Database Systems

Fall 2023

Final Project
Car Rental System

Farah Alaa El Din Khattab - 7487
Mariam Mohamed Samy - 7695
Nour Mohamed Khamis Saad - 7597
Rana Mohamed Abo El Amayem - 7697

1. Abstract

The Car Rental System is a comprehensive solution designed to automate the process of renting vehicles to customers. Aimed at streamlining operations for car rental agencies, the system provides a robust platform for vehicle management, customer handling, and transaction processing. With features such as car registration, customer profile management, real-time availability checking, and reservation management, the system significantly enhances efficiency and customer service. The system is scalable and adaptable, catering to small local agencies as well as larger international franchises with multiple branches. The purpose of this project is to provide a user-friendly, efficient, and reliable system that reduces manual work, minimizes errors, and offers a seamless rental experience. The main features include detailed vehicle tracking, customer database management, booking and reservation logistics, and financial transaction handling.

2. Introduction

The advent of technology has revolutionized many service-oriented sectors, with the car rental industry being no exception. Traditional car rental processes are often cumbersome, error-prone, and time-consuming, involving extensive paperwork and manual coordination. There is a pressing need for a Car Rental System that not only expedites the renting process but also provides a transparent and reliable platform for both the service providers and the customers.

The Car Rental System is designed to address these challenges by offering a centralized database for managing vehicles and customers, an intuitive interface for reservations and rentals, and automated processes for tracking and reporting. The objectives of the system include improving vehicle utilization, enhancing customer satisfaction, reducing operational costs, and increasing the overall efficiency of the rental process. The scope of the system extends from vehicle entry and maintenance to handling customer queries, processing payments, and generating detailed reports for administrative use. By integrating advanced functionalities and ensuring scalability, the Car Rental System aims to be a comprehensive solution for the modern car rental industry, accommodating future expansions and technological advancements.

3. System Overview

The Car Rental System is an integrated platform designed to facilitate the entire process of renting vehicles to customers, encompassing a range of functionalities from vehicle management to customer service and transaction processing. At its core, the system aims to provide a seamless, efficient, and user-friendly experience for both the rental agency staff and the customers.

3.1. Main Components and Functionalities

1. **Vehicle Management**
   - Allows the agency (admin) to register new vehicles into the system, including details like make, model, year, and current status (available, rented, out of service).
   - It also facilitates the tracking of each vehicle's rental history, maintenance records, and overall usage.
2. **Customer Management**
   - Manages customer profiles including personal details, rental history, and payment methods.
   - It provides functionalities for customer registration (sign up) and login.
3. **Reservation and Booking**
   - Enables customers to search for available cars based on specific criteria make reservations, and proceed to rental agreements.
   - It includes features for modifying or canceling bookings.
4. **Pricing and Payments:**
   - Automates the calculation of rental charges based on rental duration.
   - It handles transactions securely, assuming that there is only one payment method: payment by credit card/visa .
5. **Reporting and Analytics:**
   - Generates comprehensive reports for business insights, including vehicle usage statistics, customer activity, financial summaries, and performance metrics.

3.2. System Architecture and Technology Stack

The Car Rental System is built using a straightforward and robust technology stack, focusing on widely-used tools and platforms for web development. This approach ensures reliability, ease of use, and a wide support community. Below is an overview of the architecture and technologies employed:

Front-End:
- **CSS with Bootstrap Templates**: The user interface leverages CSS for styling and layout, with Bootstrap templates providing a responsive and modern design. Bootstrap's grid system and pre-built components enable quick and consistent UI development across different screens and devices.
- **Native PHP/HTML/JavaScript**: The client-side logic and structure are handled using native PHP, HTML, and JavaScript. This traditional web development approach ensures broad compatibility and straightforward implementation. HTML structures the content, PHP dynamically generates web pages, and JavaScript enhances interactivity and user experience.

Back-End:

- **Native PHP**: The server-side application logic is written in native PHP, a popular scripting language known for its simplicity and effectiveness in web development. PHP scripts handle data processing, business logic, and interaction with the database, generating dynamic content based on user requests.

Database:

- **MySQL**: The system uses MySQL as its relational database management system (RDBMS). MySQL stores all data related to vehicles, customers, rentals, transactions, and more. It provides robust data management capabilities, ensuring data integrity, security, and efficient access.

