CS353
Project Final Report

Car Sharing System: ***Passenger's Guide to the Galaxy***

Group 23

Ismail Kerimov 21300355
Selim Mıdıkoğlu 21200394
Suat Enes Koç 21301776
Mustafa Yıldız 21001467

http://passengersguidetothegalaxy.weebly.com

May 12, 2017

Table Of Contents

1. Description of the System

This system is a web application that uses database systems heavily. This application is aiming to help its users to share their rides or to participate the rides of other drivers. We call our application as passenger's Guide to the Galaxy (PGG). PGG joins the drivers who has empty seats in their cars and the passengers who wants to take exact same route with the drivers. Drivers are able to collect money when they accept the reservations of passengers.

The users of passenger's Guide to the Galaxy (PGG) can be participated in a trip either as a passenger or as a driver. Drivers will publish their own travel plans, give number of passengers they will accept and ticket money for each. Passengers will look at the publicities and choose trips that would be suitable for them.

Inter-city Travel Drivers will give paths that they go every weekday. So if a passenger relates to that path, he/she can share a car for one or more days. Inter-urban Travel Driver would give an announcement specified date, time for leaving, departure city, target city and intermediate stops. Passenger will choose a publication which fits him/her the most.

2. Final E/R Diagram

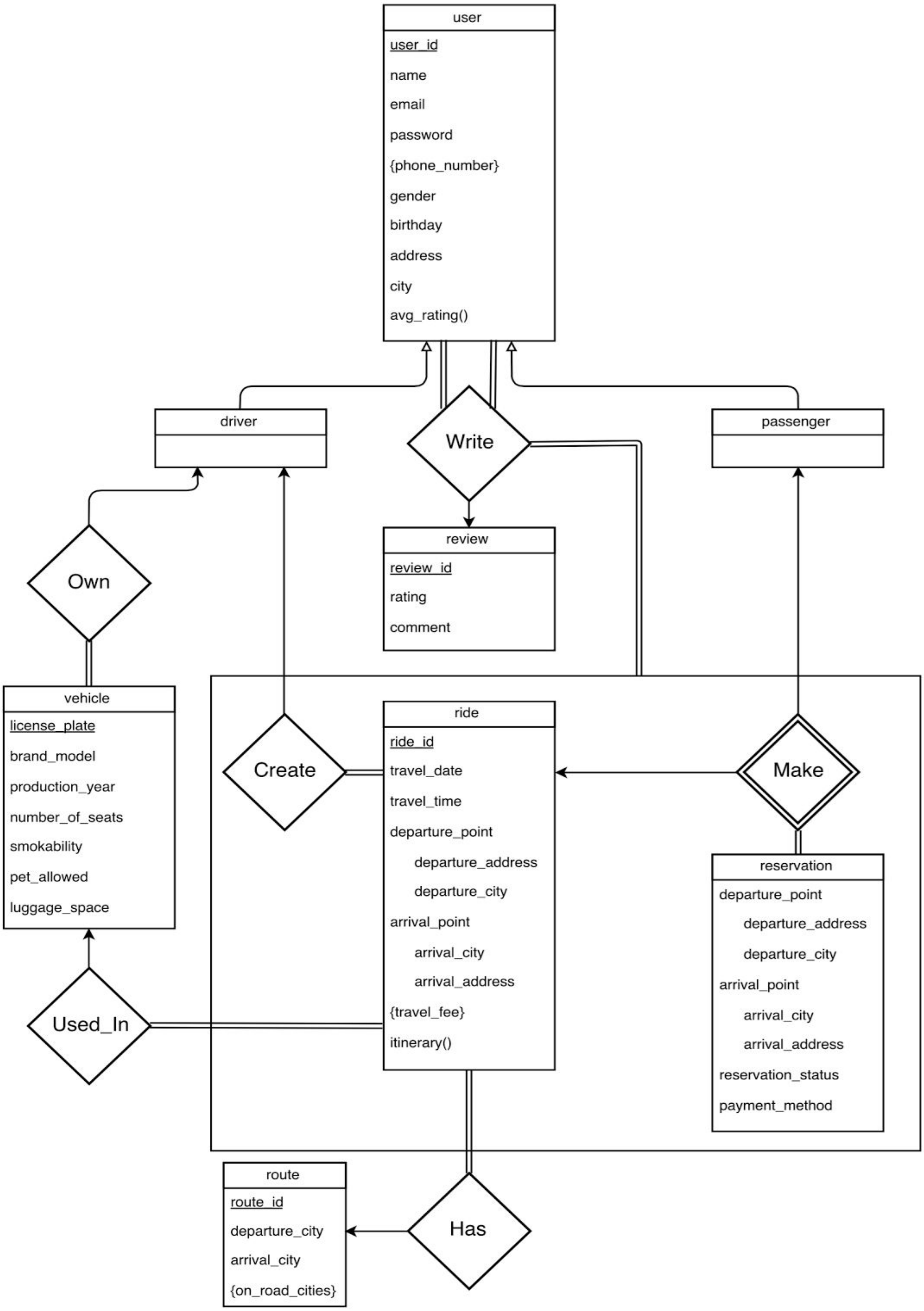


Figure 1. Final E/R Diagram

3. Final List of Tables

3.1. User

user(user_id, email, password, name, gender, birthday, address, city, avg_rating)

3.2. User’s Phone Numbers

user_phone(user_id, phone_number)

FK: user_id references user

3.3. Driver

driver(driver_id)

FK: driver_id references user(user_id)

3.4. Passenger

passenger(passenger_id)

FK: passenger_id references user(user_id)

3.5. Ride

ride(ride_id, license_plate, driver_id, route_id, travel_date, travel_time,
departure_address, departure_city, arrival_city, arrival_address)

FK: license_plate references vehicle

FK: driver_id references driver

FK: route_id references route

3.6. Ride’s Fee

ride_fee (ride_id, travel_fee)

FK:ride_id references ride

3.7. Review

