PHP Special Project

Car Rental Application

Document Specifications

Version 1.23.31

Georgealdly Boursiquot

12-Jun-17

Contents

1. Introduction

1.1 System Purpose

The purpose of this specification is to define the requirements and developments of the Car Rental PHP Application. This application will handle a business needs of renting out vehicles to customers, maintaining records and data on vehicle fleet, operating the customer portal website, and reporting the state of the system to the company.

1.2 System Scope

The functional scope of the system is represented in four different aspects of the system: Customer Service Module (CS), Web Portal Module (WP), Fleet Management Module (FM) and Reports and Analytics Module (RA).

1.2.1 The scope of the Fleet Management module is keeping track of the rental vehicle fleet. This module's purpose will be to contain the data on the vehicle fleet and information about the specific vehicles. The Fleet Management module will not handle the renting of the vehicles.

1.2.2 The scope of the Customer Web Portal is to rent vehicles to customers online in the absence of an employee. The module will interface with the fleet management module but will not perform any of that module's duties. The same goes for the Customer Service module.

1.2.3 The scope of the Customer Service Module is to provide a clear and easy to use layout for employees and customers to follow along with as they work out a rental. As mentioned above the module will interface with the fleet management module but will not perform any of that module's duties.

1.2.4 The scope of the Reports and Analytics Module is to provide a predetermined spot that will contain all of the reports made from employees on car rentals. It will help the store managers and corporate office when they need to file reports and check on rentals.

You will not be provided with any support infrastructure for this project. You will design and implement all the classes needed to complete the Car Rental PHP Application. Although, help will be provided for any SQL related questions.

More details about the application are provided in the Specifications section.

2. System Overview

2.1 System Context

The system should be designed with four modules as mentioned before. These modules are the customer service module, the web portal module, the fleet management module, and the reports and analytics module. These four modules will make up the structure of the system. The customer service module will be the part of the system that is supposed to provide the employee with everything they need to perform their duties. The web portal module will be the part of the system that handles the website where the customers will be able to go through an automated process to rent cars. The fleet management module will do as its name suggests and manage the rental fleet. The reports and analytics module will be the part of the system that will be used to generate reports from the remaining three modules for the purpose organizing the data on the status of the company.

2.2 System Functions

The system will be able to rent out vehicles. The system will keep track of the rental fleet. The system will generate reports for the employees.

2.3 User Characteristics

The users will be customers which can include traveling business people, out of town visitors, and local residents in need of a car. Other users of the system will be the employees. The employees will all be using windows desktops to conduct their business on the system but the system will need to accommodate the variety of devices that the customers will have. The customers will have mobile devices running several different operating software including android, iOS, and blackberry. The full website also must accommodate these different devices.

You must design a solution to the Car Rental Application. The design will reflect everything you've learned during this Block HTML5, CSS3, JAVASCRIPT, JQUERY, PHP, OOP and MVC. You are expected to design your implementation using Object Oriented Programming, design your own classes for representing the basic entities the program manipulates. You are also expected to design your database in order to implement your solution. Using the MVC design pattern is a must.

3. Requirements

3.1 Pages (Customers)

1. Home Page
    - Display Cars, brief description of the site
2. Cars
    - Available cars by type
    - Search
3. FAQ
    - Frequently Asked Question
4. Contact
    - Contact page
5. Car
    - Show car's information
6. Rent
    - Rent process of a car
7. Registration
    - Client's registration page
8. Login / Logout / Forgot Password
    - Log in and Log out

3.2 Pages (Employees)

1. Dashboard
    - Display Cars
    - Stats on system
        i. How many cars in system
        ii. How many clients in system
        iii. How many rented car in system
2. Cars
    - CRUD to manipulates Cars
    - Search
3. Rents
    - CRUD to manipulates Rents
    - Search
4. Returns
    - CRUD to manipulate the returning of a vehicle
5. Clients
    - CRUD to manipulates Clients
    - Search
6. Invoices
    - View Invoices
7. Login / Logout / Forgot Password

3.3 Pages (Admin)

1. Everything Employees have
2. Reports
    - Cars
        i. How many cars in system
        ii. Car history
        iii. Current Car State
    - Rent
        i. Cars to be rented
        ii. Cars to be returned
    - Clients
        i. How many clients in system
        Client's history

