**GRADUATE PROJECT**

**“Metro Ticket Booking”**

**Submitted by-**

**Dipika Mahashabde (CWID:803004274)**



**CPSC 473 - Web Programming and Data Management**

**Professor: Prof. Kenytt Avery**

**Department of Computer Science**

**California State University, Fullerton**

**Spring 2016**

**1. Introduction**

**“Metro ticket booking”** the basic aim of the web-application is to reduce the time people take to stand in the line and buy tickets for the metro. Our application will help them book the ticket on-line prior to going to the station for the journey. Our customers will be able to see the train schedules prior to book their ticket and also see the time slot allotted to the train to travel from different locations. They will also be able to check the fare of their travel. Also, they can book tickets for max 5 people (ppl). Thereafter the customer may proceed to save their transaction and checkout to payment. The customer may also login at a later time and check his saved “history”.

Technologies used:

* AngularJS
* Express.js
* Node.js
* CouchDB
* Nano

The URL for github is given as follows:

**Steps for installing and running the application:**

* Install CouchDB on the terminal :-

sudo su -c “apt-get install couchdb”

* After the installation start the service :-

sudo service couchdb start

you can also see on this link :-

http://localhost:5984/\_utils

* Download and install node.js from https://nodejs.org/
* Install express server by typing the following command on the terminal in your folder of project:-

“npm install express”.

* Install body-parser by typing the following command on the terminal in your folder of project:-

“npm install body-parser”.

* Install nano server by typing the following command on the terminal in your folder of project:-

“npm install nano”.

* In Couchdb we need to create our database for information of trains therefore run a JavaScript file named createdb.js using the following command on the terminal in your folder of project:-

node createdb.js

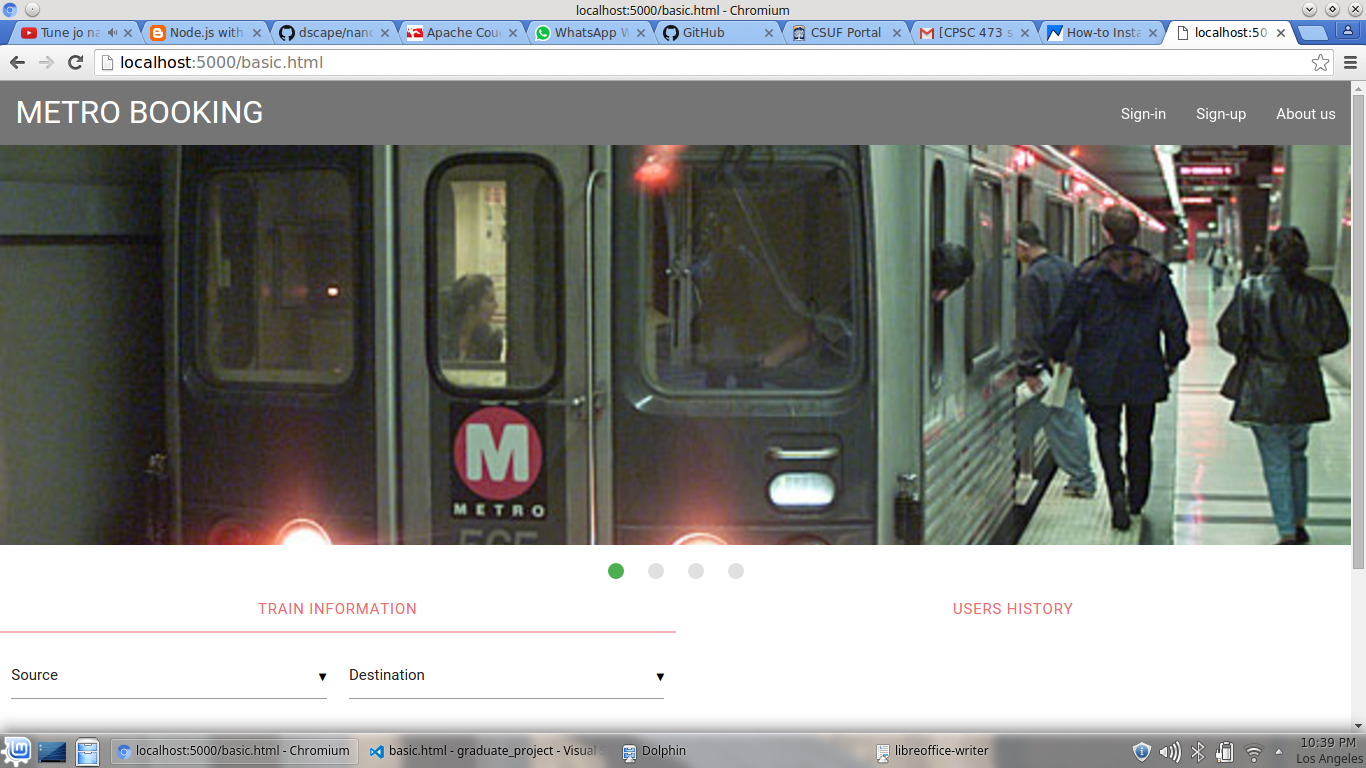
* Run node server using the following commandon the terminal in your folder of project (for running server side javascript file):

“node app.js”.

