Final Report

Members: Giordanno Castro & Warren Scipio
Proposal Idea: Modified Taxi Service Webpage
Date: 12-10-15

Our group project will work on the proposal of the taxi service webpage. This proposal defines a webpage that allows users to request taxi cabs, similar to Uber. However, the website allows customers to input options such as the desired driver or type of vehicle. Customers will also have the option to store favorite destinations in order to make it faster to request a taxi. The website will also allow the registration of new drivers. Drivers will have the ability to toggle their availability as well as to be able to accept or deny requests from customers around them.

Warren's proposal allowed a user to search for multiple locations. Based on the proximity to these locations and the preferences of the user's friends, the service suggests a list of places to go out. However, getting the information of these locations seemed too complicated. We would have needed to apply to certain private API's and due to time constraints we cannot afford to do that. Therefore we plan to only allow the users to store their favorite restaurants and cinemas, then they can select these places from their favorite locations list. The taxi driver will already know the destination location without the user having to explicitly state it. These feature combined with the ability to store favorite locations such as the user's work location will make our service really useful. Just the ability to store places gives our service and advantage over Uber which does not allow to do so. However, our service will also allow to pick a preferred driver that the customer trusts.

The work was divided as such:

- Warren worked on the customer side of the project. Therefore, he will make sure that the database was populated with customer data such as the username, passwords, first and last name. He also made sure that the data of the customer is saved to the webpage and retrieved from it as well. Items such as the list of preferred drivers and favorite destinations. Users are able to privately request a pickup from one of their preferred drivers.
- Giordanno was responsible for the driver side of the webpage. Therefore he had to make sure that the drivers could sign up with their personal data. However, for each driver there is an associated vehicle that needs to be appropriately stored with its respective data. For a registered driver, the database needed to be able to distinguish whether the driver is available or not. The driver is also able to see any new services that he can accept based on the type of vehicle that he/she owns. Drivers are also able to accept any private requests from users that have them stored as their favorite drivers.

Major Problems encountered

The biggest problem encountered seemed to be using google maps, the team overestimated how much control we would have over the map and how much information we would get from it. That forced us to use Lat/Long data when it came to locations on the map. Another challenge was learning php and getting a single page to read and write to the database and update itself. This proved to be quite a complicated task due to the complexity of the database's design. However, the database schema required on the project proposal was a great help.

Explanations regarding code and implementation

Our website connects two users, a ‘driver’ and a ‘client’. Clients can request a driver to pick them up as well as select different options such as the desired type of vehicle and the pickup location and destination. On the driver side, the webpage collects all the personal data and appropriately stores it. The website also displays any client that is requesting a ride and that the driver can pick up.

Our web layout uses two main pages one for client and one for drivers. Both pages contain an iframe within them that updates and changes based on the clicks in the page. The pages within the iframe do the majority of the querying. A session is passed among all of the iframe files associated with the main page active, to pass the username variable of the user currently logged in.

The driver side of the webpage uses a two php files per site. One php file is mostly focused on the html side of the code, displaying the basic structure of the website. Meanwhile the second php file is focused on accessing the database elements and either retrieving or modifying them.

On the other hand, the client side of the webpage uses a single php file. This file contains the html structure of the website. However, when it is necessary to access the database, the file recurses on itself and access pure PHP code.