4. Entities Properties

4.1 Employee

PROPERTY	DATA TYPE (SIZE)	CONSTRAINTS	DESCRIPTION
ID	INT	Primary Key	Employees ID
Username	VARCHAR(10)	NOT NULL	Employee's username
Password	VARCHAR(32)	NOT NULL	Employee's password
Full Name	VARCHAR(50)	NOT NULL	Employee's full name
Email	VARCHAR(100)	NOT NULL	Employee's email
Level	INT	NOT NULL	Employee's level
Status	INT	NOT NULL	Employee's status

4.2 Customer

PROPERTY	DATA TYPE (SIZE)	CONSTRAINTS	DESCRIPTION
ID	INT	Primary Key	Customer's ID
Username	VARCHAR(10)	NOT NULL	Customer's username
Password	VARCHAR(32)	NOT NULL	Customer's password
Full Name	VARCHAR(50)	NOT NULL	Customer's full name
Date of Birth	DATE	NOT NULL	Customer's date of birth
Phone	VARCHAR(200)	NOT NULL	Customer's phone number
Email	VARCHAR(100)	NOT NULL	Customer's email
Address	VARCHAR(255)	NOT NULL	Customer's address
Driver #	VARCHAR(20)	NOT NULL	Customer's Driver's licence
Credit Card #	VARCHAR(20)	NOT NULL	Customer's Credit Card Number

4.3 Car

PROPERTY	DATA TYPE (SIZE)	CONSTRAINTS	DESCRIPTION
ID	INT	Primary Key	Car's ID
Brand	VARCHAR(108)	NOT NULL	Car's Brand (Toyota, Ford)
Model	VARCHAR(108)	NOT NULL	Car's Model (Matrix, Escape)
Type	VARCHAR(108)	NOT NULL	Car's Type (Sedan, Van, Truck)
Tank Capacity	VARCHAR(108)	NOT NULL	Car's Tank Capacity (70 Litters)
Gas Consumption	VARCHAR(108)	NOT NULL	Car's Gas Consumption (7 L/100Km)
Color	VARCHAR(108)	NOT NULL	Car's Color
Number of Passenger	INT	NOT NULL	How many people can fit in the car
Rent Price	DOUBLE	NOT NULL	How much rent is per day

Image	TEXT	NULL	Image or Images of the car
Description	TEXT	NULL	Car's description
Status	INT	NOT NULL	Employee's status

4.4 Rental

PROPERTY	DATA TYPE (SIZE)	CONSTRAINTS	DESCRIPTION
ID	INT	Primary Key	Rent ID
Car ID	INT	NOT NULL	Car ID
Customer ID	INT	NOT NULL	Customer ID
Date Start	DATETIME	NOT NULL	Date rent starts
Date End	DATETIME	NOT NULL	Date rent ends
TOS Accepted	BOOLEAN	NULL	Has the customer accepted the TOS
Cancelled	BOOLEAN	NULL	Has the customer cancel the reservation
Inspected	BOOLEAN	NULL	Has the car been inspected
Notes	TEXT	NULL	Write any notes regarding the rent

4.4 Return Rental

PROPERTY	DATA TYPE (SIZE)	CONSTRAINTS	DESCRIPTION
ID	INT	Primary Key	Rent ID
Rental ID	INT	NOT NULL	Rental ID
Return Date	DATETIME	NOT NULL	The date the car was returned
Inspected	BOOLEAN	NOT NULL	Has the car been inspected
Damage	BOOLEAN	NOT NULL	Has the car been damaged
Notes	TEXT	NULL	Write any notes regarding the rent
Gas Level	Double	NOT NULL	Percentage of Gas level
Millage	DOUBKE	NOT NULL	The retuned Millage

4.4 Invoice

PROPERTY	DATA TYPE (SIZE)	CONSTRAINTS	DESCRIPTION
ID	INT	Primary Key	Rent ID
Return ID	INT	NOT NULL	Return ID
Charge	DOUBE	NOT NULL	Total Rent Charge
Additional Charge	DOUBLE	NOT NULL	In case of damages

5. Functional Requirements

5.1: The System will contain a Customer Service module that will allow Store and Corporate employees to provide information to customers

5.2 The System will contain a Customer Service module that will allow Store and Corporate employees' access to the system for the purpose of Creating "Rental Agreements"

5.3: The System will contain a Customer Portal Website that will provide information to the public and customers about the company and operations of the company.

