

Assignment 2
Johnny Rosas
Patrick Simpauco
Jenny Tran
10/13/23

BeAvis Car Rental System

System Overview:

The BeAvis rental system was developed to replace the pen-and-paper rental process. The car rental software system can be accessed through mobile and website applications. Both applications will have the same functionality and will both be user friendly. Every new user that begins their rental process will be prompted for their location/GPS services, which will allow the system to look for the nearest rental car center. In order for the user to rent a car, they will need to create an account with the following information(email, user, password, ...see UML). Shortly after creating their accounts, the user must verify their accounts using a verification code sent to their email to finalize the process. Not only do the accounts give rental access to users, the accounts also give the user access to toggle on/off sharing location/GPS services with the system. The rental system will be like any other, with the ability to browse rental cars, make a rental, view rental history, and lookup locations. The user will also have the choice to save their preferred payment method for convenience. The system will also take care of all the user's confidential information, payment info, rental agreements/contracts, and rental history by having separate data centers. To conclude, the system will give access to Employee accounts to pull up customers' rental status and contracts, check the number of cars available, and check to see if any cars need to be pulled for maintenance.

Software Architecture Overview:

Software Architecture diagram of all major components

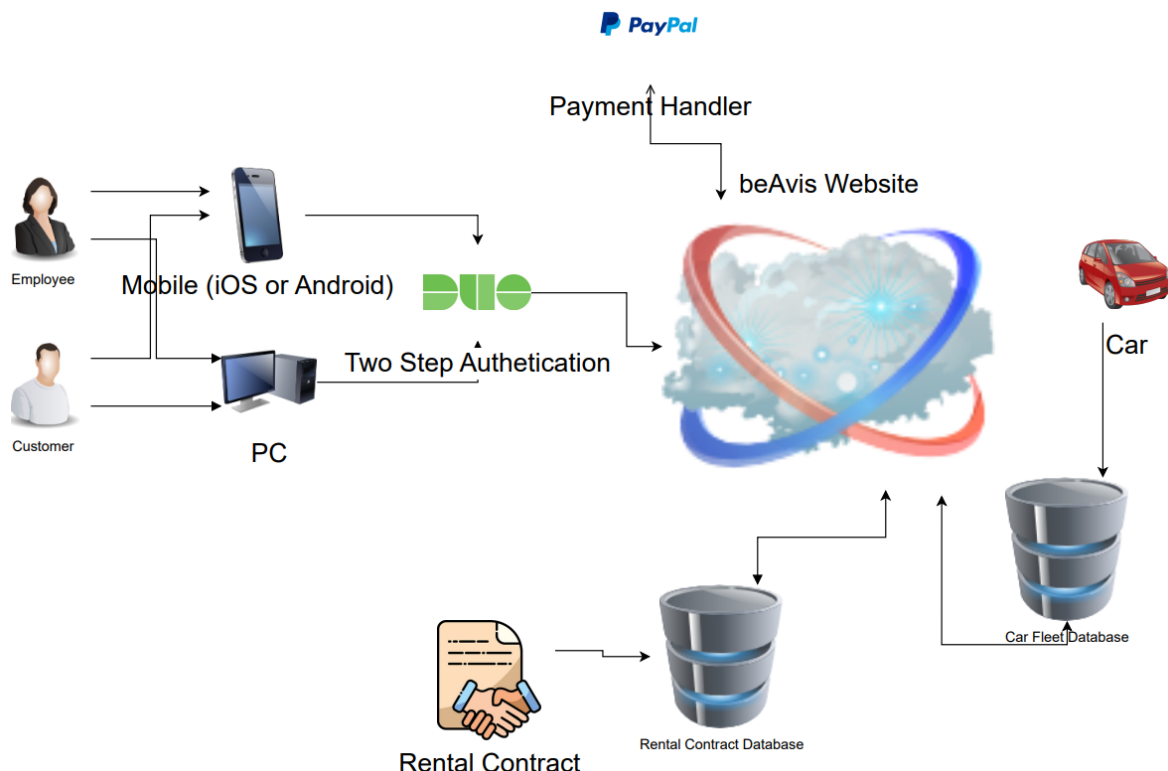


Diagram Components

1. Employee and Customer

- These represent the different users that will interact with the system. Users are divided into two categories: customers and employees. Each contains various ways of interacting with the system and will perform actions according to what kind of user they are.