* Run the application using the following url on the browser (for running basic.html):

http://localhost:5000/basic.html

The page will open up something like this :-



The user-interface is done using Materializecss (<http://materializecss.com/>)

the components used are:-

1. Slider :- it continuously slide images saved on the site.

2. NavBar :- it contains A logo or brand link, and the navigations links.

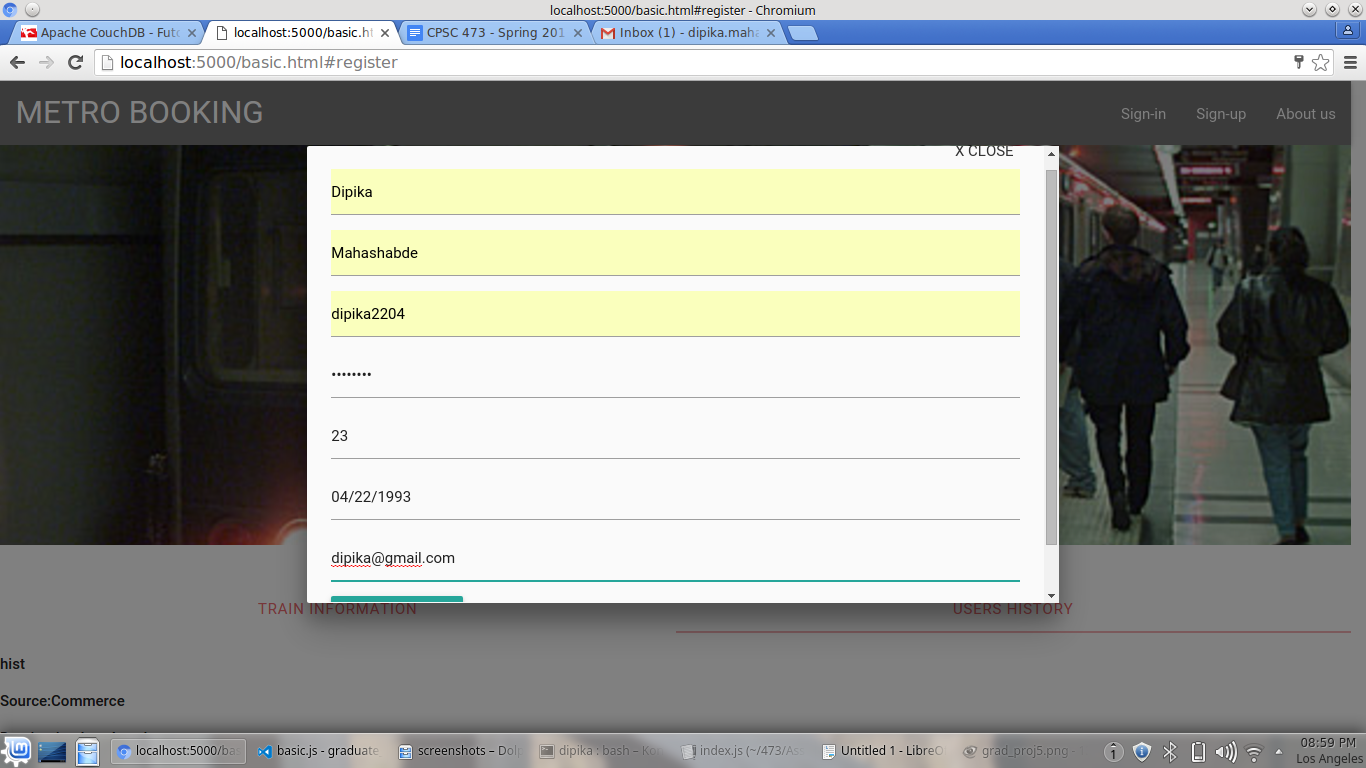
3.Modal :- Use a modal for dialog boxes, confirmation messages, or other content that can be called up.

4. tabs :- The tabs structure consists of an unordered list of tabs that have hashes corresponding to tab ids. Then when you click on each tab, only the container with the corresponding tab id will become visible. You can add the class .disabled to make a tab inaccessible.