5.4: This information will include Locations, Hours of operation, contact information, available rental vehicles, rental packages, price estimate, and any other information deemed necessary by the company.

5.5: The System will generate an inventory capacity report for the desired time.

5.6: The System will allow for new vehicles to be added to the inventory of the company.

5.7: The System will contain a Report and Analytics Module that will serve the purpose of generating reports on the operations of the company.

5.8: The System will contain a Report and Analytics Module that will be capable of generating reports for items such as sales, vehicle history, operations and rentals.

5.9: The system will allow the customer to input the desired date options to determine vehicle availability. If vehicle is available then the system will move onto the next step. If vehicle is not available then the system will prompt the customer to pick another vehicle and/or date.

5.10: The system will display only vehicles that are currently available to rent to the customer during the rental process.

5.11: The system will create a customer account if the customer is not in the system.

5.12: The customer account will require name, address, date of birth, driver's License, and credit card number.

5.13: The system will provide a report of the vehicles that are scheduled to be rented out and returned for the specified day.

5.14: at time of pickup the system will allow the employee to bring up the specified reservation for verification purpose.

5.15: The system will require the employee to verify the information of the Rental and to then input into the system a notice that the agreement has been gone over and accepted.

5.16: The customer will have twenty four (24) hours before pickup time to call and cancel the reservation else it will be charged automatically

5.17: At time of return of rental vehicle the system will allow the employee inspect the vehicle. And mark if it was damaged or not.

5.18 The "vehicle inspection form" will require information including damage to the vehicle, level of fuel in the gas tank, mileage, and a section for any other issues.

5.19: After the vehicle inspection is done then the system will show a final invoice which can be printed for the costumer to sign. This invoice will contain the total amount charged by the rental company including primary rental and additional charges.

5.20 The system will provide employees with a login.

5.21 The system will provide employees with a logout.

5.22 The system will provide employees with the option to reset / change their login password.

5.23: The system will allow employees with quick an efficient access to customer information.

5.24: The system will provide employees with a screen view of all customer pickups for a specified day.

5.25: The system will provide employees with a screen view of an existing reservation of a customer.

5.26: The system will provide customers to login on the web portal and view existing reservation in the system.

5.27: The system will allow employees to check vehicle availability from inventory for customers.

5.28: The system will allow employees to get specific vehicle information for customers.

6. Use Cases

6.1: Add Employee in System

Primary Actor	Administrator
Pre-Condition	System is running Admin is logged in
Post-Condition	A new employee has been added in the system
Main Success Scenario	1. Admin enters URL (mysite.com/new_employee.php) 2. System displays a new car form 3. Admin fills out the form properly 4. System asks for confirmation 5. Admin confirms 6. System confirms 7. System sends Login credentials to employee by mail 8. System returns back to Employees main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) Admin Cancels form 1. System returns to Employees main page 7b) Admin deny confirmation 1. Return to MSS # 3

6.2: Edit Employee in System

Primary Actor	Administrator
Pre-Condition	System is running Admin is logged in Employee exists in system
Post-Condition	An employee has been edited in the system
Main Success Scenario	1. Admin enters URL (mysite.com/employee.php?id=123) 2. System displays prefilled employee form with employee's info 3. Admin fills out / edits the form properly 4. System asks for confirmation 5. Admin confirms 6. System confirms 7. System returns back to Employees main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) Admin Cancels form 1. System returns to Employees main page 7b) Admin deny confirmation 1. Return to MSS # 3

6.3: Delete Employee in System

Primary Actor	Administrator
Pre-Condition	System is running Admin is logged in Employee exists in system
Post-Condition	An employee has been deleted in the system
Main Success Scenario	1. Admin enters URL (mysite.com/employee.php?id=123) 2. System displays prefilled employee form with employee's info 3. Admin clicks on delete 4. System asks for confirmation 5. Admin confirms 6. System confirms 7. System returns back to Employees main page
Extensions	5c) Admin Cancels form 2. System returns to Employees main page 7b) Admin deny confirmation 2. Return to MSS # 3