2. Mobile and PC

- The BeAvis Car Rental system will be hosted on a website or mobile application. As per the user requirements, the mobile application is with the latest version of iOS or Android. While the design and layout may vary between mobile application and the website, the functionality should be the same.
- The single arrow connection displays its utilization by employees and customers.

3. Two Step Authentication

- Users will have the option to enable two step authentication via 3rd party authentication apps. This feature is not enabled initially, but can be added by the user.
- Its connection pointing to the system represents how the authentication process will lead to the access of the BeAvis website and mobile app.

4. Payment Service

- Payment information can be added to a customer's account through the BeAvis website or mobile app. Transactions are handled externally (system not specified) to ensure the privacy of customer data.
- The two-way connection between the payment service and the BeAvis website display how transactions are handled separately while information is still relayed between both services.

5. BeAvis Website

- This contains both the customer and employee portals of the system. Contains most of the functionality related to the rental process for customers and the rental management process for employees.

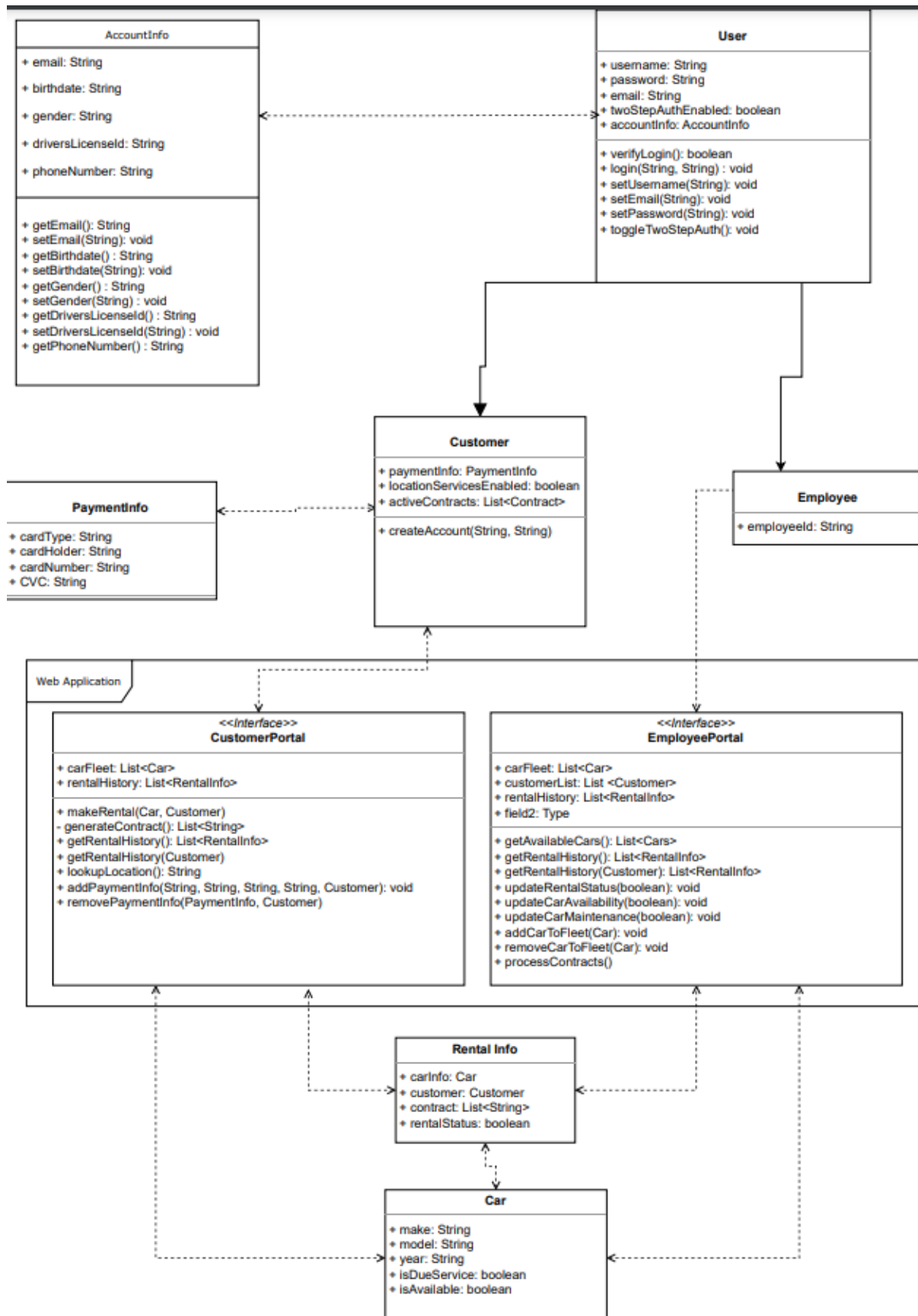
6. Rental Contract and Car Fleet Databases

