

## CSCE -608 Database Systems - Fall 2016

Aditya Emani - 524002563

### Report on Course Project #1 - Carpool

#### **Topic: Design and implementation of a Carpool System between US Cities**

This project (Carpool) is designed to enable users Carpool between major US cities. Any user travelling between two locations can post ride-seeking passengers. There is a feature to cancel the post in case of emergency. Passengers can look for available rides between locations and book. There is a feature for the passenger to cancel booked ride. No payment gateways are integrated in this project.

Front End: HTML/CSS

Server Side: PHP (Hypertext Preprocessor)

Backend: MySQL

Server: <https://www.000webhost.com>

Website: <https://carpool.000webhostapp.com/login.php>

#### **Database Schema: Database is designed with the help of MySQL**

The database for this application is designed using MySQL. The name of the database is: **id18655\_prasaadem**. It consists of 4 tables namely users, Ride\_Meter, us\_cities, booking.

The tuples for table us\_cities is collected from the data set obtained from this website <https://www.uscitieslist.org>. All other information in the rest of tables is not real.

#### **Table Structure:**

Users: The users table is used to store key user information such as the user name, email address, user Id and password. The structure of users table is as follows:

Server: localhost:3306 - Database: id18655\_register - Table: users

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary Unique More
2	<b>username</b>	varchar(50)	utf8_unicode_ci		No	None			Change Drop Primary Unique More
3	<b>email</b>	varchar(50)	utf8_unicode_ci		No	None			Change Drop Primary Unique More
4	<b>password</b>	varchar(50)	utf8_unicode_ci		No	None			Change Drop Primary Unique More
5	<b>trn_date</b>	datetime			No	None			Change Drop Primary Unique More

us\_cities: This table has all the information about the popular cities in the United States. The structure of this table is as follows:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id</b>	int(11)			No	None			Change Drop Primary Unique More
2	<b>name</b>	varchar(35)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
3	<b>county</b>	varchar(72)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
4	<b>state_code</b>	char(2)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
5	<b>state</b>	varchar(20)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
6	<b>zip_codes</b>	varchar(1685)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
7	<b>type</b>	varchar(23)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
8	<b>latitude</b>	decimal(8,5)			Yes	NULL			Change Drop Primary Unique More
9	<b>longitude</b>	decimal(8,5)			Yes	NULL			Change Drop Primary Unique More
10	<b>area_code</b>	char(3)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More
11	<b>population</b>	int(11)			Yes	NULL			Change Drop Primary Unique More
12	<b>households</b>	int(11)			Yes	NULL			Change Drop Primary Unique More
13	<b>median_income</b>	int(11)			Yes	NULL			Change Drop Primary Unique More
14	<b>land_area</b>	bigint(20)			Yes	NULL			Change Drop Primary Unique More
15	<b>water_area</b>	bigint(20)			Yes	NULL			Change Drop Primary Unique More
16	<b>time_zone</b>	varchar(30)	utf8_unicode_ci		Yes	NULL			Change Drop Primary Unique More

Ride\_Meter: This table is used to store the information of all the available rides. Any ride posted by user is saved into this table. The structure of this table is as follows.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>Ride_id</b>	int(6)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop Primary More
2	<b>name_user</b>	varchar(30)	utf8_unicode_ci		No	None			Change Drop Primary More
3	<b>source_loc</b>	varchar(30)	utf8_unicode_ci		No	None			Change Drop Primary More
4	<b>Destination</b>	varchar(30)	utf8_unicode_ci		No	None			Change Drop Primary More
5	<b>Start_date</b>	datetime			Yes	NULL			Change Drop Primary More
6	<b>cost</b>	int(6)			No	None			Change Drop Primary More

bookings: This table has just two columns in it. It stores the Ride\_id from the previous table and the name of the user. When a passenger books a ride, all the information about his booking is stored in this table. The structure of this table is as below:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	book_id	int(6)		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
<input type="checkbox"/>	2	user_name	varchar(32) utf8_unicode_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>

### Components: Registration, Login, Session, Post/Cancel Ride – Book/Cancel Ride

There are 5 major components in this project.

Registration: A new user should register him to log in to the application. The registration screen helps user with the registration process. Any user can register with a user name, password and email address. The user password is saved into the database after encryption. The registration screen is as below:

Registration

Username

Password

Email

REGISTER

Login: After the registration, the user can login to the application through the login screen with his username and password. The login screen is as below:

The image shows a login form with a light gray background. At the top, there is a header section with the text "Log In". Below this, there are two input fields. The first input field contains the text "user1". The second input field contains five dots, representing a password. Below the input fields, there is a large teal button with the text "LOGIN" in white. At the bottom of the form, there is a link that says "Not registered yet? [Register Here](#)".

