

GURU NANAK DEV UNIVERSITY, AMRITSAR

Software Training Report

On

ZoomCar

Submitted in the partial fulfillment of the requirement for the award of degree of

Bachelors of Technology

In

COMPUTER SCIENCE & ENGINEERING

Batch

(2013-2017)



Submitted to

Ms. Kiranbir Kaur

Ms. Priyanka Mahajan

Submitted By-

Samridhi

(2013CSA1136)

DEPARTMENT OF COMPUTER ENGINEERING AND TECHNOLOGY

DECLARATION

I , **Samridhi** student of B.tech 8th Sem. CSE 2013-2017, hereby declare that six months industrial training report entitled “ZOOMCAR” submitted in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Computer Science & Engineering in our original work and has not been submitted for the award in any other university or institution.

Ms. Kiranbir Kaur (Guide)

Ms Priyanka Mahajan (Co-Guide)

Samridhi

(2013CSA1136)

ACKNOWLEDGEMENT

This project is an acknowledgement in itself to the intensity, drive and competence of many individuals, who have contributed to it. I list here only a few of them. One individual is Mr. **Amrinder Pal Singh**, of V.M.M Education, Amritsar. I am extremely thankful to him for devoting his valuable time and imparting knowledge to us. With his valuable guidance, the course up to the completion of software became smooth without much cared hurdles. He was always friendly and encouraging during the development of project.

I express thanks and gratitude to **Mr. Sandeep Sharma**, H.O.D computer science department, for his encouraging support and guidance in carrying out the project

I thank **Ms. Kiranbir Kaur** , Project guide, for providing me with an excellent project and guiding me in completing our project successfully.

I pay special thanks to all the staff members of the Department of Computer Science for their help at every stage. They helped in every respect during the tenure..

Lastly, I thank my parents and friends for their moral support in every sphere. Their vital push infused sense of insurgency in me, I am thankful to them for their assistance and cooperation.

Samridhi

Table Of Contents

<u>Sr No.</u>	TOPIC	Page No.
1.	Organization Profile	4
2.	Introduction To Project	6
3.	Module Description	8
4	Front End	9
5.	Back End	11
6.	Hardware & Software Requirements	12
7.	Software Tools To Be Used	13
8.	ER Diagrams	14
9.	Data Flow Diagrams	17
10.	Testing	21
11.	Screen Shots	22
12.	Suggestions & Recommendations	39
13.	Bibliography	40

Company Profile

VMM Education

VMM Education's journey started in January 2005 with a vision of bringing computer education of global standard to the holy city of Amritsar. Today seven years later, VMM Education or VMM, as it is popularly known, is the largest and the most trusted computer centre of the region, with annual turn out of more than 1000 students.

It specializes in six weeks training in Java, Microsoft .NET and PHP; six months training in IOS, Android, PHP, Python and Advanced Java, while keeping the syllabus up to date with current industry standards. We have managed to successfully train over seven thousand engineers who are currently working in Global Multinationals like TCS, Tech Mahindra, Infosys, CSC or pursuing higher studies in world-class Institutions.

VMM is today the favourite choice among students of various engineering colleges for pursuing their six months or six weeks industrial training. A unique "industry-endorsed curriculum," crafted by professionals of VMM which enhances the job-readiness and employability of learners and equips them for the IT Industry. We offer air-conditioned labs and classrooms that are equipped with the latest hardware and software technologies required for the training.

To provide IT education which can match with the global IT standards, VMM also undertakes industrial projects from UK & USA under the banner of Venus Software Solutions like im4schools.co.uk and more. This allows the students to work on live projects and make projects for the industry. Some of our products include Point of sale software for Super Markets with barcode reader support, Finger print attendance Management System that works for schools, colleges and other institutes, Remote LAN Controller which is used to view remote desktops on LAN or WAN.

Problem Analysis

- People or for example University students have to go somewhere but are not having any conveyance .
- They have to depend on their friends or the public transport, which can be troublesome at times.
- And there are people of the city itself who have spare vehicles at home and are willing to rent them out for some hours or days at their convenience .
- Thus a platform is required where the people who want to rent a self driving car and people who can provide their cars on rent , such a platform is Zoomcar.

Project Plan

Introduction to Project

A project that is totally based on giving the car/bikes on rent for self- drive to anywhere in the town or outside the city. Giving a short introduction to the project -it just simplifies the car rentals so that the people who are coming from different countries, states or cities can book their own car for their ease to travel any place they want to. Adding more to it, we have seen an application of zoom-car which exhibits the same criteria followed by this project but this project not only includes car but also the bikes which would be provided to the clients. The front end of the project is Advance Java and the back end would be MySQL. There would be 3 interfaces –Admin Interface, Vehicle owner Interface, User Interface.

FEATURES OF EXSISTING PROJECT

- User can get a car on rent
- User can give a car on rent.
- Admin can add and delete the car categories.
- Admin can manage the all cars details under a particular category.
- User can go anywhere and can ride himself.
- Provide the service of e-wallet
- Provide the security to users while they are pooling.
- Provide the history of car booking at to user
- Provide the transaction details of both seeker and provider

Feasibility Study

LIMITATION OF EXSISTING PROJECT

- Do not provide the service of e-wallet.
- Do not provide the history of Booking.
- Can't differentiate between human n robot.
- Do not provide the details of both seeker and provider

OVERCOME THE LIMITATION

- Provide the service of e-wallet
- Provide the history of Booking.
- Provide the transaction details of both seeker and provider

E-WALLET SERVICE

A user can do payment via e-wallet services, if he failed to pay the cash. Respective money will be deducted from his account

Description of all the interfaces is as follows:

Admin Interface:

1. Login
2. Change/Recover Password
3. Manage Cars/Bikes
4. Update Additional information about vehicles (price range, description etc.)
5. Manage Merchants
6. View Bookings

Vehicle Owner Interface:

1. Registration / Login
2. Change/Recover Password
3. Manage Vehicles under specified categories
5. Vehicle owner can view bookings of his vehicles

User Interface:

1. Sign-up/Login
2. Change/Recover Password
3. Search vehicle based on category and city from where car is availed.
4. Book vehicle by providing specified date and time.
5. Payment can be done through online gateway or can be paid directly to the service center.

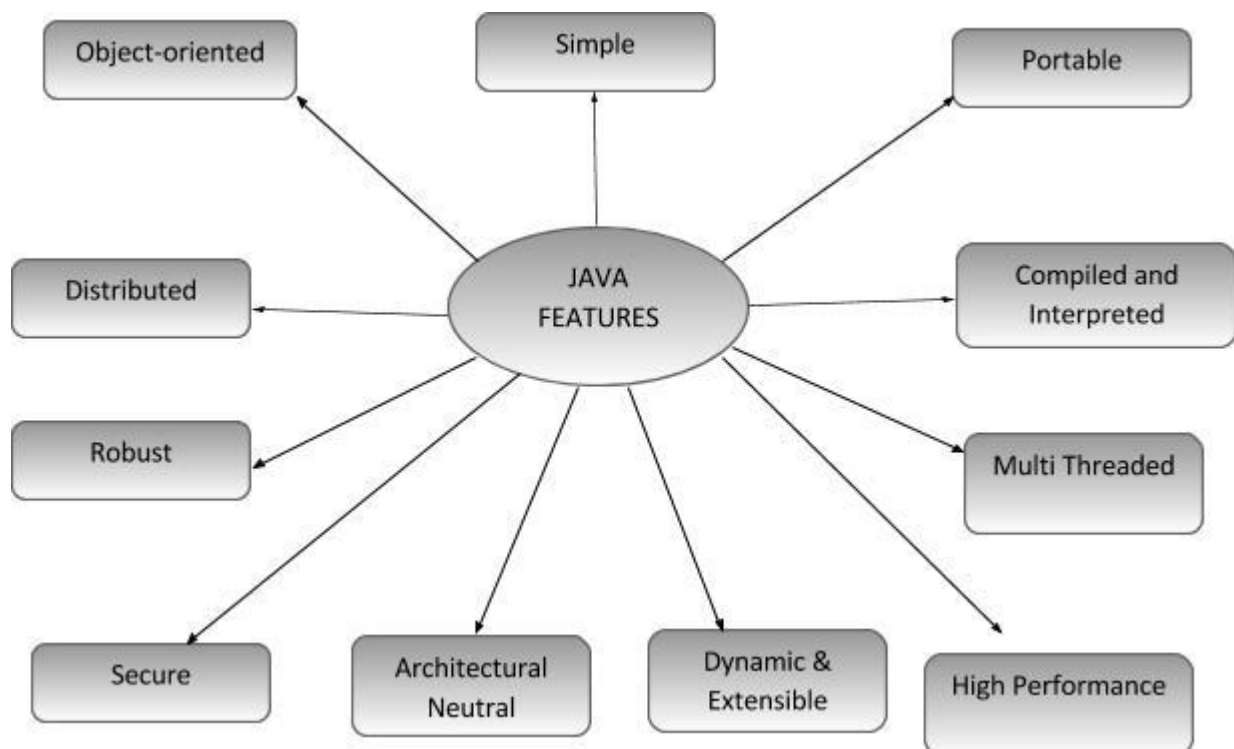
Technologies Used

Front End

1) Java:

Java is a general purpose and the most popular object-oriented programming language. Java was developed by James Gosling and his colleagues at Sun Microsystems in the early 1990's.