Working of the application:-

step 1:-

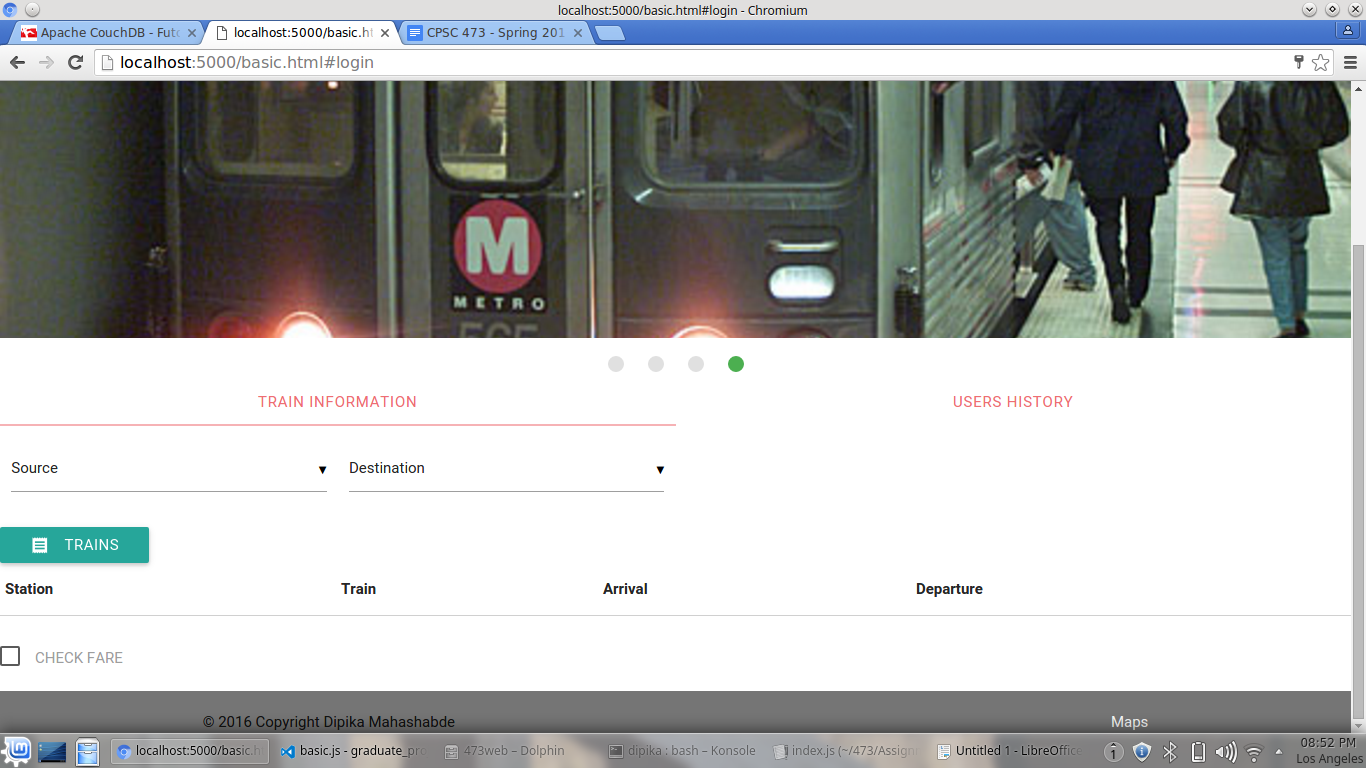
Sign-up (registeration modal)