6.4: Display All employees

Primary Actor	Admin
Pre-Condition	System is running Employee exists in system
Main Success Scenario	1. User enters URL (mysite.com/employee.php) 2. System displays all employees 3. User is able to go to edit page of the employee

6.5: User login

Primary Actor	Administrator / Employee / Clients
Pre-Condition	System is running No one is logged User exists in system
Post-Condition	User is logged in
Main Success Scenario	1. User enters URL (mysite.com/login.php) 2. System displays login form 3. User fills authentication 4. System return to Home Page
Extensions	3b) Authentication failed 1. System shows errors message 2. Return to MSS # 2 3c) User cancels login 1. System returns to Home Page 3d) It's the first time user is logging in (excepts an Admin) 1. System asks to change current password 2. User fills the form to change current password 3. System return to MSS # 4 3e) User forgets password 4. System asks for email 5. System send renew password email to User 6. System return to MSS # 4

6.6: User Change password

Primary Actor	Administrator / Employee / Clients
Pre-Condition	System is running No one is logged User exists
Post-Condition	Password has been changed
Main Success Scenario	1. User enters URL (mysite.com/change_password.php) 2. System displays change password form 3. User fills forms 4. System return to Login Page
Extensions	3b) Password mismatched 1. System shows errors message 2. Return to MSS # 2 3c) User cancels form 2. System returns to Home Page

6.7: Add Car in System

Primary Actor	Administrator / Employee
Pre-Condition	System is running User is logged in
Post-Condition	A new car has been added in the system
Main Success Scenario	1. User enters URL (mysite.com/new_car.php) 2. System displays a new car form 3. User fills out the form properly 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Cars main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) User Cancels form 1. System returns to Cars main page 7b) User deny confirmation 1. Return to MSS # 3

6.8: Edit Car in System

Primary Actor	Administrator / Employee
Pre-Condition	System is running User is logged in Car exists in system
Post-Condition	A car has been edited in the system
Main Success Scenario	1. User enters URL (mysite.com/car.php?id=123) 2. System displays prefilled car form with car's info 3. User fills out / edits the form properly 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Cars main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) User Cancels form 1. System returns to Cars main page 7b) User deny confirmation 1. Return to MSS # 3

6.9: Delete Car in System

Primary Actor	Administrator
Pre-Condition	System is running User is logged in Car exists in system
Post-Condition	A car has been edited in the system
Main Success Scenario	1. User enters URL (mysite.com/car.php?id=123) 2. System displays prefilled car form with car's info 3. User clicks on delete 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Cars main page
Extensions	5c) User Cancels form 1. System returns to Cars main page 7b) User deny confirmation 1. Return to MSS # 3

6.10: Display All Cars

Primary Actor	All
Pre-Condition	System is running Car exists in system
Main Success Scenario	1. User enters URL (mysite.com/cars.php) 2. System displays all the cars

6.11: Display Single Car

Primary Actor	All
Pre-Condition	System is running Car exists in system
Main Success Scenario	1. User enters URL (mysite.com/cars.php?id=123) 2. System displays all the info of the car 3. User is able to go to rent page from here

6.12: Add Customer in System

Primary Actor	Administrator / Employee
Pre-Condition	System is running User is logged in
Post-Condition	A new customer has been added in the system
Main Success Scenario	1. User enters URL (mysite.com/new_customer.php) 2. System displays a new customer form 3. User fills out the form properly 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Customer main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) User Cancels form 1. System returns to Customer main page 7b) User deny confirmation 1. Return to MSS # 3

6.13: Edit Customer in System

Primary Actor	Administrator / Employee / Customer
Pre-Condition	System is running User is logged in Customer exists
Post-Condition	A customer has been edited in the system
Main Success Scenario	1. User enters URL (mysite.com/ customer.php?id=123) 2. System displays prefilled customer form with customer's info 3. User fills out / edits the form properly 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Cars main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) User Cancels form 1. System returns to Customer main page 7b) User deny confirmation 1. Return to MSS # 3

6.14: Delete Customer in System

Primary Actor	Administrator
Pre-Condition	System is running User is logged in Customer exists
Post-Condition	A customer has been edited in the system
Main Success Scenario	1. Admin enters URL (mysite.com/customer.php?id=123) 2. System displays prefilled customer form with customer's info 3. Admin clicks on delete 4. System asks for confirmation 5. Admin confirms 6. System confirms 7. System returns back to Customer main page
Extensions	5c) Admin Cancels form 1. System returns to Customer main page 7b) Admin deny confirmation 1. Return to MSS # 3