- Per the client's requirements, online contracts are kept in secure conditions and the history of rental contracts should be viewable by employees.
- The car fleet database holds the information about cars in the rental lot. Its connection to the system allows both customers and employees to view and interact with cars for the rental process. The arrow pointing to the database represents how users will contribute information that will eventually be archived to the database.

7. Rental Contract and Car

- Both components show the actual objects being represented with the databases.

UML Class Diagram



Description of classes, attributes and operations:

1. AccountInfo:

- Containing basic information about the user that will be helpful in verifying their identity in real life for rental
- Attributes: User's identification information like birthdate, gender, driver licenses and phone number
- Operation: Getter and setter methods to view and modify user's information

2. User:

- Encapsulates all information and actions related to individuals who interact with the system. Users have the ability to create accounts by providing personal details, verify their identities, and configure security settings like two-factor authentication.
- Attributes: It stores and manages user authentication and authorization details.
- Operations: Users can create their accounts, update their profile information, and set preferences related to their user experience within the system. It plays a pivotal role in ensuring secure access and personalized interactions for system users.

3. Customer:

- Subset of User, representing those users who utilize the car rental service as customers.
- Attributes: This class contains additional attributes and functionalities tailored to customer-specific interactions, such as activating GPS service and viewing current contract(s). Payment methods are linked to facilitate seamless financial transactions.
- Operation: Customer can create their own account

4. Employee:

- Subtype of User, represents the workforce within the car rental company. Employees have distinct privileges by accessing into EmployeePortal
- Operation: n/a

5. PaymentInfo:

- Deals with the financial aspect of user interactions. This class ensures secure and convenient payment processing when users make reservations or settle rental fees within the system.
- Attributes: It holds details about users' payment methods, such as credit card numbers, expiration dates, and security codes.
- Operations; n/a

6. CustomerPortal:

- The user interface tailored for customers. It provides a user-friendly platform for customers to access the car rental system.
- Attribute: Customer can access list of car and their rental history

- Operation: Within this portal, customers can browse available cars, make reservations, and manage their rental history, creating a seamless and user-centric experience.

7. EmployeePortal:

- EmployeePortal serves as the interface designed for employees of the car rental company. It offers the tools and functionalities necessary for managing various aspects of the car rental operations.
- Attribute: Employee can access list of car, list of customers and their specific rental history
- Operation: Employees can review customer information, monitor the availability of cars, and update car statuses directly through this portal.

8. RentalInfo:

- Focuses on recording and managing rental transactions within the system.
- Attributes: it includes details such as car information, customer who rents the car, rental status, and rental agreements.
- Users can create new rentals and access their rental history, enhancing transparency and tracking within the car rental system. And employee can access them.

9. Car:

- Car class represents individual vehicles available within the car rental system. Cars are essential components of the rental process, and this class facilitates car management, including availability tracking and scheduling maintenance when needed.
- Attributes: It includes attributes like car make, model, year, if it's available or if service is needed.

Development plan and timeline

Partitioning of tasks

Johnny Rosas - System Overview, software architecture diagram explanation

Jenny Tran - Description of classes, attributes and operations

Patrick Simpauco - UML Diagram and Software Architecture Diagram