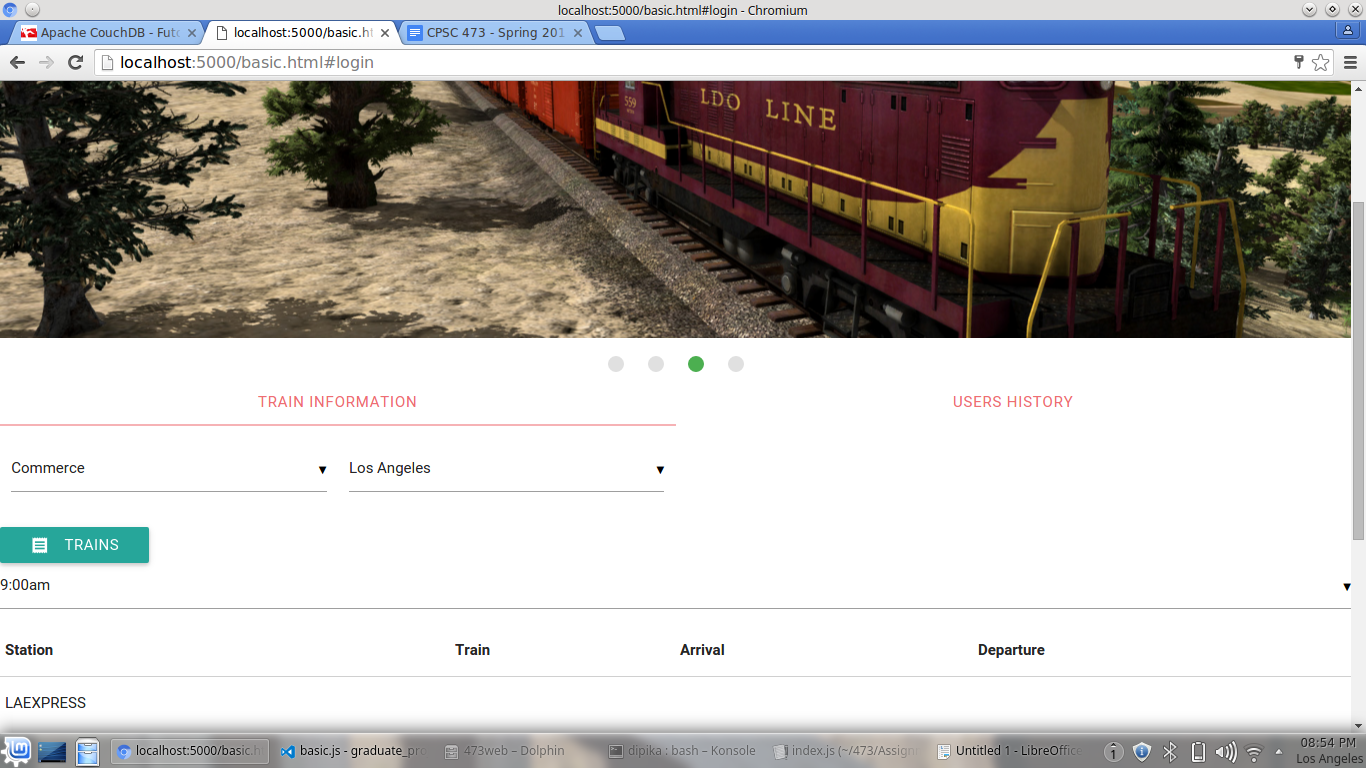
The customer has to fill in their information to be stored in the database, and also be able to login later.

The function adding\_to\_json() is the function in basic.js called when the customer submits his info. The data is send to the server in the structure of json. This function calls the sever function user\_info() to connect to the database.

