Project Description

Buğra Derre [derre@lehre.dhbw-stuttgart.de]

School of Engineering, Computer Sciences, Prof. Dr.-Ing. Alfred Strey

Business Overall Goal

EvilCar wants to extend its business and increase its market share. Therefore EvilCar brings new features to their customers:

1. the spontaneous car rental enables customers to rent a car spontaneously at their next EvilCar branch. They can choose their rental return spontaneously.
2. The EvilCar fleet is fully connected. With those brand-new cars a customer is able to book services, e.g. Spotify or the Massage service.

Business Terms

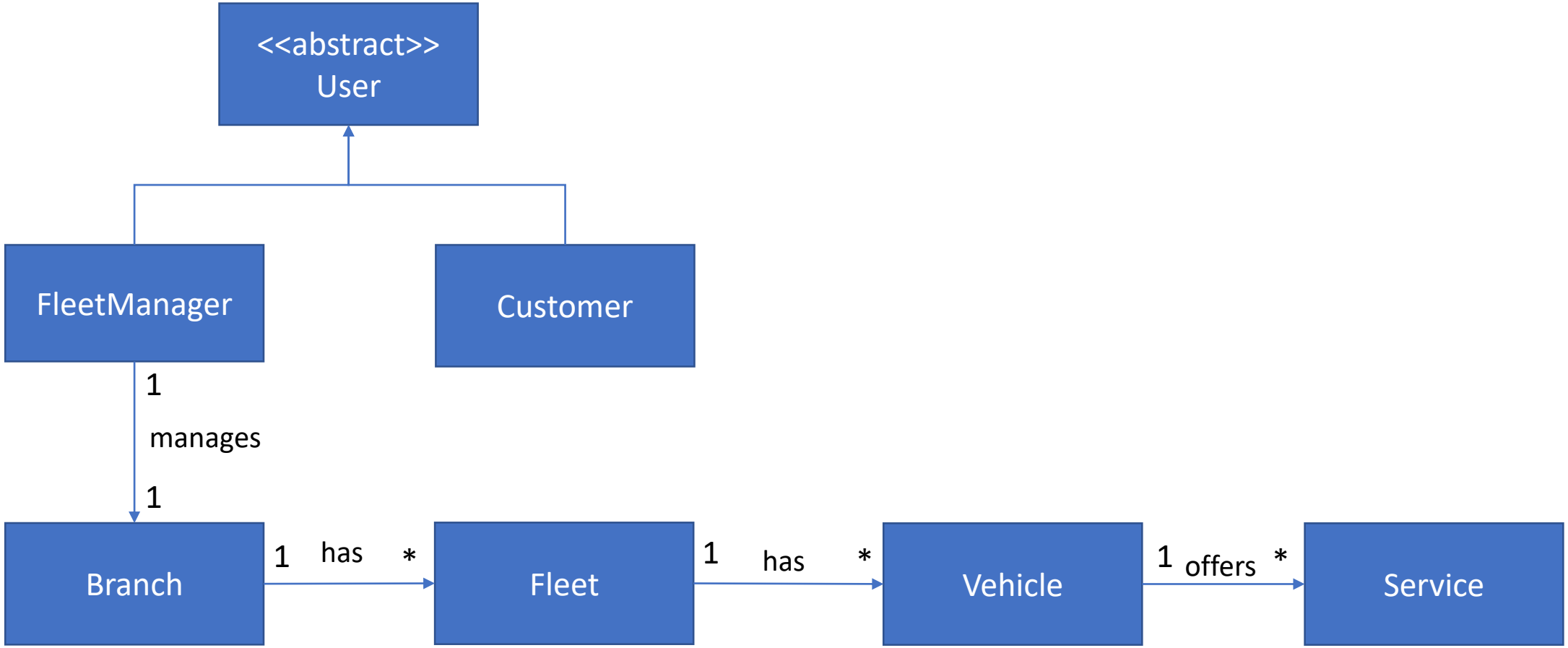
- **Fleet managers** are responsible for a EvilCar branch (1) and its fleets.
- A **branch** belongs to a fleet manager and has several fleets (1..*).
- A **fleet** is a pool of vehicles (1..*) that belongs to a EvilCar branch.
- A **vehicle** offers services that a customer can consume. For example customers can rent a navigation service.
- **Customers** rent vehicles and its services.

Your Goal

You will help EvilCar in administering the EvilCar **fleet management portal**. This portal is tremendously important to achieve the goals of EvilCar 2025.

The first prototype that you build will be a pure console application (no server technologies, graphic and web interfaces). This enables EvilCar for a rapid development and testing of their business ideas.