Server:

- **XAMPP:** XAMPP is an easy-to-install Apache distribution containing MariaDB (MySQL), PHP, and Perl. It is used as a local development environment and server solution. In this project, XAMPP provides the necessary tools to run the application locally for development, testing, and demonstration purposes.
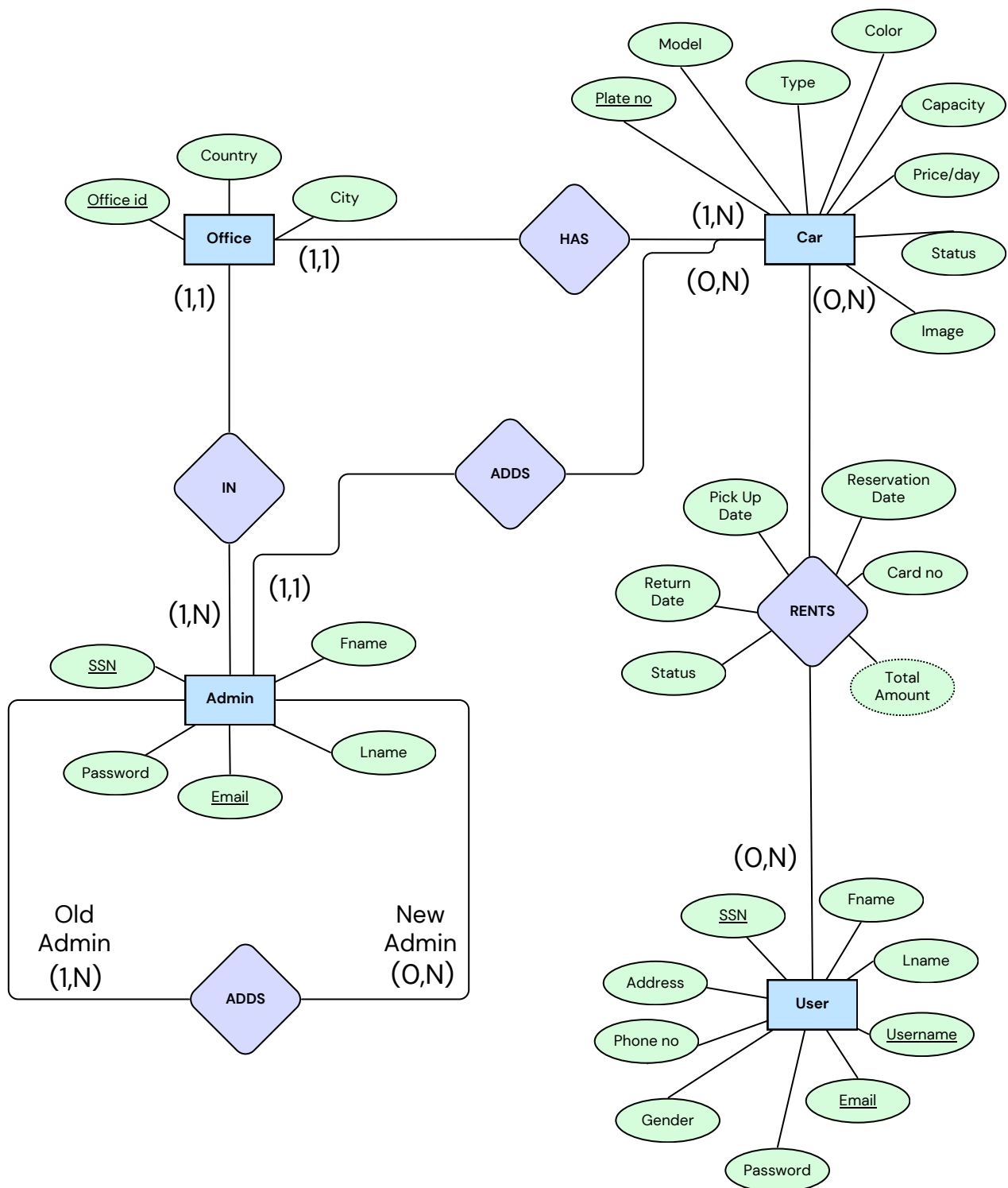
Development

- **Local Development:** The entire system is developed and tested locally using the XAMPP package, which includes an Apache HTTP server, MySQL database, and support for PHP and Perl programming languages. This setup simulates a real server environment on a local machine, making development and debugging more convenient.

This architecture and technology stack provide a solid foundation for the Car Rental System, ensuring it is accessible, maintainable, and scalable. The use of widely-adopted technologies like PHP, MySQL, and Bootstrap means that the system can be easily understood and modified by developers with standard web development skills. It also ensures that the system can be hosted on a wide range of platforms and environments, providing flexibility in deployment and operations.