After you register this page will appear :-

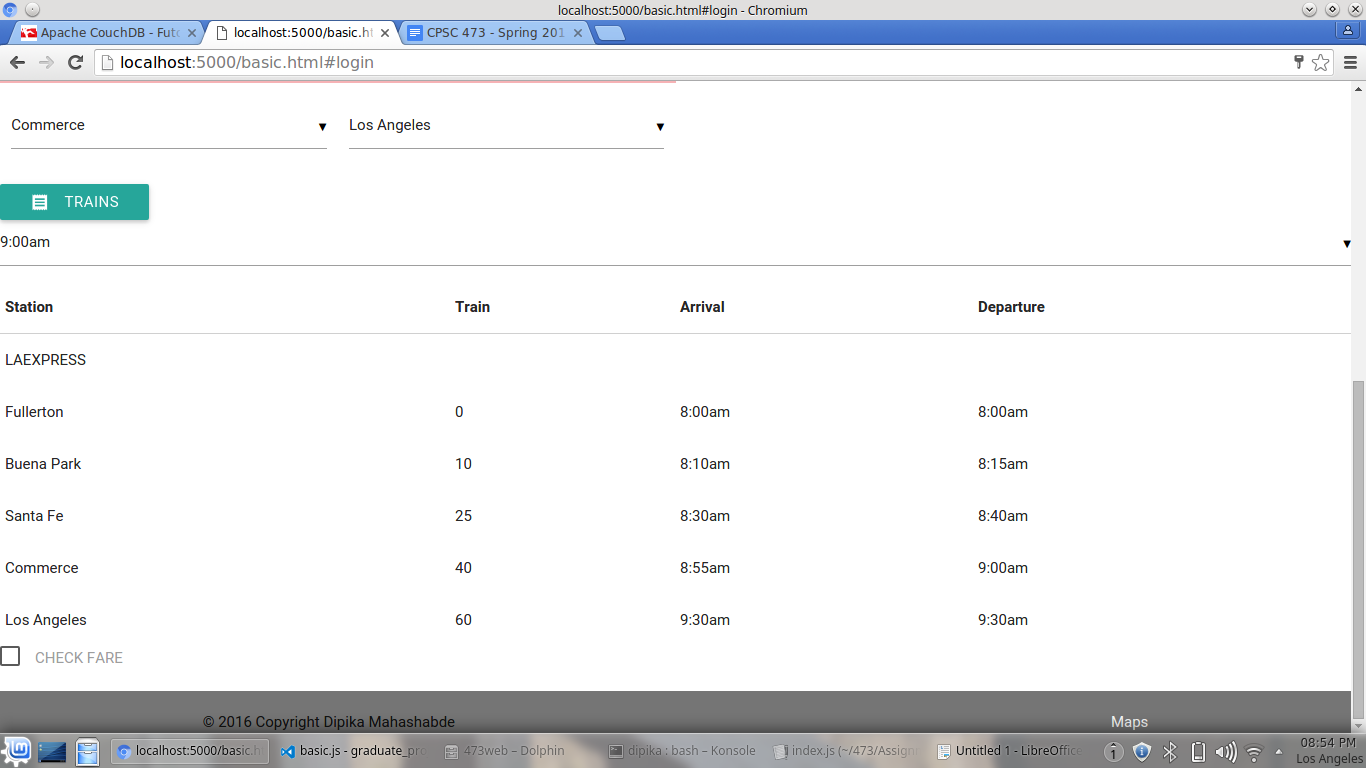


Step 2 :- check for train

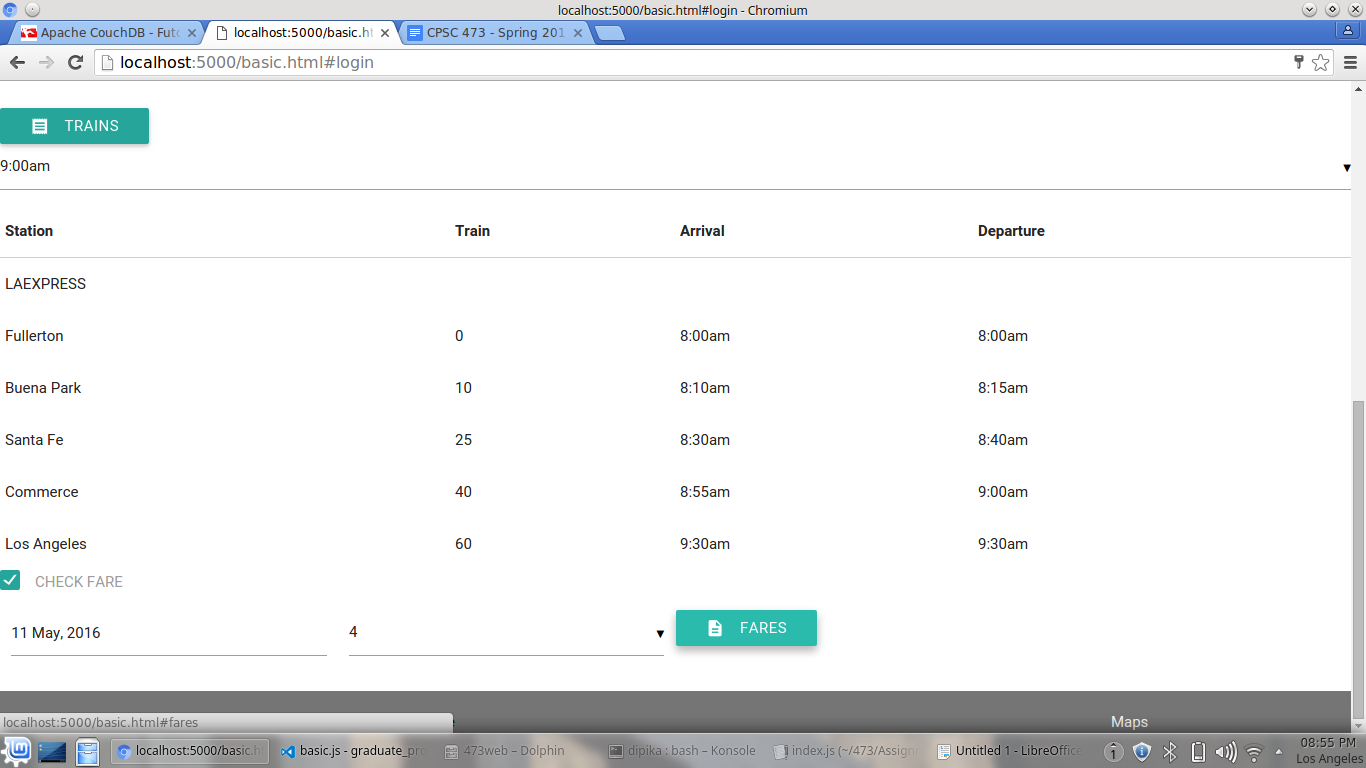


from the select button select the source and destination location where you want to go from a location to search if a train exist for this route.

If there is a train you will get the train schedules and time slots it travels from the database.