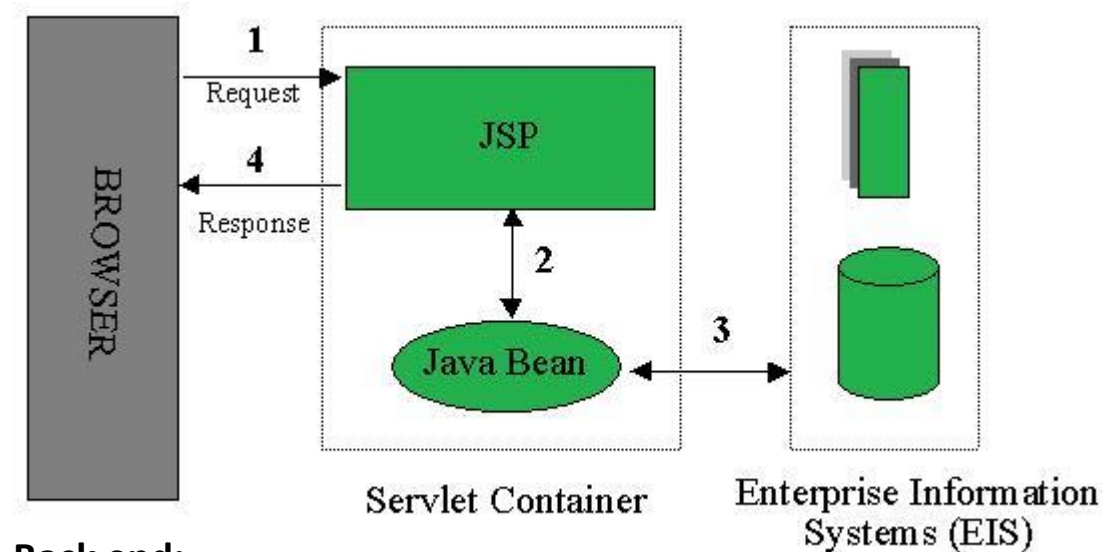
Due to its simplicity and easy to learn and advanced features, we opted this language for our six months industrial training. This language supports many interesting features that make it an ideal language for software development. In addition to the object-oriented features, it also provides features such as platform independence, security, multithreading, portability; etc. which makes it well suited for the web and networked services, applications, platform-independent desktops, robotics and any other embedded devices.



2) JSP (Java Server Pages)

JSP technology is used to create dynamic web applications. JSP pages are easier to maintain than a Servlet. JSP pages are opposite of Servlets as a servlet adds HTML code inside Java code, while JSP adds Java code inside HTML using JSP tags. Everything a Servlet can do, a JSP page can also do it.

JSP enables us to write HTML pages containing tags, inside which we can include powerful Java programs. Using JSP, one can easily separate Presentation and Business logic as a web designer can design and update JSP pages creating the presentation layer and a Java developer can write server-side complex computational code without concerning the web design. And both the layers can easily interact over HTTP requests.



Back end:



MySQL:

MySQL is an open source Relational Database Management System. MySQL is very fast, reliable, and flexible Database Management System. It provides a very high performance and is multi-threaded and multi-user Relational Database management system.

MySQL is one of the most popular relational database Management System on the web. The MySQL Database has become the world's most popular open source Database, because it is free and available on almost all the platforms. The MySQL can run on Unix, Windows, and Mac OS.

Features:

The following list describes some of the important Features of MySQL Database Software.

- Written in C and C++.
- Tested with a broad range of different compilers.
- Works on many different platforms.
- The server is available as a separate program for use in a client/server networked environment. It is also available as a library that can be embedded (linked) into standalone applications. Such applications can be used in isolation or in environments where no network is available.

Statements and Functions

- Full operator and function support in the SELECT and WHERE clauses of queries. For example:
- Full support for SQL GROUP BY and ORDER BY clauses. Support for group functions such as COUNT().
- Support for LEFT OUTER JOIN and RIGHT OUTER JOIN with both standard SQL and ODBC syntax.
- Security- A privilege and password system that is very flexible and secure, and that allows host-based verification. Passwords are secure because all password traffic is encrypted when you connect to a server.
- Scalability and Limits- Handles large databases. We use **MySQL** Server with databases that contain 50 million records. We also know of users who use **MySQL** Server with 60,000 tables and about 5,000,000,000 rows.
- Up to 64 indexes per table are allowed (32 before **MySQL** 4.1.2). Each index may consist of 1 to 16 columns or parts of columns. The maximum index width is 1000 bytes (500 before **MySQL** 4.1.2).
- Connectivity- Clients can connect to the **MySQL** server using TCP/IP sockets on any platform. On Windows systems in the NT family (NT, 2000, XP, or 2003), clients can connect using named pipes. On Unix systems, clients can connect using Unix domain socket files.
- In **MySQL** versions 4.1 and higher, Windows servers also support shared-memory connections if started with the --shared-memory option. Clients can connect through shared memory by using the --protocol=memory option.
- The Connector/J interface provides **MySQL** support for Java client programs that use JDBC connections. Clients can be run on Windows or Unix. Connector/J source is available.

Hardware/software Requirements:

HARDWARE REQUIREMENTS:

Hardware requirements include that hardware which is required for its working. It includes:

- Pentium 4 Computer
- 512 MB RAM
- High Speed Internet Connection

SOFTWARE REQUIREMENTS:

The technical specifications of requirements for the software are as follows:

- Operating System (Windows, Linux, MacOS)
- Java run time environment
- NetBeans (Java IDE with Android development plugins)
- Any Web Browser (Chrome, Firefox, etc.)

Tools to be Used:

1.JAVA DEVELOPMENT KIT (JDK)

The Java Development Kit (JDK) is a software package that includes all the basic components that makeup the java environment. These include the Java compiler, Java Interpreter, an applet viewer that lets you see applets without opening a Java-compatible web browser, Debugger, Class file disassembler, Header and Stub file generator and Document Generator.

2. NetBeans - INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)



NetBeans IDE is a free, open source, popular integrated development environment used by many developers. It provides built-in support for developing in Java, C, C++, XML, and HTML.

Minimum Hardware Configurations for Windows:

Microsoft Windows XP Professional SP3/Vista SP1/Windows 7 Professional:

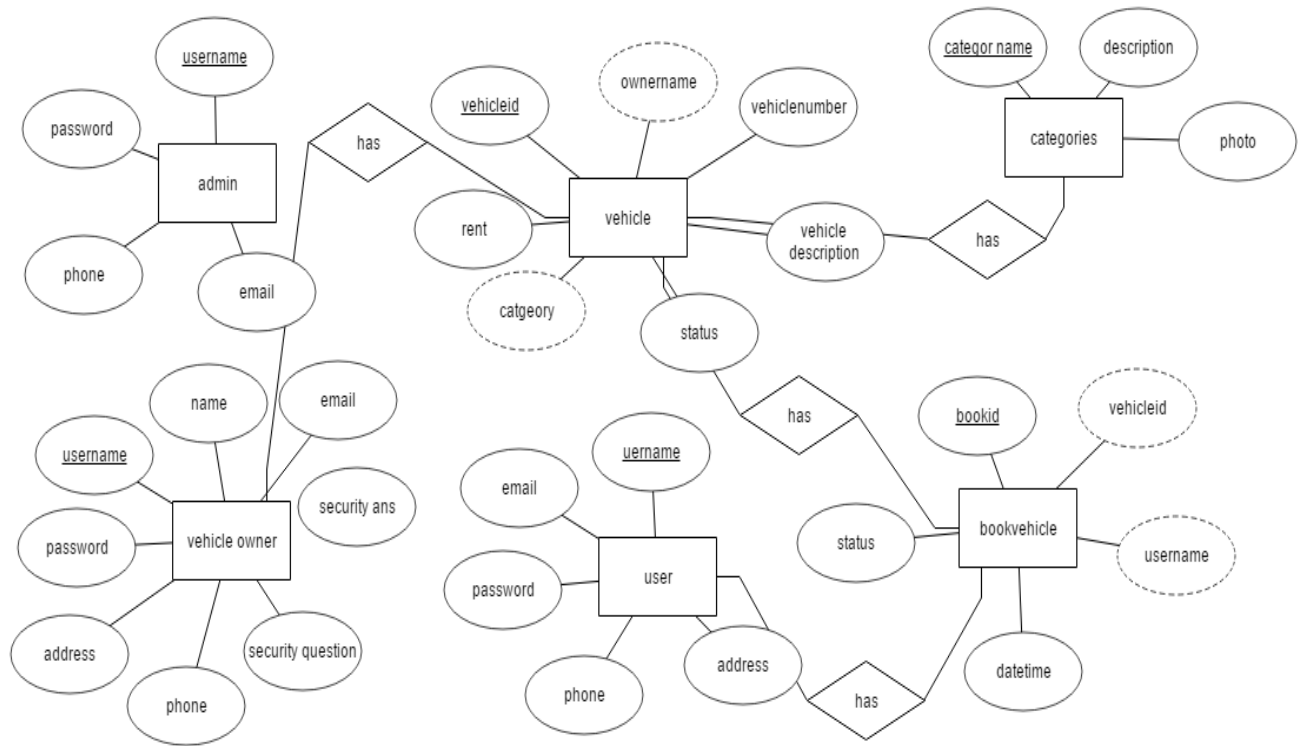
Processor: 800MHz Intel Pentium III or equivalent

Memory: 512 MB

Disk space: 750 MB of free disk space

Design

ER Diagram



Tables

Admin

Username	Varchar	Primary key
Password	Varchar	
Name	Varchar	
Email	Varchar	
Phoneno	Long	

User table

Username	Varchar	Primary key
Password	Varchar	
Email	Varchar	
Phoneno	Long	
Address	Varchar	

