apache.org/what-is-maven.html>) websites, respectively.

After completing these steps, you should be ready to build and test CarHub360. If you have any issues or questions during the installation process, please consult the troubleshooting section of this documentation or contact us for assistance.