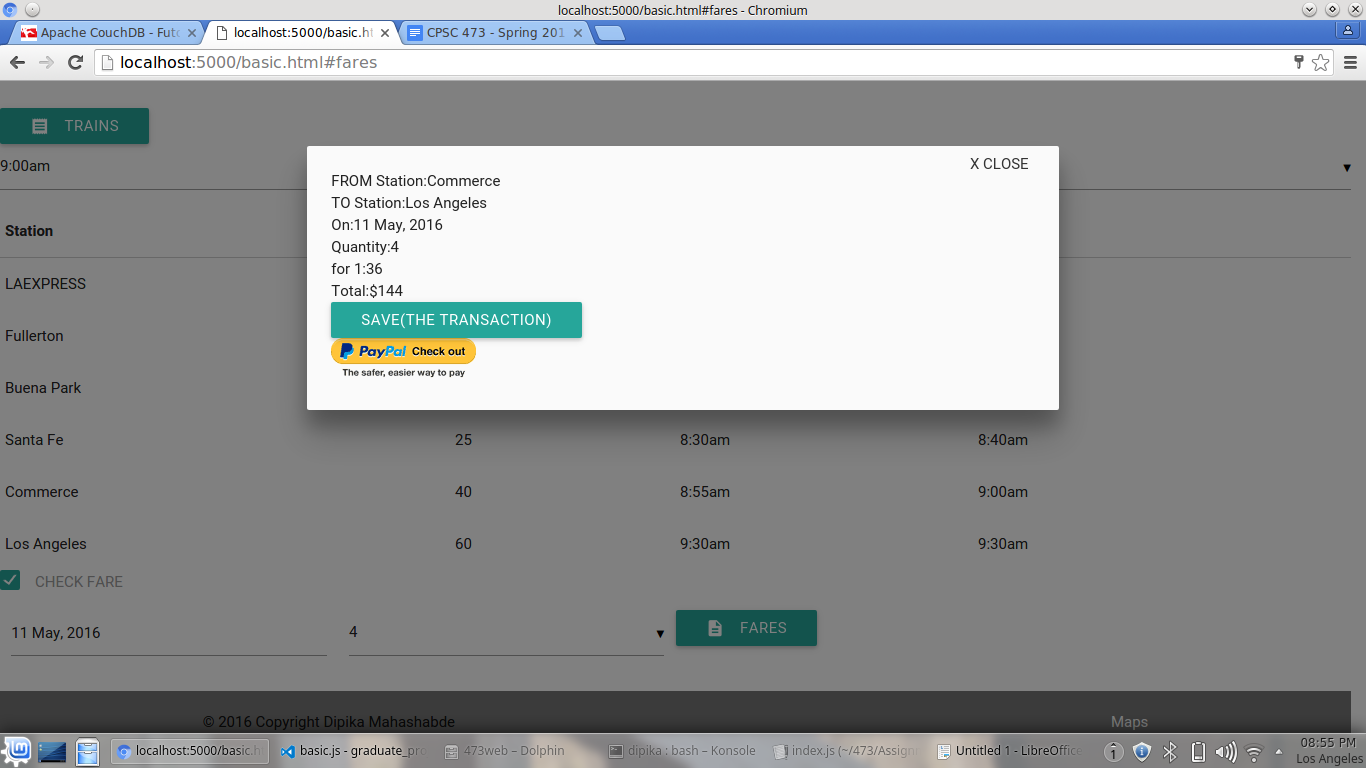


Step 3:- check the fare



the checkbox at the bottom help you to check the fare of the train. Before you do u need to insert the date of journey and how many people. And then check the fare.