Categories

Catname	Varchar	Primary key
Description	Varchar	
Photo	Varchar	

Vehicle

Vehicleid	Int	Primary key
Catname	Varchar	Foreign key to categories
Ownername	Varchar	Foreign key to vehicle owner
Rent	Int	
Description	Varchar	
Status	Varchar	
Vehicle number	varchar	

Vehicle owner

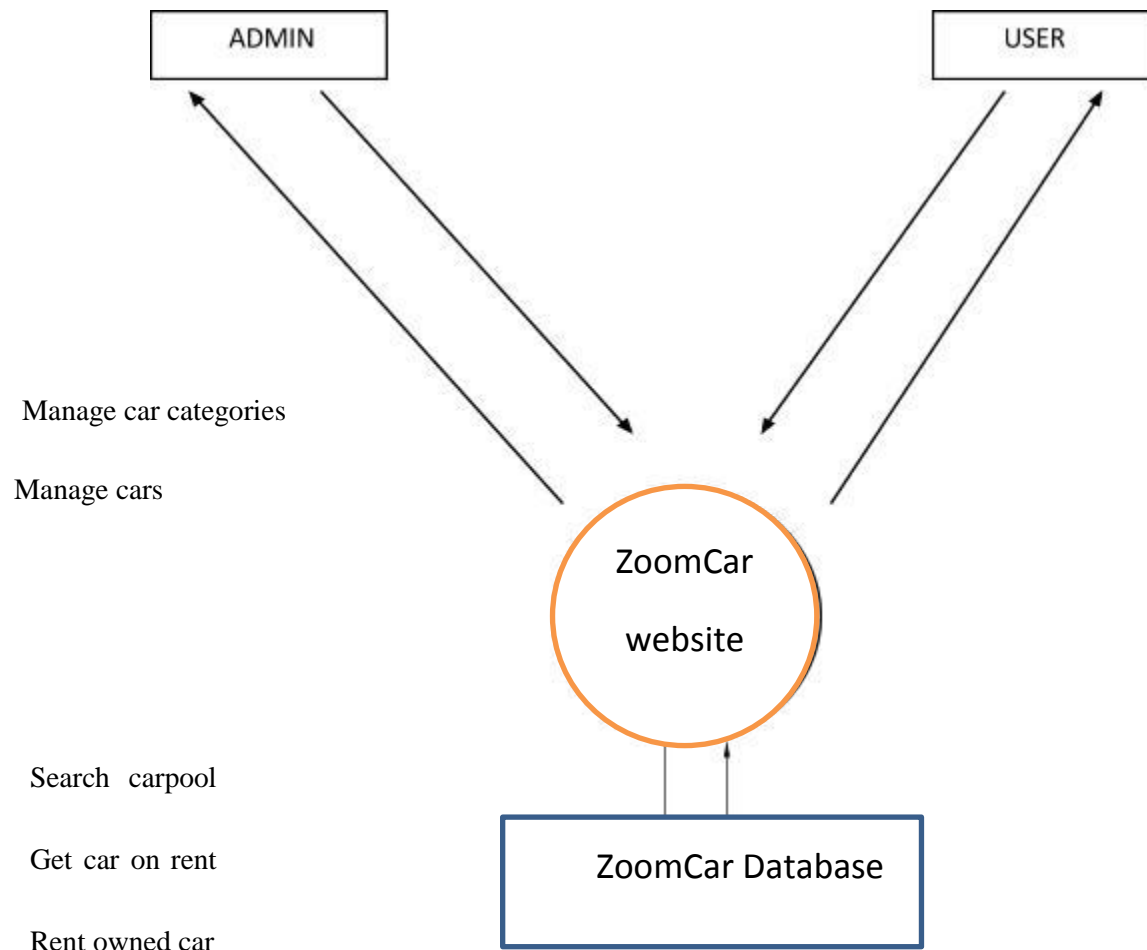
Username	Varchar	Primary key
Password	Varchar	
Name	Varchar	
Phone	Long	
Email	Varchar	
Security question	Varchar	
Security answer	Varchar	
Address	varchar	

Book table

Bookid	Int	Primary key
Username	Varchar	Foreign key to user
Vehicle id	Int	Foreign key to vehicle
Status	Varchar	
Datetime	Timestamp	