4. System Design
4.1. ERD

Office entity attributes: Office id, Country, City

Car entity attributes: Plate no, Model, Type, Color, Capacity, Price/day, Status, Image

Admin entity attributes: SSN, Fname, Lname, Email, Password

RENTS relationship attributes: Pick Up Date, Reservation Date, Return Date, Card no, Status, Total Amount

User entity attributes: SSN, Fname, Lname, Username, Email, Address, Phone no, Gender, Password

Office — HAS — Car: Office (1,1), Car (1,N)

Office — IN — Admin: Office (1,1), Admin (1,N)

Admin — ADDS — Car: Admin (1,1), Car (O,N)

Car — RENTS — User: Car (O,N), User (O,N)

Admin — ADDS — Admin: Old Admin (1,N), New Admin (O,N)

4.2. Relational Schema

car

plate_no	model	type	year	color	capacity	price_per_day	status	office_id	image

user

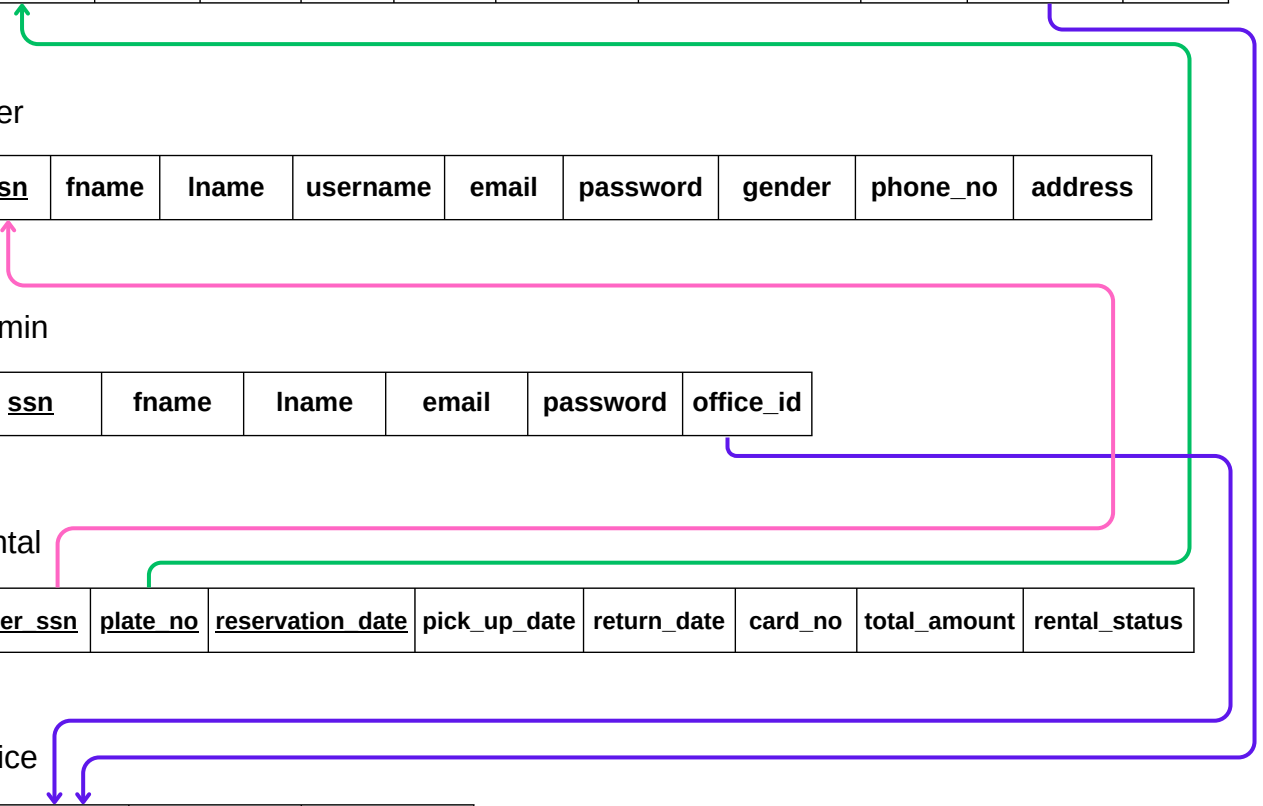| ssn	fname	lname	username	email	password	gender	phone_no	address

admin

ssn	fname	lname	email	password	office_id

rental

user_ssn	plate_no	reservation_date	pick_up_date	return_date	card_no	total_amount	rental_status

office

office_id	country	city

5. Assumptions

- **Offices:** The system has default offices (6 offices) which are entered in the initial queries and CANNOT be edited or altered.