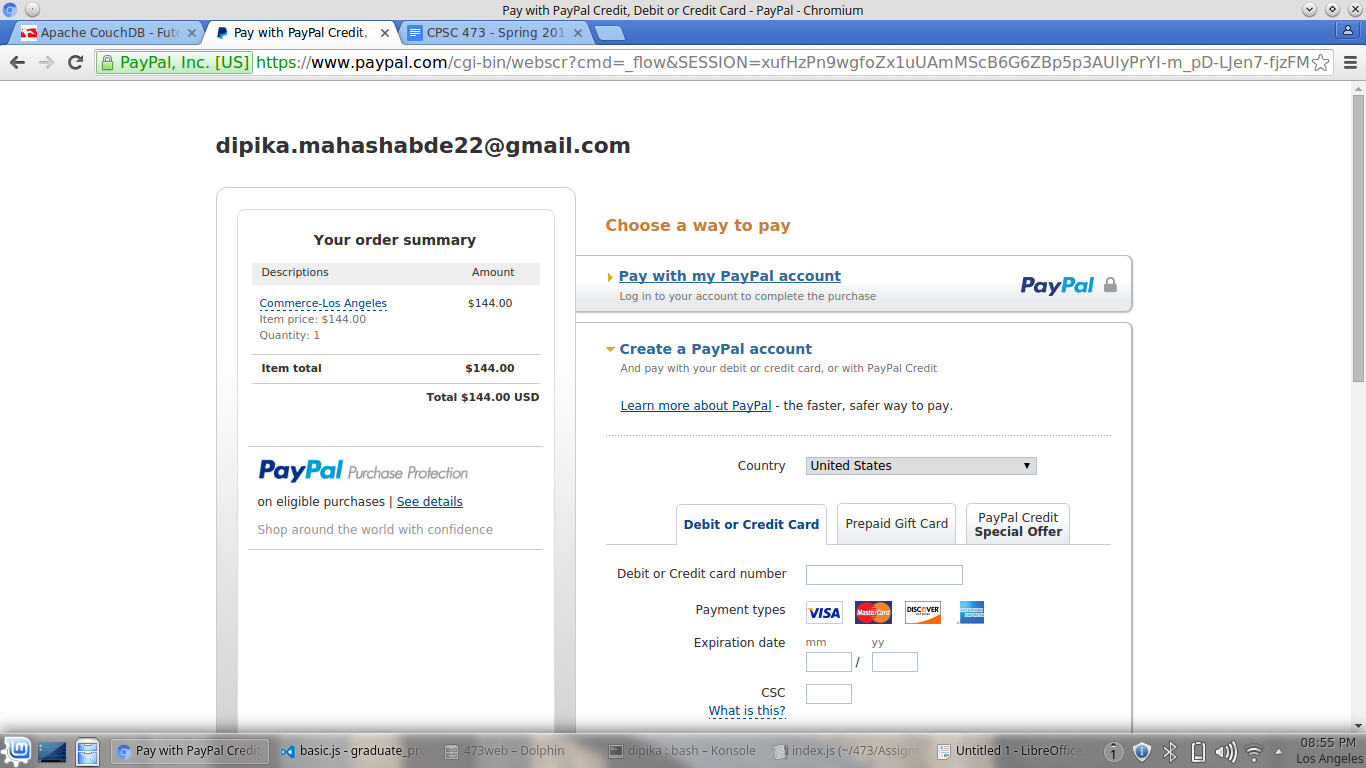
A new modal will be open to show the fare of one person and how many you want

****

step 4:- save your transaction

press the button for saving your transaction in the history.

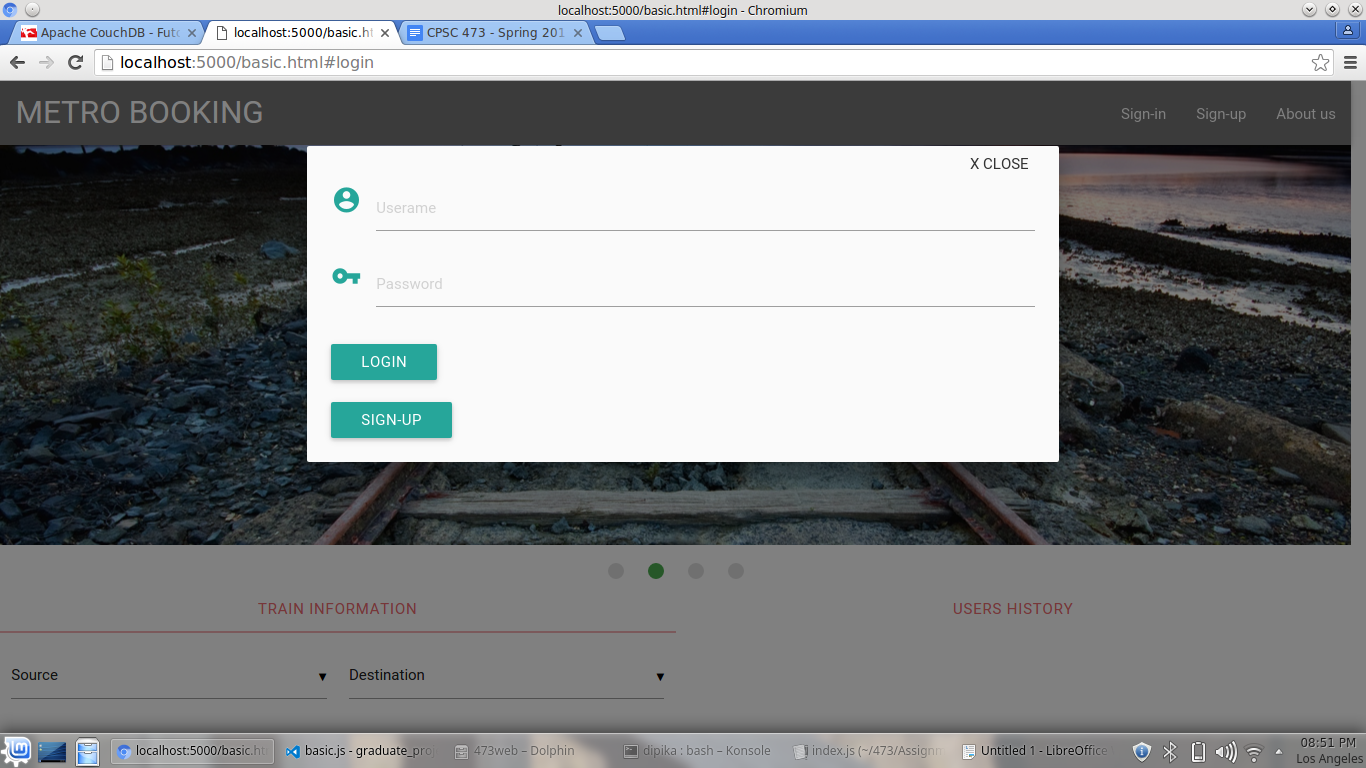
It will be reflected to the user's history.