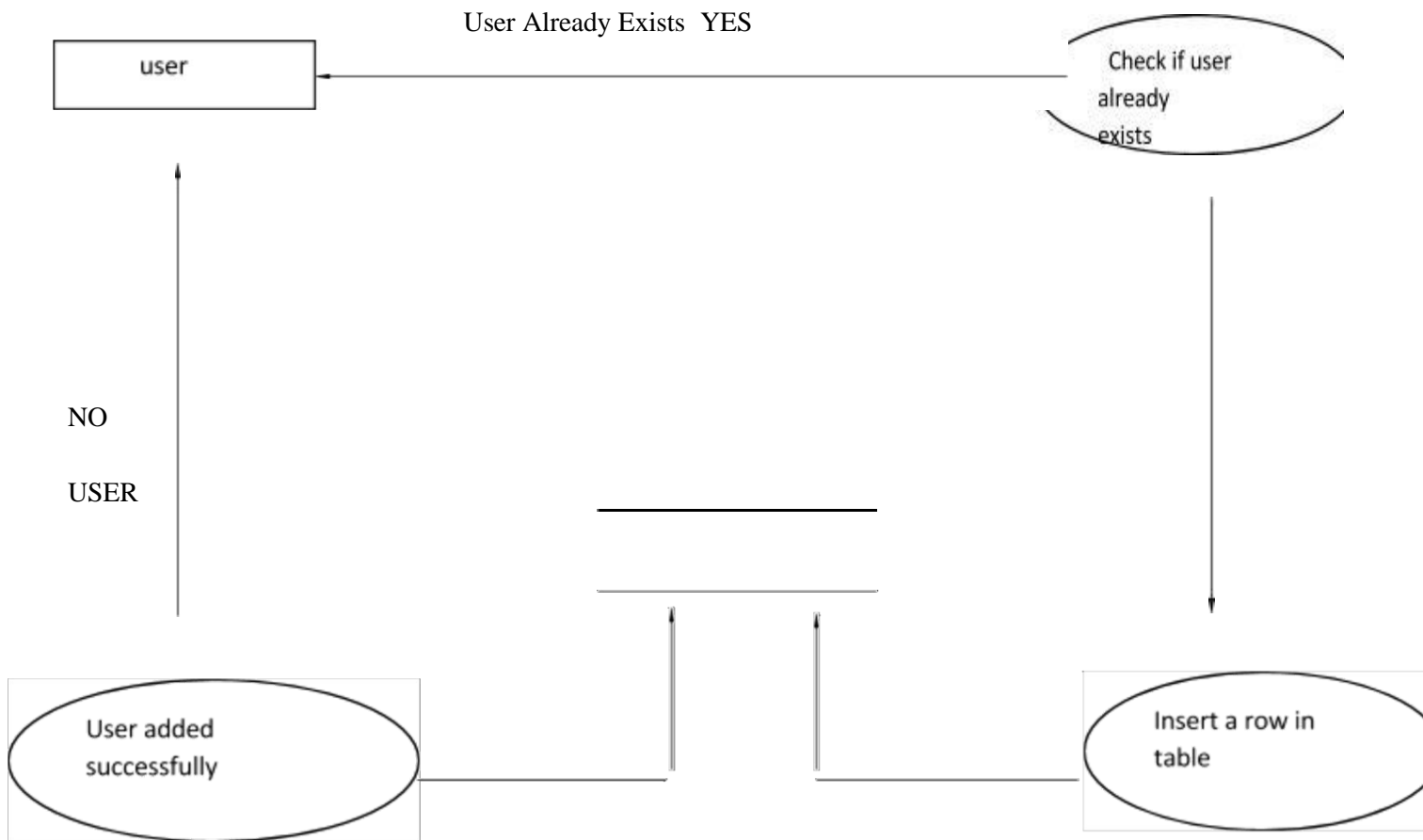
DATA FLOW DIAGRAMS

LEVEL 0 DFD



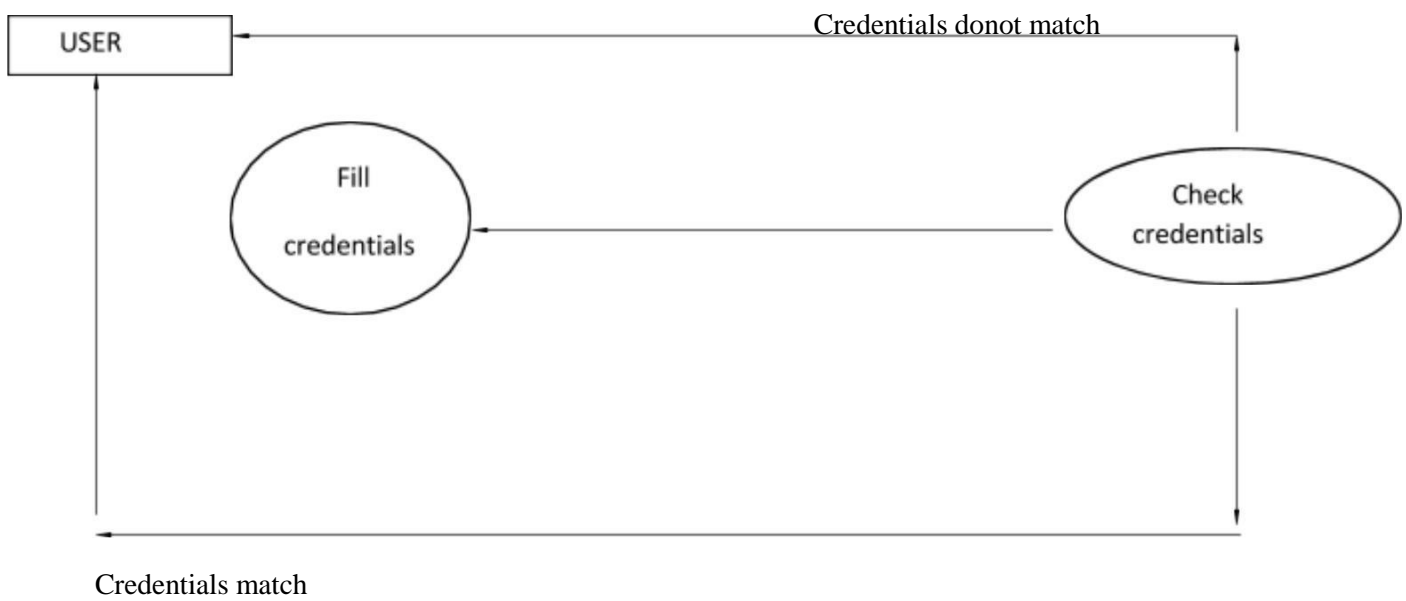
LEVEL 1 DFD

USER SIGNUP

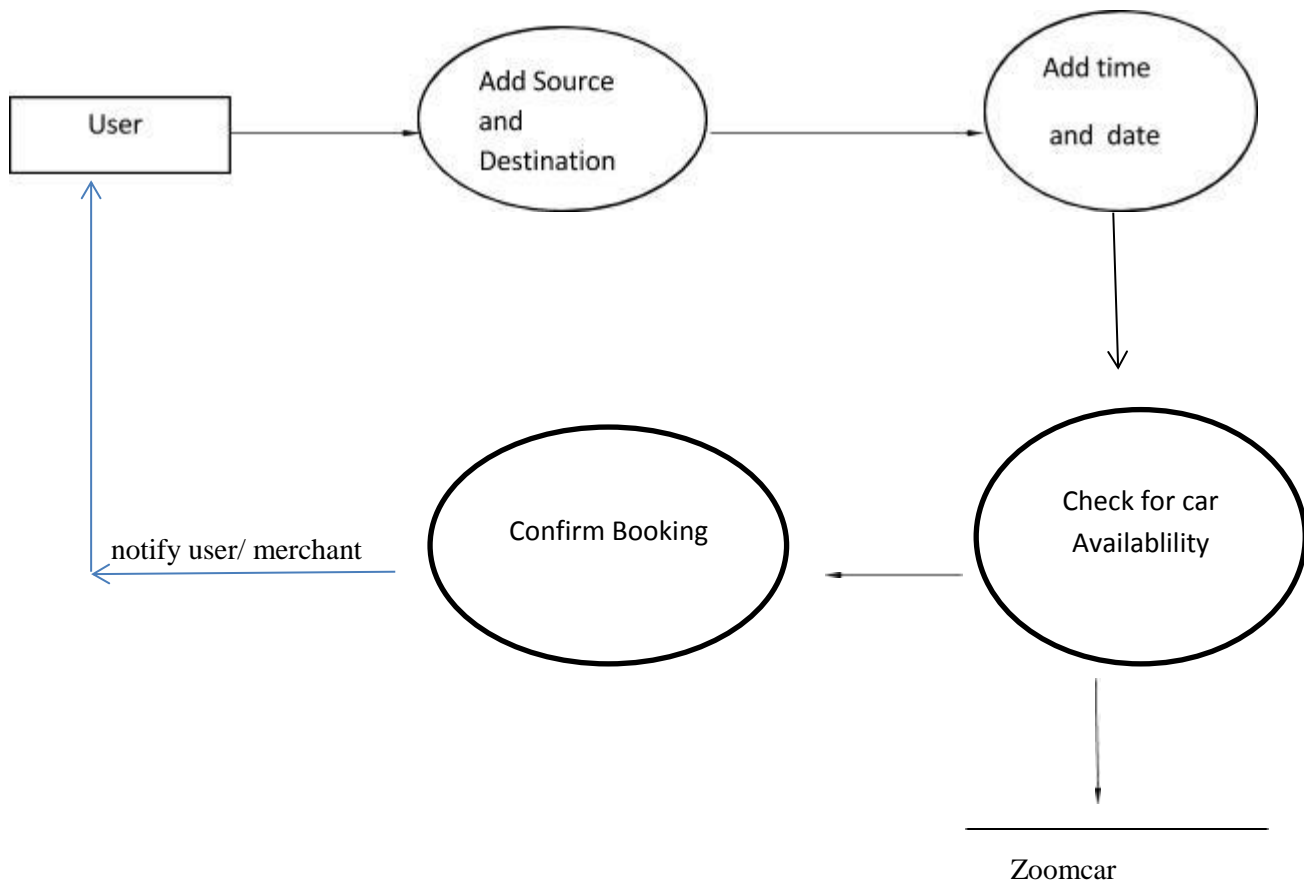


USER LOGIN

LEVEL 1 DFD



LEVEL 1 DFD



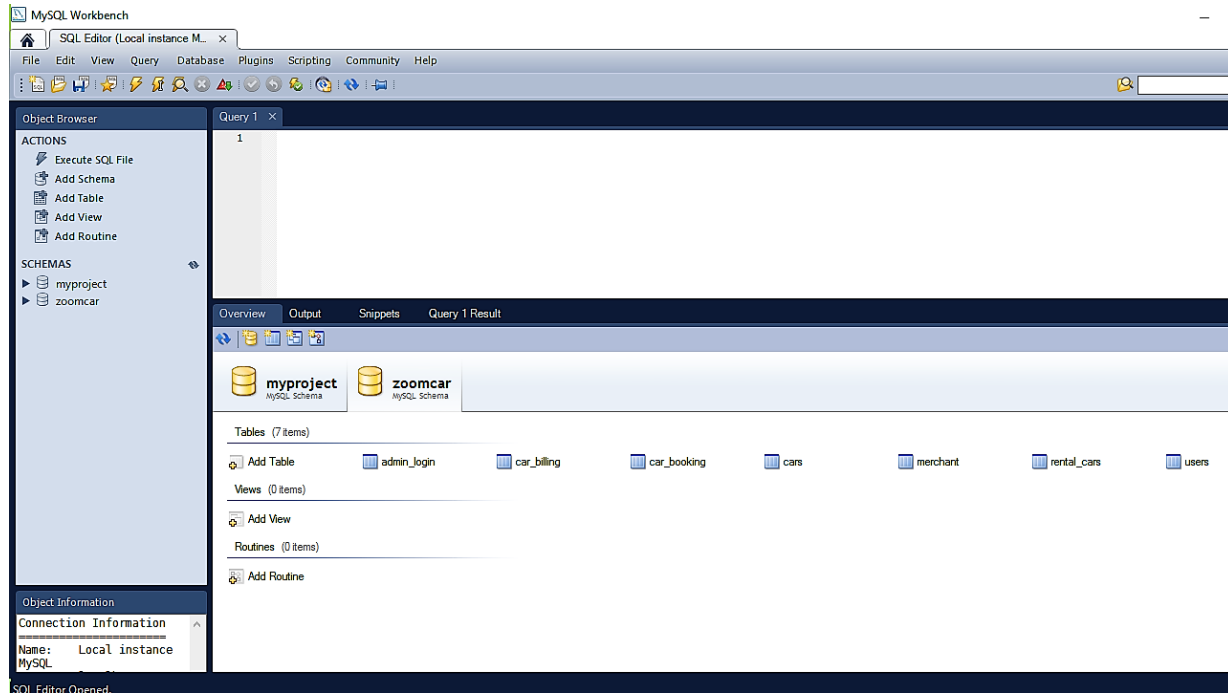
Testing

The checks for validations are:

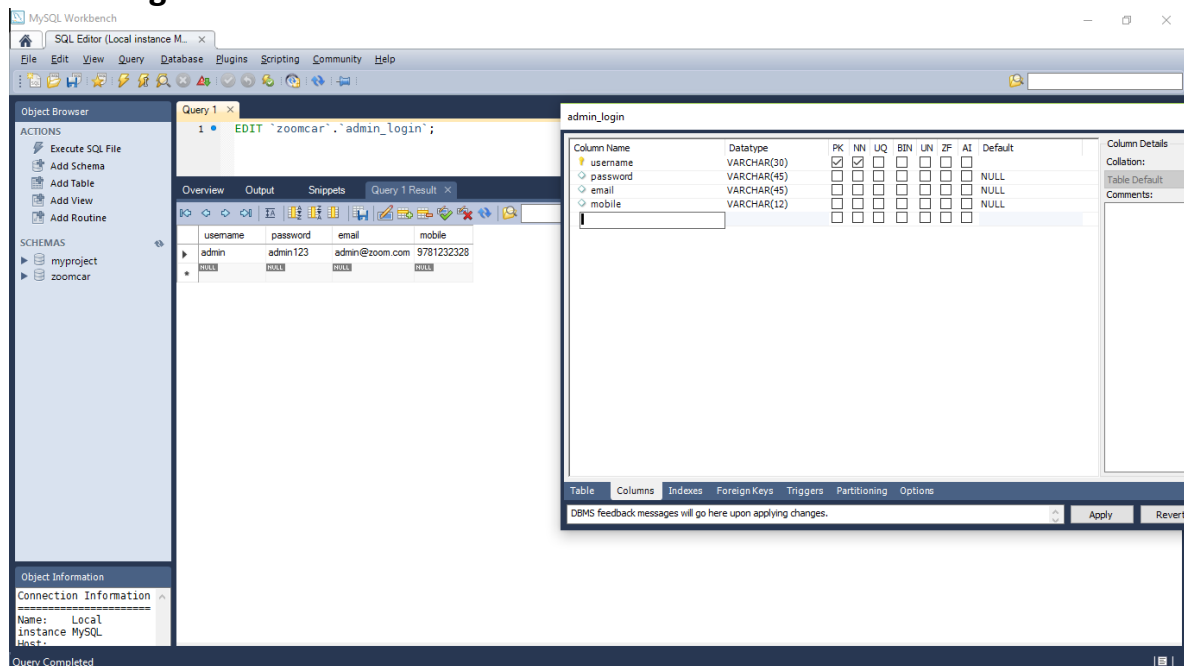
- A person book a car of a previous date that is past by.
- Person can enter the date of return of car to be less than the date of car booking
- The car is not booked if it is not available on a particular date
- Username and password checks are used.
- A merchant can't manage the cars/ bikes if he is blocked by the admin, or his status is not yet approved.