Domain


<<abstract>>
User
FleetManager
Customer
1
manages
1
Branch
1
has
*
Fleet
1
has
*
Vehicle
1
offers
*
Service

Movie Script

Scenario 1: Seed Data

Problem statement: A prototype is insufficient when you have no seed data (=demo data). Provide seed data in the form of input files, either XML or JSON:

- John Evilman shall be the fleet manager of the branch located in Hamburg, Germany. He owns two fleets. The first fleet provides small motorized four-wheel vehicles for his customers. The second fleet offers midsize motorized vehicles.

- Liz Evilgirl is the fleet manager of the branch located in Stuttgart, Germany. She specialized on large executive vehicles. The first fleet provides sport vehicles, the second fleet electric vehicles.

Why? Scenario 1: Seed Data

- Think about your file structure and learn the downsides and upsides of different file structures (choose from a nested file structure, flat file structure or a mix of both)
- Learn about the usage and file handling in C#
- Hint: You are allowed to use libraries for example Json.NET
- Hint: Every data record has a universal unique identifier. Use the GUID class to create one (Guid.NewGuid). This identifier is visible only in the file.

Scenario 2: Managing Fleets

Problem statement: The fleet manager has to administer his fleets.

- The fleet manager must be able to add new vehicles, edit vehicles, remove vehicles and to rebook a vehicle from one fleet to another fleet.
- The fleet manager is restricted to administer fleets of his branch only.
- The fleet manager needs to log in with his credentials first (username and password)

rebook as an option of edit

Why? Scenario 2: Managing Fleets

- Learn about identity management
- Learn about authentication and authentification which is an essential part of business applications

Scenario 3: Managing Customers

Problem statement: The fleet manager has to administer his customers.

- The fleet manager must be able to add new customers, edit their profile information, and remove them from the customer database.
- The fleet managers share a common customer database across all branches.

Why? Scenario 3: Managing Customers

- Learn about identity management
- Learn about authentication and authentification which is an essential part of business applications

Scenario 4: Managing Vehicle Services

Problem statement: The fleet manager has to administer vehicles.

- A vehicle provides different services. See the detailed list of services that different vehicles provide. Think of two other services that could be interesting for a driver.
- The possible services of the vehicle depend on the vehicle type. A large executive vehicle offers a seat massage service whereas a small sized vehicle does not.
- The services have a price. See the detailed list of services
- The administration of the services take effect on all vehicles of all branches

Scenario 4: Managing Vehicle Services

Service	Price	Small	Midsize	Large	Electric
Navigation	5 €	✓	✓	✓	✓
Massage	15 €	✗	✗	✓	✗
Charging Station Finder	10 €	✗	✗	✗	✓
Spotify	8 €	✗	✓	✓	✓
Think of two more services					
…					

Why? Scenario 4: Managing Vehicle Services

- Learn about domain modeling of business software applications
- Learn about using approriate software programming mechanisms

Scenario 5: Make the cash

Problem statement: The EvilCar needs money.

- The fleet manager needs to generate bills for the customer. The customer pays for the services and the vehicle he rented. Prices contribute on a per day basis.
- The price of a rental are calculated by the vehicle type (not its model). See a detailed list of the types and the prices.

Scenario 5: Make the cash

Model (Small)	Price
Smart fortwo	30 €

Model (Medium)	Price
Volkswagen Golf	60 €

Model (Large)	Price
BMW X6	90 €

Model (Electric)	Price
Tesla Model S	130 €

Why? Scenario 5: Make the cash

- Learn about domain modeling of business software applications
- Learn about using approriate software programming mechanisms

Scenario 6: GDPR

Problem statement: The EvilCar is located in the EU. Since 2018 they need to regard the GDPR.

- One of the laws convey that the customer has the right for transparency. The customer has the right to see the collected data. Since EvilCar collects all bills of a customer they need to show which tours the customer did.
- The customer has the right to be deleted. For EvilCar that means that they need a functionality to delete all data related to a customer.

Why? Scenario 6: GDPR

- Hint: Do it asynchronous.
- Learn about the modern problems of software application
- Learn about data handling

Deadline

- When: 12.05.2019, 23:59
- How: the project‘s source code as zip file
- To: derre@lehre.dhbw-stuttgart.de
- Please
  - write the group name and its members in the mail text

Programming Guidelines

- Feature Complete?
- Readability
- Clean Code
  - DRY
  - KISS
  - No need to optimize code for performance
  - Naming of members
  - Methods <= 20 LOC