6.15: Display All Customers

Primary Actor	Administrator / Employee
Pre-Condition	System is running Customer exists in system
Main Success Scenario	1. User enters URL (mysite.com/customers.php) 2. System displays all the customers

6.16: Display Single Customer

Primary Actor	Administrator / Employee
Pre-Condition	System is running Customer exists in system
Main Success Scenario	1. User enters URL (mysite.com/customer.php?id=123) 2. System displays all the info of the customer 3. User is able to go to rent page from here with the customer id

6.17: Add new Rental in System

Primary Actor	Administrator / Employee / Customer
Pre-Condition	System is running User is logged in
Post-Condition	A car rental has been added in the system
Main Success Scenario	1. User enters URL (mysite.com/rental.php) 2. System displays a new car form 3. User fills out the form properly 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Cars main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) User Cancels form 1. System returns to Cars main page 7b) User deny confirmation 1. Return to MSS # 3
Errors	1. Past date 2. Car already taken 3. Date range, no more than 1 week

6.18: Edit Rental in System

Primary Actor	Administrator / Employee
Pre-Condition	System is running Employee is logged in Rental must exist in system
Post-Condition	A rental has been edited in the system
Main Success Scenario	1. User enters URL (mysite.com/rental.php?id=123) 2. System displays prefilled car form with cars info 3. User fills out / edits the form properly 4. System asks for confirmation 5. User confirms 6. System confirms 7. System returns back to Cars main page
Extensions	3b) Form Field Error 1. System shows errors message 2. System displays form with errors 3. Return to MSS # 3 with information 5c) User Cancels form 1. System returns to Cars main page 7b) User deny confirmation 1. Return to MSS # 3

6.19: Display All Rentals

Primary Actor	Administrator / Employee
Pre-Condition	System is running Rental exists in system
Main Success Scenario	1. User enters URL (mysite.com/rentals.php) 2. System displays all the rental

6.20: Display Single Customer

Primary Actor	Administrator / Employee
Pre-Condition	System is running Rental exists in system
Main Success Scenario	1. User enters URL (mysite.com/rental.php?id=123) 2. System displays all the info of the rental 3. User is able to go to return page from here with the rental id

6.21: Return Car in System

Primary Actor	Administrator / Employee
Pre-Condition	System is running Employee is logged in Rental must exist in system
Post-Condition	A car has been returned in the system
Main Success Scenario	1. User enters URL (mysite.com/return.php) 2. System asks for rental number 3. User enters rental number 4. System displays rental form with cars info 5. User clicks on return 6. System display the return form 7. User fills out the form properly 8. System asks for confirmation 9. User confirms 10. System confirms 11. System display invoice to be printed
Extensions	3b) Unknown rental number 1. System shows errors message 2. Return to MSS # 3 with information 3c) User Cancels form 2. System returns to Cars main page 7b) Form Field Error 1. System shows errors message 2. System displays form with errors Return to MSS # 3 with information

	9b) User deny confirmation 2. Return to MSS # 3 11b) After printing user is able to go back to cars page

6.22: Display All Invoices

Primary Actor	Admin
Pre-Condition	System is running Invoice exists in system
Main Success Scenario	1. User enters URL (mysite.com/invoices.php) 2. System displays all the invoices

6.23: Display All Car Statistics

Primary Actor	All
Pre-Condition	System is running Cars exists in system
Main Success Scenario	1. User enters URL (mysite.com/cars_stats.php) 2. System displays all the info of the car with the history

6.23: Display All Client Statistics

Primary Actor	All
Pre-Condition	System is running Clients exists in system
Main Success Scenario	3. User enters URL (mysite.com/customers_stats.php) 4. System displays all the info of the customers with their history

7. Correction Grid

Code –80%

Each use case is worth something; I'll write down the weighs later on

Design – 10%

Since you're able to use a template to start, I'll only be checking if all the pages exist, and the implementation of the pages

Teacher's Appreciation – 10%

- User Friendly
- Design
- Comments
- Teacher's appreciation