Screenshots

Database overview



Admin login



Vehicle Categories –Cars and bikes

MySQL Workbench

SQL Editor (Local instance M...)

File Edit View Query Database Plugins Scripting Community Help

Object Browser

ACTIONS

- Execute SQL File
- Add Schema
- Add Table
- Add View
- Add Routine

SCHEMAS

- myproject
- zoomcar

Query 1

```
1 • EDIT 'zoomcar'. 'cars';
```

Overview Output Snippets Query Result

Fetches 15 records. Duration: 0.000 sec, fetched in: 0.000 sec

car_name	maker	photo	num_seats	car_type	car_price	p_per_day	security	type
BALENO	MARUTI SUZUKI	/car_images/baleno.jpg	8	SUV	900000	4000	1000	4wheeler
BREEZA	MARUTI SUZUKI	/car_images/brezza.jpg	8	SUV	1100000	4500	11000	4wheeler
CRETA	HYUNDAI	/car_images/creta.jpg	5	SUV	1000000	3000	10000	4wheeler
HONDA ACTIVA 4G	HONDA	/car_images/honda_activa.jpg	2	SCOOTER	50000	500	500	2wheeler
HONDA CITY	HONDA	/car_images/hondacity.jpg	5	SEDAN	900000	3000	1000	4wheeler
MAHINDRA XUV	MAHINDRA	/car_images/mahindra_xuv.jpg	8	SUV	1500000	4500	15000	4wheeler
MARUTI SWIFT	MARUTI	/car_images/swift.jpg	5	HATCHBACK	500000	3000	1000	4wheeler
MARUTI SWIFT DZIRE	MARUTI SUZUKI	/car_images/dzire.jpg	5	SEDAN	600000	3500	1000	4wheeler
ROYAL ENFIELD CLASSIC 350	ROYAL ENFIELD	/car_images/royalfield_classic.jpg	2	BIKE	106000	1000	1060	2wheeler
ROYAL ENFIELD HIMALAYAN	ROYAL ENFIELD	/car_images/royalfield_himalayan.jpg	2	BIKE	150000	1500	1500	2wheeler
ROYAL ENFIELD THUNDERBIRD	ROYAL ENFIELD	/car_images/royalfield_thunderbird.jpg	2	BIKE	176000	1700	1760	2wheeler
SUZUKI ACCESS	SUZUKI	/car_images/suzuki_access.jpg	2	SCOOTER	50000	500	500	2wheeler
VERNA	HYUNDAI	/car_images/verna.jpg	5	SUV	500000	5000	1000	4wheeler
YAMAHA FASCINO	YAMAHA	/car_images/yamah_fascino.jpg	2	SCOOTER	55000	500	550	2wheeler
YAMAHA YZF R15	YAMAHA	/car_images/yamaha_yzf.jpg	2	BIKE	111000	1000	1110	2wheeler

Object Information

Connection Information

Name: Local instance MySQL

Host: localhost:3306

Query Completed

MySQL Workbench

SQL Editor (Local instance M...)

File Edit View Query Database Plugins Scripting Community Help

Object Browser

ACTIONS

- Execute SQL
- Add Schema
- Add Table
- Add View
- Add Routine

SCHEMAS

- myproject
- zoomcar

cars

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
car_name	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	*
maker	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
photo	VARCHAR(200)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
num_seats	VARCHAR(3)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
car_type	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
car_price	VARCHAR(10)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
p_per_day	VARCHAR(10)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
security	VARCHAR(10)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
type	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Column Details

Collation:

Table Default

Comments:

Table Columns Indexes Foreign Keys Triggers Partitioning Options

DBMS feedback messages will go here upon applying changes.

Apply Revert Close

Object Information

Connection Information

Name: Local instance MySQL

Host: localhost:3306

Query Completed

Merchant(The sellers)

The screenshot shows the MySQL Workbench interface. The 'Query 1 Result' tab displays the 'merchant' table data. The table has columns: username, password, name, email, phone, address, city, state, photo, status, and joiningdate. The data is as follows:

username	password	name	email	phone	address	city	state	photo	status	joiningdate
argu	123	Arju	argu@gmail.com	79759785	12, Queens Road	AMRITSAR	PUNJAB	./merchant_images/download (1).jpg	pending	2017-04-15
rdhi	123	Rishi	rdhi@gmail.com	97854321	121, Rags Garden	JALANDHAR	Punjab	./merchant_images/download (1).jpg	active	2017-04-12
sam	123	samrishi	samrishi@gmail.com	9781232328	120, Gita/Gobind Singh Nagar	AMRITSAR	PUNJAB	./merchant_images/download (1).jpg	pending	2017-05-21
samrishi	123	Samrishi	samrishi4chawla@gmail.com	987654321	120, Model Town	JALANDHAR	Punjab	./merchant_images/download.jpg	active	2017-04-11
sarv	123	sarv	sarv@gmail.com	313131211	34, Rankashap	LUDHIANA	PUNJAB	./merchant_images/download.jpg	pending	2017-04-15

The 'Column Details' pane on the right shows the table structure for 'merchant':

Column Name	Datatype	PK	NN	UQ	BN	UN	ZF	AI	Default
username	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
password	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
name	VARCHAR(100)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
email	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
phone	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
address	VARCHAR(150)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
city	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
state	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
photo	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
status	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
joiningdate	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Users

The screenshot shows the MySQL Workbench interface. The 'Query 1 Result' tab displays the 'users' table data. The table has columns: username, email, phone_no, and password. The data is as follows:

username	email	phone_no
sam	samrishi@gmail.com	9781232328
samrishi	samrishi94chawla@gmail.com	9781232328
samrishi_chawla	samrishi@gmail.com	9781232328
NULL	NULL	NULL

The 'Column Details' pane on the right shows the table structure for 'users':

Column Name	Datatype	PK	NN	UQ	BN	UN	ZF	AI	Default
username	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
email	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
phone_no	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
password	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Vehicles owned by merchants Available for booking – table rental_cars

The screenshot shows the MySQL Workbench interface. The 'Query 1' editor contains the SQL command: `EDIT 'zoomcar'.rental_cars;`. The 'Query 1 Result' pane displays the following data:

id	car_name	merchant_username	num_cars	merchant_city
4	BREEZA	samidhi	1	JALANDHAR
5	HONDA CITY	samidhi	1	JALANDHAR
6	MARUTI SWIFT	anju	1	AMRITSAR
7	MARUTI SWIFT DZIRE	anju	5	AMRITSAR
8	BALENO	sarv	3	LUDHIANA
9	MARUTI SWIFT	sarv	2	LUDHIANA
10	NULL	NULL	NULL	NULL
11	HONDA ACTIVA 4G	samidhi	1	JALANDHAR
12	YAMAHA YZF R15	samidhi	1	JALANDHAR
13	BREEZA	NULL	1	NULL
14	SUZUKI ACCESS	NULL	1	NULL
15	NULL	NULL	NULL	NULL

The 'rental_cars' table structure is shown in the right pane:

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
car_name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
merchant_username	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
num_cars	VARCHAR(3)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
merchant_city	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Car Booking table

The screenshot shows the MySQL Workbench interface. The 'Query 1' editor contains the SQL command: `EDIT 'zoomcar'.car_booking;`. The 'Query 1 Result' pane displays the following data:

booking_id	car_id	booking_date	user	p_per_day	bill_id
26	6	2017-04-01	samidhi	3000	1
27	6	2017-04-02	samidhi	3000	1
28	6	2017-04-03	samidhi	3000	1
29	6	2017-04-04	samidhi	3000	1
30	4	2017-05-13	samidhi	4500	2
31	4	2017-05-14	samidhi	4500	2
32	4	2017-05-15	samidhi	4500	2
33	4	2017-05-16	samidhi	4500	2
34	4	2017-05-17	samidhi	4500	2
35	4	2017-05-18	samidhi	4500	2
37	7	2017-04-01	samidhi	3500	4
38	7	2017-04-02	samidhi	3500	4
39	4	2017-04-26	samidhi	4500	5
40	4	2017-04-27	samidhi	4500	5
41	4	2017-04-28	samidhi	4500	5
42	4	2017-04-29	samidhi	4500	5
43	7	2017-05-23	samidhi	3500	6
44	7	2017-05-24	samidhi	3500	6
45	NULL	NULL	NULL	NULL	NULL

The 'car_booking' table structure is shown in the right pane:

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
booking_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
car_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
booking_date	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
user	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
p_per_day	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
bill_id	INT(11)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Car Billing Table

MySQL Workbench

SQL Editor (Local instance M...)