Session: The session of the user is maintained until he logs out of the application. This way the user doesn't have to log in every time he loads a page.

Once the user is logged in, he will be redirected to the home page. User can navigate from home page to his dashboard. The dashboard has all the instructions on how to access every feature of the application. A dashboard in this application looks like below:

Home	Post a Ride	Cancel Posted Ride	View Available Rides	Cancel Booked Ride	Logout
------	-------------	--------------------	----------------------	--------------------	--------

## Welcome to Carpool Dashboard.

Home takes you to dashboard

If you are a driver, to post a ride, navigate to "Post a Ride"

If you are a driver, to cancel a posted ride, navigate to "Cancel posted Ride"

If you are a passenger, to book a ride, navigate to "View Available Rides"

If you are a passenger, to cancel a booked ride, navigate to "Cancel Booked Ride"

Remember, your session is stored. To logout of your account, navigate to Logout

This website is created by **Aditya Emani** as a part of coursework for TAMU CSCE 608, Database Systems Course **not for commercial purpose..**

Post/Cancel Ride: A user can post a ride through the below page.

Home	Post a Ride	Cancel Posted Ride	View Available Rides	Cancel Booked Ride	Logout
------	-------------	--------------------	----------------------	--------------------	--------

## Post New Ride

Source City  
Amana

Destination City  
Alleman

Cost  
80

Date of Travel  
2016-10-11 23:45:33

SUBMIT

This website is created by **Aditya Emani** as a part of coursework for TAMU CSCE 608, Database Systems Course **not for commercial purpose..**

A user can cancel a posted ride through the below page. He can select the radio button corresponding to the ride he wants to cancel from the right and at the bottom of the page, click the button cancel. **If a user cancels a posted ride, all the bookings of passengers who booked his ride get cancelled.**

Home	Post a Ride	Cancel Posted Ride	View Available Rides	Cancel Booked Ride	Logout
------	-------------	--------------------	----------------------	--------------------	--------

## Cancel Posted Rides

Ride Id	Posted By	Source	Destination	Cost	Start Time	Cancel
1034	user1	Alburnett	Albion	40	2016-10-11 23:48:08	<input checked="" type="radio"/> 1034

**Book Ride:** A user can book a ride through the below page. All the rides posted by users are displayed in this page. User can select the radio button corresponding to the ride he wants to book from the right and at the bottom of the page, click the button Book.

Home	Post a Ride	Cancel Posted Ride	View Available Rides	Cancel Booked Ride	Logout
------	-------------	--------------------	----------------------	--------------------	--------

## Book a Ride

Ride Id	Posted By	Source	Destination	Cost	Start Time	Book
1	Satya	Ackley	Ackley	15	0000-00-00 00:00:00	<input checked="" type="radio"/> 1
2	Satya	Ackley	Ackworth	20	0000-00-00 00:00:00	<input type="radio"/> 2
3	Sardar	Ackley	Adair	25	0000-00-00 00:00:00	<input type="radio"/> 3
4	Sardar	Ackley	Adel	30	0000-00-00 00:00:00	<input type="radio"/> 4
5	Sardar	Ackley	Afton	35	0000-00-00 00:00:00	<input type="radio"/> 5
6	Sardar	Ackley	Agency	40	0000-00-00 00:00:00	<input type="radio"/> 6
7	Sardar	Adel	Airneworth	45	0000-00-00 00:00:00	<input type="radio"/> 7

**Cancel Ride:** A user can cancel a ride through the below page. He can select the radio button corresponding to the ride he wants to cancel from the right and at the bottom of the page, click the button cancel.

## Challenges:

1. One of the major challenges I faced in creating this project is in writing the SQL queries along with PHP to obtain information from database. Stackoverflow answered most of the questions.
2. Loading data obtained from PHP and SQL into a html table. The table has to be dynamic and php script solved the problem.
3. Previously, I assigned, for the booking or cancel a ride, every row of table a button. It then got difficult to identify which button was pressed as PHP references a button call by its name and the table is dynamic in size. So every button must have a separate function. So I replaced the buttons from each row with a radio button. As we know that all the radio buttons in a form have the same name and different values, it fixed the problem.

## References:

1. The url for the web project is:  
<https://carpool.000webhostapp.com/login.php>
2. Server is hosted in: <https://www.000webhost.com>
3. Database and server code is hosted in :  
<https://databases-auth.000webhost.com/index.php>
4. <http://stackoverflow.com> for the dynamic height of HTML Table
5. us\_cities dataset from <https://www.uscitieslist.org>
6. HTML & CSS queries from <http://www.w3schools.com>