office_id	country	city
1	Egypt	Alexandria
2	USA	New York City
3	Egypt	Cairo
4	Canada	Toronto
5	Egypt	Hurghada
6	USA	Chicago

- Initial System Setup: The Car Rental System is pre-configured with a primary administrative account and an initial office location to ensure immediate operational capability upon installation. This initial setup is designed to provide a foundation for the system's configuration and use.
    - **Initial Admin Credentials:**
        - **Email:** mainadmin@gmail.com
        - **Password:** 123456789
    - **Initial Office Location (No.1):**
        - **Country:** Egypt
        - **City:** Alexandria
- All admins know the correct office id to the location while adding a new car.
- Car status is updated automatically by the system.
- Admins allowed operations:
    - Add new car.
    - Add new admin.
    - Update care info.
    - See all data and search through it.
    - Can't change users info.
- Payment is only electronic via Visa / credit card.
- Users can't reserve a car if it overlaps with another reservation.
- No refund if a rental was canceled

```sql
29
30  CREATE TABLE `admin` (
31      `ssn` int(11) NOT NULL,
32      `fname` varchar(255) DEFAULT NULL,
33      `lname` varchar(255) DEFAULT NULL,
34      `email` varchar(255) DEFAULT NULL,
35      `password` varchar(50) DEFAULT NULL,
36      `office_id` int(11) DEFAULT NULL
37  );
38
39  CREATE TABLE `car` (
40      `plate_no` varchar(20) NOT NULL,
41      `model` varchar(50) DEFAULT NULL,
42      `car_type` varchar(50) DEFAULT NULL,
43      `color` varchar(50) DEFAULT NULL,
44      `capacity` int(11) DEFAULT NULL,
45      `price_per_day` decimal(10,2) DEFAULT NULL,
46      `status` varchar(50) DEFAULT NULL,
47      `office_id` int(11) DEFAULT NULL,
48      `image` varchar(400) DEFAULT NULL
49  );
50  CREATE TABLE `office` (
51      `office_id` int(11) NOT NULL,
52      `country` varchar(50) DEFAULT NULL,
53      `city` varchar(50) DEFAULT NULL
54  );
55
56  CREATE TABLE `rental` (
57      `user_ssn` int(11) NOT NULL,
58      `plate_no` varchar(8) NOT NULL,
59      `reservation_date` date NOT NULL,
60      `pick_up_date` date DEFAULT NULL,
61      `return_date` date DEFAULT NULL,
62      `card_no` varchar(30) DEFAULT NULL,
63      `total_amount` decimal(10,2) DEFAULT NULL,
64      `status` varchar(255) DEFAULT NULL
65  );
66
```

```sql
CREATE TABLE `user` (
  `ssn` int(11) NOT NULL,
  `fname` varchar(255) DEFAULT NULL,
  `lname` varchar(255) DEFAULT NULL,
  `username` varchar(255) DEFAULT NULL,
  `email` varchar(255) DEFAULT NULL,
  `password` varchar(50) DEFAULT NULL,
  `gender` char(1) DEFAULT NULL,
  `phone_no` varchar(11) DEFAULT NULL,
  `address` varchar(255) DEFAULT NULL
);
```

```sql
ALTER TABLE `admin`
  ADD PRIMARY KEY (`ssn`),
  ADD KEY `office_id` (`office_id`);


--
ALTER TABLE `car`
  ADD PRIMARY KEY (`plate_no`),
  ADD KEY `office_id` (`office_id`);


ALTER TABLE `office`
  ADD PRIMARY KEY (`office_id`);

ALTER TABLE `rental`
  ADD PRIMARY KEY (`user_ssn`,`plate_no`,`reservation_date`),
  ADD KEY `fk_rental_car` (`plate_no`);


ALTER TABLE `user`
  ADD PRIMARY KEY (`ssn`);


ALTER TABLE `office`
  MODIFY `office_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;


ALTER TABLE `admin`
  ADD CONSTRAINT `admin_ibfk_1` FOREIGN KEY (`office_id`) REFERENCES `office` (`office_id`);


ALTER TABLE `car`
  ADD CONSTRAINT `car_ibfk_1` FOREIGN KEY (`office_id`) REFERENCES `office` (`office_id`);


ALTER TABLE `rental`
  ADD CONSTRAINT `fk_rental_car` FOREIGN KEY (`plate_no`) REFERENCES `car` (`plate_no`),
  ADD CONSTRAINT `fk_rental_user` FOREIGN KEY (`user_ssn`) REFERENCES `user` (`ssn`);
```