File Edit View Query Database Plugins Scripting Community Help

Object Browser

ACTIONS

- Execute SQL File
- Add Schema
- Add Table
- Add View
- Add Routine

SCHEMAS

- myproject
- zoomcar

Object Information

Connection Information

Name: Local
Instance: MySQL
Host:

Query 1

1 • EDIT 'zoomcar'. 'car_billing';

Overview Output Snippets Query 1 Result

Fetch 5 records. Duration: 0.000 sec, fetched in: 0.000 sec

bill_id	from_date	to_date	date_of_booking	date_of_billing	extra_charges	tot_amt	remarks	taxes	coupon_used	discount	net_amt
1	2017-04-01	2017-04-04	2017-04-21	2017-04-25	0	12000		0		0	12000
2	2017-05-13	2017-05-18	2017-04-21	2017-04-26	0	27000		300		0	27300
4	2017-04-01	2017-04-03	2017-04-25	2017-04-26	0	7000		0		0	7000
5	2017-04-26	2017-04-30	2017-04-26	2017-04-27	0	18000		0		0	18000
6	2017-05-23	2017-05-25	2017-05-21	2017-05-21	0	7000		0		0	7000
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

car_billing

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
bill_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
from_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
to_date	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
date_of_booking	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
date_of_billing	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
extra_charges	DOUBLE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
tot_amt	DOUBLE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
remarks	VARCHAR(200)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
taxes	DOUBLE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
coupon_used	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
discount	DOUBLE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
net_amt	DOUBLE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Column Details

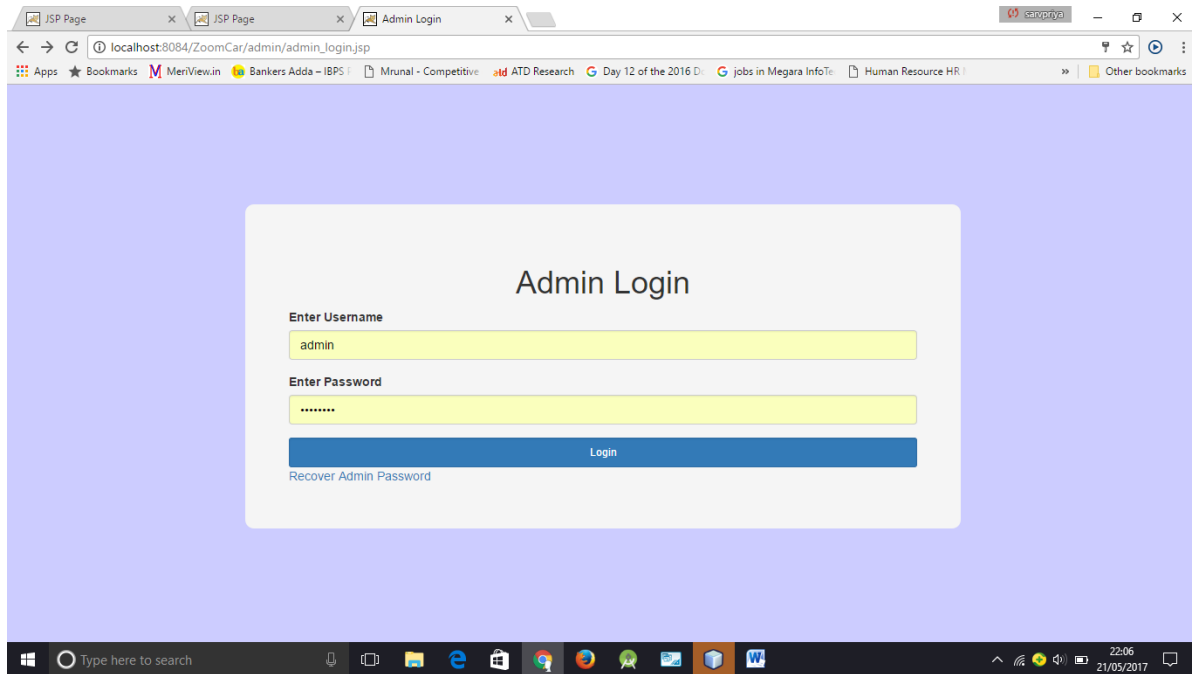
Collation:

Table Default:

Comments:

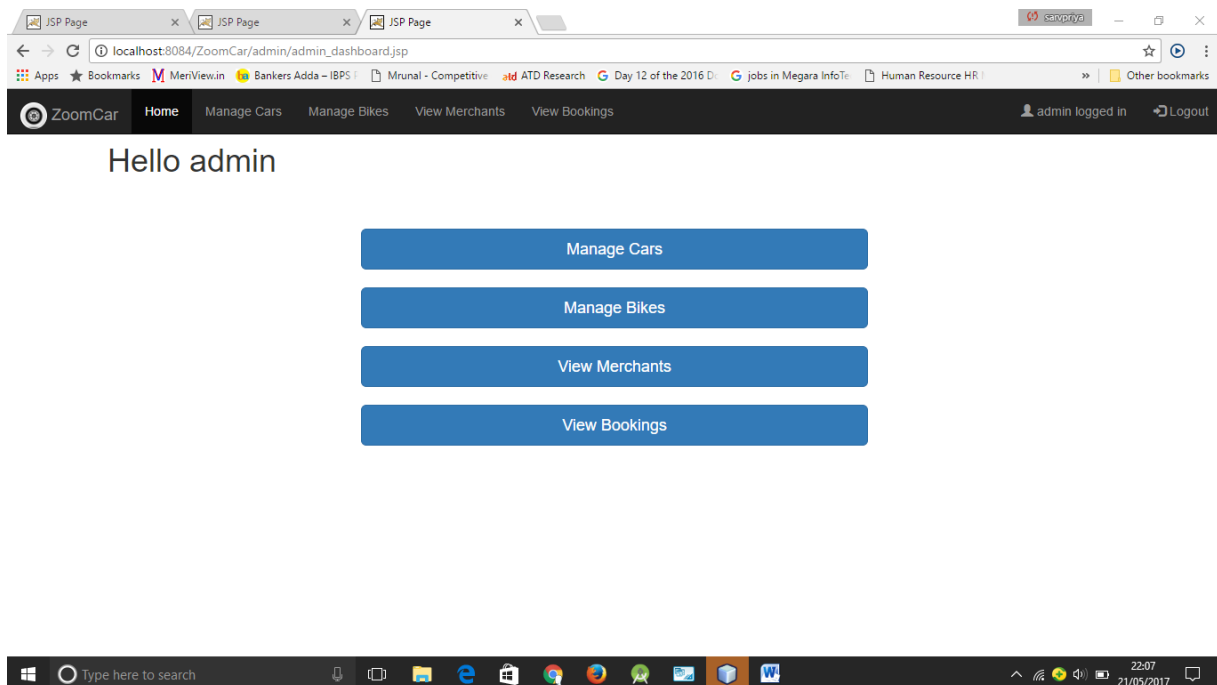
Query Completed

Admin login



The screenshot shows a web browser window with the URL `localhost:8084/ZoomCar/admin/admin_login.jsp`. The page has a light blue background. In the center, there is a white box titled "Admin Login". Inside this box, there are two input fields: "Enter Username" with the text "admin" and "Enter Password" with masked characters "*****". Below these fields is a blue "Login" button. At the bottom of the white box, there is a link that says "Recover Admin Password". The browser's taskbar at the bottom shows various application icons and the system clock indicating 22:06 on 21/05/2017.

Admin dashboard




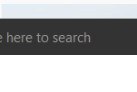


The screenshot shows the "Admin dashboard" page in a web browser. The URL is `localhost:8084/ZoomCar/admin/admin_dashboard.jsp`. The page features a dark navigation bar at the top with the "ZoomCar" logo and several menu items: "Home", "Manage Cars", "Manage Bikes", "View Merchants", and "View Bookings". On the right side of the navigation bar, it indicates "admin logged in" and provides a "Logout" link. Below the navigation bar, the text "Hello admin" is displayed. The main content area contains four blue buttons stacked vertically: "Manage Cars", "Manage Bikes", "View Merchants", and "View Bookings". The browser's taskbar at the bottom shows the same application icons and system clock as the previous screenshot, now at 22:07 on 21/05/2017.