Step 5 :- payment

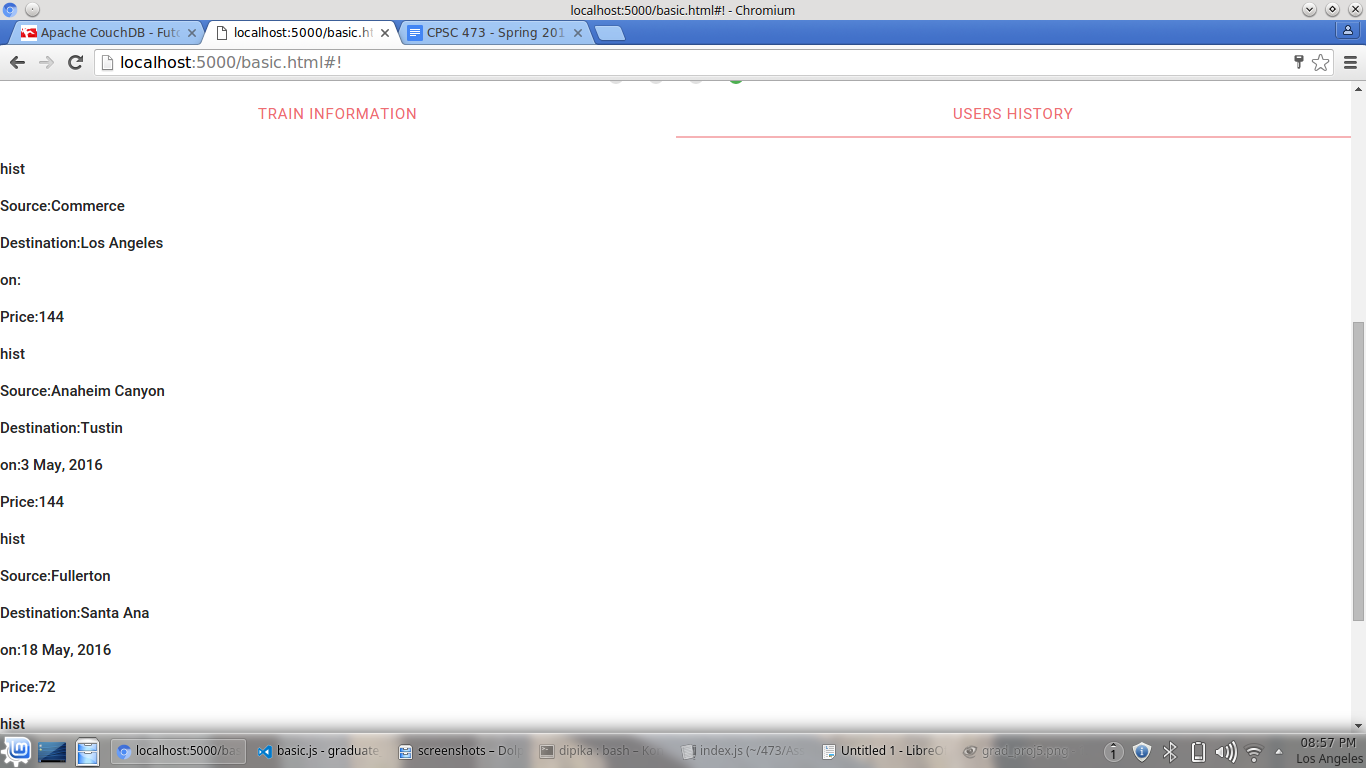
after pressing the button of paypal you will be redirected to the paypal dummy site.

And Yippie!! ticket is booked.

Step 6 :- login to check the history

****

after you login click on the user's history ti see your history.

****