CSE311L Database Management System

Section 9

Spring 2020

North South University

Submitted To: Rifat Ahmed Hassan (Rah1)

Sajid Ahmed

Project Title: Uber Insider

Name	Id	Group
Farhan Hasin Saad	1821493642	D
Raisa Nusrat Chowdhury	1821640642	D
Anjoom Nur	1821258642	D

Introduction : Uber Insider management is a database management system that stores personal data of the driver and owner for better control of their company. The system will allow registered owner and drivers to record official information in the database. An admin can view any driver's record using the system and also view the customer's report of any drivers.

Context:

The traditional paper-based system has some flaws. A manager might lose or damage the information or cannot monitor the driver's information, owner's information when it is needed. In this project we tried to make a webapplication for the administrator panel,drivers and customers of Uber. It is a user friendly webpage which will help the all the users to view and monitor their data more effieciently in a quick and easy way.

Administrator panel is in control of everything , Admin can login or register and add,edit or delete information according to the company's necessity. Driver who is registered in database can login with his id and password and view how much he has earned and how much he has to pay to the Uber authority. We also have an option that is used by a customer to file report against the ride he/she has chosen by mentioning the drivers Id or license plate. This webpage contains details of each car,car owner and driver.

Technology stacks :
1. MySQL and XAMPP : we have made "uber_insider" database and the tables in the database using MySQL
2. HTML: The frontend of the webpage is structured using HTML.
3. CSS: Styling elements have been added to the homepage using CSS.
4. PHP: We have used PHP in the backend so that our webpage connects with the database and functions smoothly.

Front-End Details: Entering the website, first we'll land into the login page which has 3 panels

1. **Employee Login:** If we login as employee (which refers to the admin panel) we'll get to the home page. Home page shows us the total number of drivers, Total earning of the month and Uber revenue of the month. This statistical information is dynamic. It will update automatically as we add more data. On the upper middle we see 4 options Driver List, Owner List, Car List and logout. If we click on the Driver List it'll show us all the driver's details that are registered in the database. Admin can add new drivers or edit driver details or delete drivers from the database using the interface. Driver must have an owner registered in the database for which we added owner id on the top of the driver registration form. We added a dropdown menu so we don't have to remember their name manually anymore. If the owner is not registered we added an owner button n the right corner which would take us to the owner registration form. Owner List has all the details about the owner and an edit/delete button on each row to update or drop the information. This page has an add button which takes us to the owner registration page to add owners to the database. When we register in the driver where we have option of Car no, car name and car color which will automatically add into Car list.

2. **Driver Login:** Drivers who are already registered can login using the provided id from admin panel and their phone number as their password. Drivers can not register on their own, an admin must add them into the database on admin panel

3. **Complaint Box:** Customer can insert phone no of the driver, Car No and the complain details to file a complaint. There is option of customer name and phone no but which is not mandatory. A notification will go to the admin with the exact time the customer has filed the complaint.

Most importantly our Login System is session handled and each page is session handled.

Back-End Details:

1. We used aggregate functions for the home page details.

Count function: To determine the total number of the drivers register in the company. From this and admin can easily know how many drivers are working for the company and if the company needs any more drivers or not

Sum function: Sum function can easily find the total income of the drivers in a month and also find the uber contribution of all the drivers which will help the company to know about their monthly income of a year. Also this function automatically count the each driver's uber contribution of a month

2. Driver Personal information display on the profile

3. Owner Personal information display on the profile

4. Driver register facility

5. Driver information updates and deletes facility

6. Owner information registers facility

7. Driver login processing after the registration

8. View each complain from the customer report box

9. And delete facility after the issue is solved

We provided server-side algorithms of user-facing web application elements. We used PHP to create logic which helped our web applications functions correctly. For example: Driver registration process, owner registration process etc.

Development:

The complete development time can be divided in four phases. In each phase, we focused on a particular target and completed it step by step.
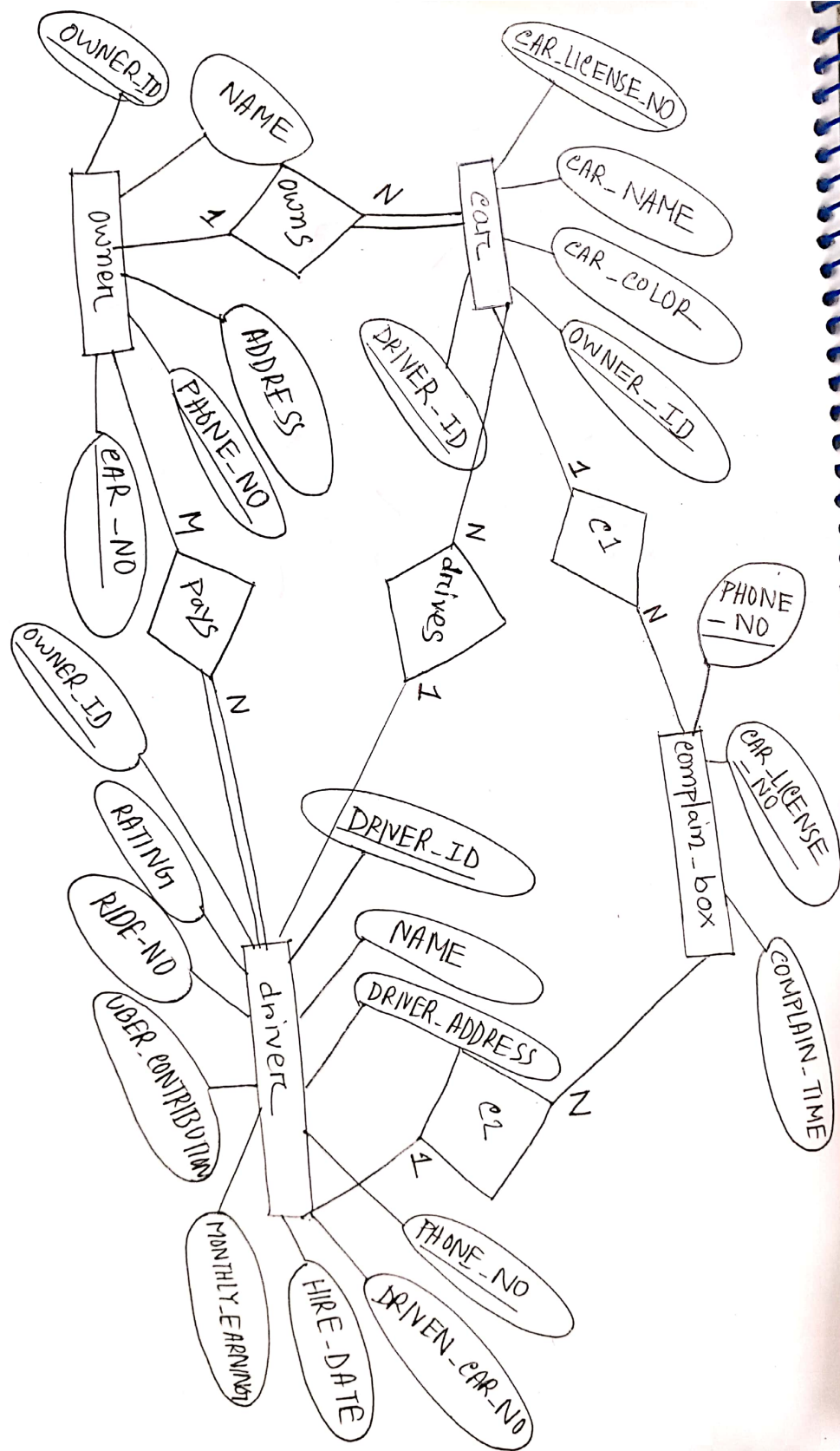
Phase 1: In this phase, we designed the ER diagram and Database Schema. The first prototype was later scrapped and a new optimized schema was developed in the projected time.

Phase 2: In this phase, the front end structure was designed. The pages we would be using in order to implement the idea of this project were developed using HTML and styled using CSS. In some of the pages, we started using Php like login pages.

Phase 3: This phase is focused on backend development. We used Php to completely develop the backend side and MySQL to connect database and perform query.

Phase 4: This was the testing phase. The system was running fluently and performing desired functions. Some minor error was found and solved

ER Diagram: (Shown in the next page)

ER Diagram

Entities and attributes:

owner: OWNER_ID, NAME, ADDRESS, PHONE_NO, CAR_NO

car: CAR_LICENSE_NO, CAR_NAME, CAR_COLOR, OWNER_ID, DRIVER_ID

driver: DRIVER_ID, OWNER_ID, RATING, RIDE_NO, UBER_CONTRIBUTION, MONTHLY_EARNING, HIRE_DATE, DRIVEN_CAR_NO, PHONE_NO, NAME, DRIVER_ADDRESS

complain_box: PHONE_NO, CAR_LICENSE_NO, COMPLAIN_TIME

Relationships:

owner — owns (1 : N) — car

owner — pays (M : N) — driver

car — drives (N : 1) — driver

car — c1 (1 : N) — complain_box

driver — c2 (N : 1) — complain_box

Benefits: Company can efficiently monitor their workers. The Company gets an insight on how much they are earning and their profits. Customer feedback helps them to take action against the driver and helps to improve the system.

Future plan: While working on the project, we found that the project can be extended with more functionality. It can be connected with driver current location while driving. There can be more administrative controls. The search function can be used here for better controls.