Admin Modules

Manage Cars

Admin can Add, Edit or Delete a vehicle category


Photo	Car Name	Maker	No. of Seats	Car Type	Price	Price per Day	Security	Edit	Delete
	BALENO	MARUTI SUZUKI	8	SUV	900000	4000	1000	Edit	Delete
	BREEZA	MARUTI SUZUKI	8	SUV	1100000	4500	11000	Edit	Delete
	CRETA	HYUNDAI	5	SUV	1000000	3000	10000	Edit	Delete
	HONDA CITY	HONDA	5	SEDAN	900000	3000	1000	Edit	Delete

Add new Car Category

Add New Car Category

Car Name
VERNA

Maker
HYUNDAI

Photo

Choose File | verna.jpg

Number of Seats
5

Car Type
SUV

Car Price
500000






Price per Day
1000

Security
5000

[Submit](#)

Manage Bikes

The screenshot shows a web browser with three tabs: 'JSP Page', 'JSP Page', and 'Admin-ManageBikes'. The address bar shows 'localhost:8084/ZoomCar/admin/manage_bikes.jsp'. The page has a navigation bar with 'ZoomCar', 'Home', 'Manage Cars', 'Manage Bikes', 'View Merchants', and 'View Bookings'. The user is logged in as 'admin'. The main heading is 'Manage Bikes'. Below it is a table with the following data:


Photo	Bike Name	Maker	Bike Type	Price	Price per Day	Security	Edit	Delete
	HONDA ACTIVA 4G	HONDA	SCOOTER	50000	500	500	Edit	Delete
	ROYAL ENFIELD CLASSIC 350	ROYAL ENFIELD	BIKE	106000	1000	1060	Edit	Delete
	ROYAL ENFIELD HIMALYAN	ROYAL ENFIELD	BIKE	150000	1500	1500	Edit	Delete
	ROYAL ENFIELD THUNDERBIRD	ROYAL ENFIELD	BIKE	176000	1700	1760	Edit	Delete
	SUZUKI ACCESS	SUZUKI	SCOOTER	50000	500	500	Edit	Delete

Edit the details of the vehicle category

The screenshot shows a web browser with three tabs: 'JSP Page', 'User Dashboard', and 'Admin-ManageCars'. The address bar shows 'localhost:8084/ZoomCar/admin/edit_car.jsp?name=BREEZA'. The page has a navigation bar with 'ZoomCar', 'Home', 'Manage Cars', 'Manage Bikes', 'View Merchants', and 'View Bookings'. The user is logged in as 'admin'. The main heading is 'Edit Car Category'. Below it is a form with the following fields:

Car Name:

Maker:

Photo:  ☒ Dont Want to change the image ☐ Want to change the image

Number of Seats:

Car Type:

Car Price:

Price per Day:

Security:



[Submit](#)

Manage Merchants- Activate or Deactivate the merchants


ZoomCar Home Manage Cars Manage Bikes View Merchants View Bookings admin logged in Logout

Merchants

Merchants with Status Pending

Photo	Merchant Name	Username	Password	Address	City	State	Mobile number	Email	Status	
	samridhi	sam	123	120, Guru Gobind Singh Nagar	AMRITSAR	PUNJAB	9781232328	samridhi@gmail.com	pending	Activate
	sarv	sarv	123	34, Ranikabhag	LUDHIANA	PUNJAB	313131211	sarv@gmail.com	pending	Activate

Merchants with Status Active

Photo	Merchant Name	Username	Password	Address	City	State	Mobile number	Email	Status	
	Anju	anju	123	12, Queens Road	AMRITSAR	PUNJAB	797597685	anju@gmail.com	active	Deactivate

Admin will Approve the merchant before he can rent out his cars

Admin can deactivate a merchant in case of misbehavior etc

View Bookings of the vehicles

ZoomCar | Home | Manage Cars | Manage Bikes | View Merchants | View Bookings | admin logged in | Logout

View Bookings

Select the month you want to view the bookings of

Cars on Ride

Bill ID	Owner's Name	Car ID	Car Name	Booked By	Date of Booking	Date of Return	Number of days	Price per Day	Total Amt.	Operations
8	anju	7	MARUTI SWIFT DZIRE	sam	2017-05-25	2017-05-30	5	3500	17500	Generate Bill
7	samridhi	11	HONDA ACTIVA 4G	samridhi	2017-05-21	2017-05-24	3	500	1500	Generate Bill

Cars returned and billed

Bill ID	Owner's Name	Car ID	Car Name	Booked By	Date of Booking	Date of Return	Number of days	Price per Day	Total Amt.	Operation
2	samridhi	4	BREEZA	samridhi	2017-05-13	2017-05-18	6	4500	27000	View Bill
6	anju	7	MARUTI SWIFT DZIRE	samridhi	2017-05-23	2017-05-25	2	3500	7000	View Bill

Total Revenue generated of 2017-05 is:7000
Our Commission:700.0

Annotations:


- View all the bookings of vehicles in a month (points to the month selector)
- Admin will generate bill of the vehicle when returned (points to the Generate Bill button)
- Admin can view any bill (points to the View Bill button)
- Admin can see the total revenue generated from bookings (points to the Total Revenue text)
- The commission the company earned (1% of bill) (points to the Our Commission text)

View Bill of A particular bill id

ZoomCar | Home | My Bookings | sam logged in | Logout

Bill of samridhi

Dated: 2017-04-26




Bill Id	2
Car Id	4
Car Name	BREEZA
Booked By	samridhi
Owner	samridhi
Booked From	2017-05-13
Date of car return	2017-05-18
Number of Days	6
Rent per Day	4500
Extra Charges	0
Total Amount	27000
Taxes	300
Coupon code used	
Discount	0
Net Payable Amount	27300

Admin generates Bill for a booking

Browser tabs: JSP Page, User Dashboard, JSP Page. Address bar: localhost:8084/ZoomCar/admin/generate_bill.jsp?bill_id=8. ZoomCar logo. Navigation: Home, Manage Cars, Manage Bikes, View Merchants, View Bookings. User: admin logged in. Logout.

Bill of sam

Dated: Sun May 21 22:57:43 IST 2



Bill Id	8	Date of car return	30/05/2017
Car Id	7	Number of Days	5
Car Name	MARUTI SWIFT DZIRE	Rent per Day	3500
Booked By	sam	Extra Charges	0
Owner	anju	Total Amount	17500
Booked From	2017-05-25	Remarks	
Booked To	2017-05-30	Taxes	0.0
		Enter Coupon code(if any)	
		Discount	0.0
		Net Payable Amount	17500

Admin Generates Bill of a booking

Generate Bill

Merchant Module

Merchant Login


The screenshot shows a web browser window with the address bar displaying `localhost:8084/ZoomCar/merchant/login.jsp`. The browser tabs include 'JSP Page', 'User Dashboard', and 'JSP Page'. The main content area has a light blue background. In the center, there is a white rectangular box containing the 'Merchant Login' form. The form has the title 'Merchant Login' at the top. Below the title, there are two input fields: 'Enter Username' with the value 'sam' and 'Enter Password' with masked characters '***'. A blue 'Login' button is positioned below the password field. At the bottom of the form, there is a link that reads 'Don't Have A Merchant Account. [Click Here To Sign Up.](#)'

Merchant Sign up form

Merchant Signup

Username

Name

Photo

[Choose File](#) IMG_201704...91729.jpg

Address

City

State

Email

Mobile number

Password

Confirm Password

Merchant Dashboard

ZoomCar

Home





My Bookings

samridhi logged in Logout

samridhi Dashboard

Status active

Manage My Cars

S.No	Car Photo	Car Name	No. of Cars	Rent Received per Day	Security deposit	Operations
1		BREEZA	1	4500	11000	Edit Delete
2		HONDA CITY	1	3000	1000	Edit Delete
3		HONDA ACTIVA 4G	1	500	500	Edit Delete
4		YAMAHA YZF R15	1	1000	1110	Edit Delete

Merchant can edit or delete his cars or bikes

Merchant – Add Vehicle to rent

The screenshot shows the 'Add New Cars' form in the ZoomCar merchant dashboard. The form includes the following fields:

- Car Type:** A dropdown menu with 'SUV' selected.
- Select Car:** A dropdown menu with 'MAHINDRA XUV' selected.
- Image:** A placeholder image of a white Mahindra XUV SUV.
- Number of cars you want to rent:** A text input field with '1' entered.
- Maker of the car:** A text input field with 'MAHINDRA' entered.
- Number of seats in the car:** A text input field with '8' entered.
- Market Price of the car:** A text input field with '1500000' entered.
- Rent per day that will be received for this car while on rent:** A text input field with '4500' entered.
- Security deposit made by borrower before renting this car:** A text input field with '15000' entered.

At the bottom of the form is a blue 'Add Car' button.

Merchant- Edit the details of the vehicle

The screenshot shows the 'Edit BREEZA' modal form overlaid on the 'Manage My Cars' table. The modal form includes the following fields:

- Car Type:** A dropdown menu with 'SUV' selected.
- Select Car:** A dropdown menu with 'BREEZA' selected.
- Number of cars you want to rent:** A text input field with '1' entered.

At the bottom of the modal form are two buttons: 'Update Car' and 'Close'.

The background table, 'Manage My Cars', has the following data:

S.No	Car Photo	Car Name	Number of cars you want to rent	Market Price	Rent per day	Security deposit	Operations
1		BREEZA	1	3000	1000	1000	Edit Delete
2		HONDA CITY	1	3000	1000	1000	Edit Delete
3		HONDA ACTIVA 4G	1	500	500	500	Edit Delete
4		YAMAHA YZF R15	1	1000	1110	1110	Edit Delete

User Module

User Login / sign up



User Login

Enter the username

Enter the username

Login

Do Not Have an Account. [Click Here](#)



User Sign Up

Choose a unique username

Enter your email

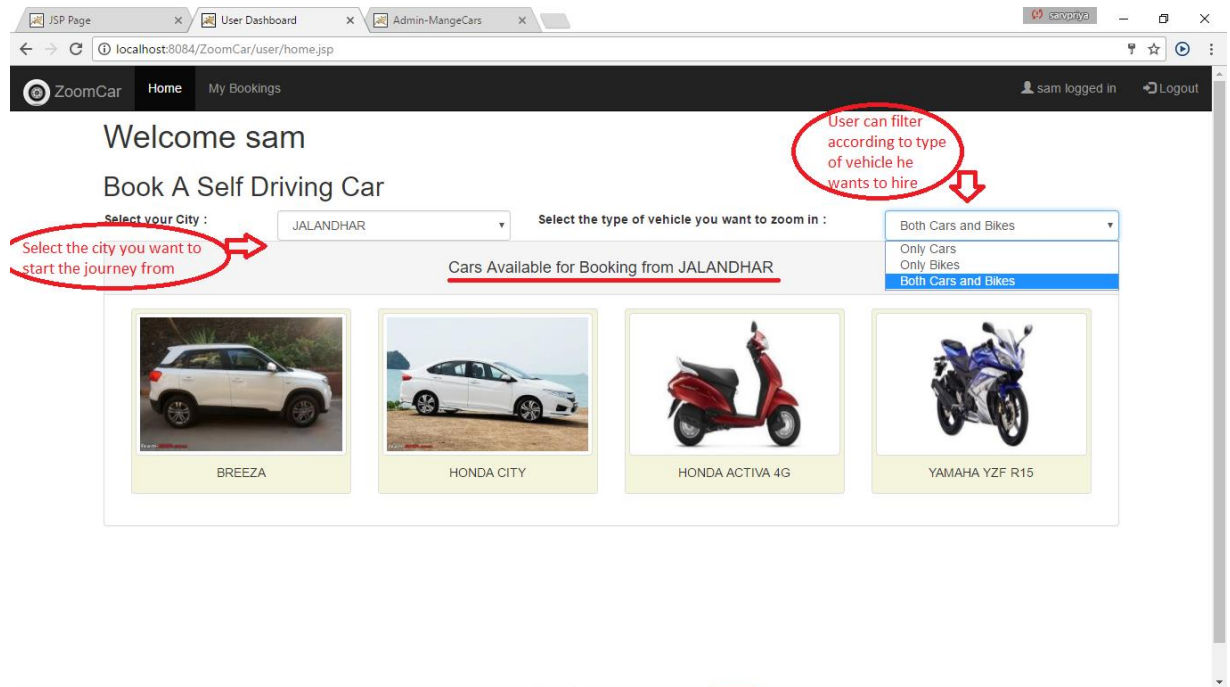
Enter your 10 digit mobile number

Enter Password

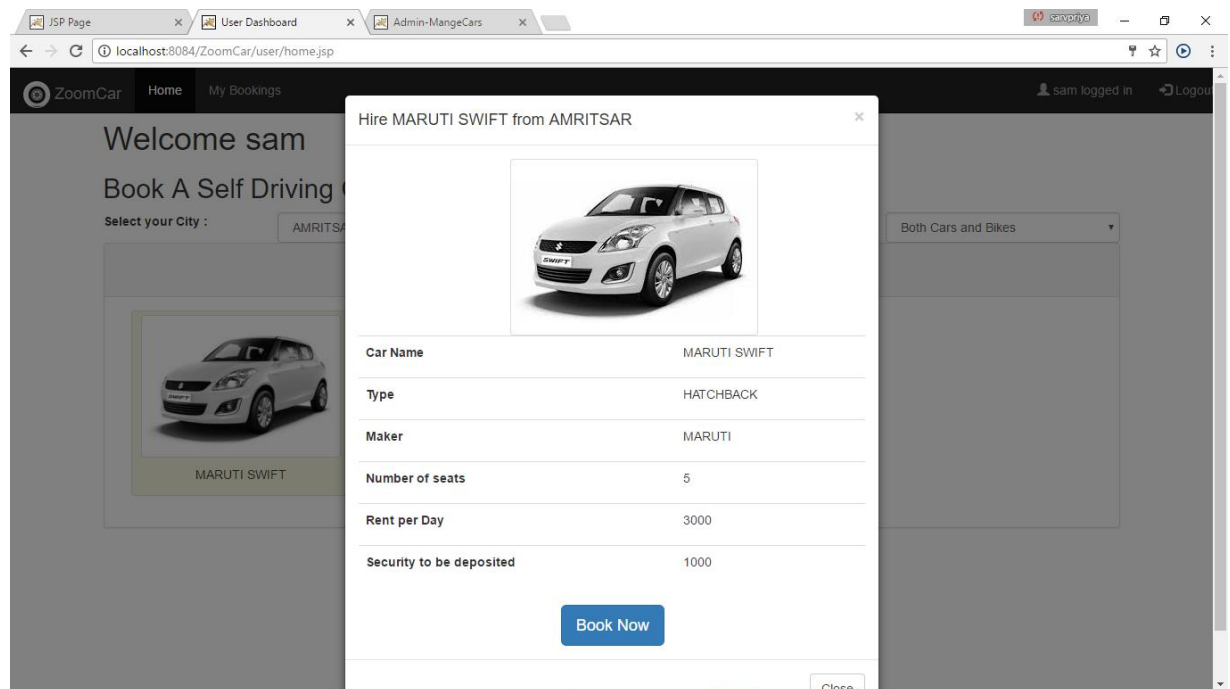
Re-Enter Password

Sign_up

User Home page to book cars/Bikes




Car Booking - select the car



Confirm Booking

ZoomCar Home My Bookings sam logged in Logout

Welcome sam



Car	MARUTI SWIFT
Car Type	HATCHBACK
Maker	MARUTI
Number of seats	5

Logout

From: AMRITSAR

Destination: JALANDHAR

Date of car issue(Car will be issued At 08:00 A.M): 21/05/2017

Date of car return (Car has to be returned By 08:00 A.M on): 23/05/2017

Number of days: 2

Rent per day: 3000

Security to be deposited prior to ride: 1000





Total Amount: 7000

Book

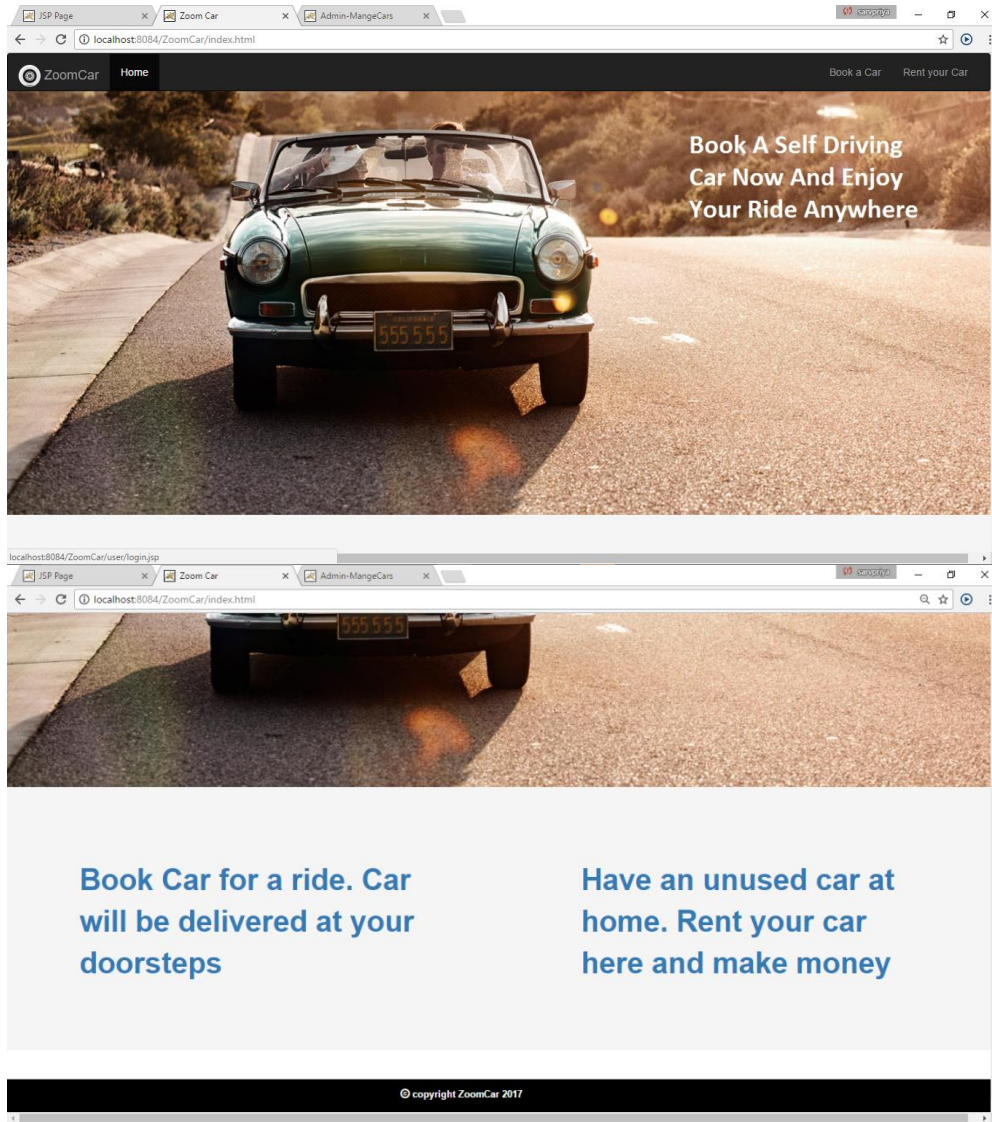
User Can view his Bookings

ZoomCar Home My Bookings samridhi logged in Logout

BOOKING HISTORY

Photo	Car	Booked From	Booked To	Billing Id
	MARUTI SWIFT	2017-04-01	2017-04-05	1
	BREEZA	2017-05-13	2017-05-19	2
	MARUTI SWIFT DZIRE	2017-04-01	2017-04-03	4
	BREEZA	2017-04-26	2017-04-30	5

The Main Page of Website



Suggestions and Recommendations:

BIBLIOGRAPHY

- www.tutorialspoint.com
- www.w3schools.com
- www.stackoverflow.com