review(review_id, reviewer_id, reviewee_id, ride_id, rating, comment)

FK: reviewee_id references user(user_id)

FK:reviewer_id references user(user_id)

FK: ride_id references ride

3.8. Making Reservation

make_reservation(ride_id, passenger_id, departure_address, departure_city,
arrival_city, arrival_address, reservation_status, payment_method)

FK: ride_id references ride

FK:passenger_id references passenger

3.9. Vehicle

vehicle(license_plate, driver_id, brand_model, production_year, number_of_seats,
smokability, pet_allowed, luggage_space)

FK: driver_id references driver

3.10. Route

route(route_id, departure_city, arrival_city)

3.11. Route's On Road Cities

route_on_road_cities(route_id, departure_city, arrival_city)

FK: route_id references route

4. Implementation Details

This system is a CRUD (abbreviation for create, read, update, delete) web application that have front-end and back-end components. For the front-end GUI design, HTML, CSS and JavaScript is used. PHP establishes the connection of website and the database in the backend webserver. Our web server is Apache. So, the system runs in LAMP stack. To create and manage the database, phpMyAdmin panel is used. The database system used is MySQL.

We used a php file for the functions of database connection, fetching, inserting, deleting and updating. To solve the problem for passing values across the pages, the system is using session structure of PHP

5. Advanced DB Features

5.1. Reports

- Obtain a list in alphabetical order of users who have participated at least 30 rides:

```
SELECT user_id
FROM user u, ride r, make_reservation m
WHERE u.user_id = r.driver_id
        OR (u.user_id = m.passenger_id
        AND m.reservation_status = ‘accepted’)
GROUP BY user_id
HAVING COUNT(*) >= 30
```

- List cities and the money generated for each destination city in descending order:

```
SELECT departure_city, arrival_city, SUM(travel_fee)
FROM route JOIN ride USING (route_id) R1, ride_fee R2
WHERE R1.ride_id = R2.ride_id
GROUP BY departure_city
ORDER BY 3 DESC
```

5.2. Views

- List passenger’s accepted reservations:

```
CREATE VIEW passengers_accepted_reservations AS
        SELECT ride_id, departure_city, arrival_city
        FROM passenger p, ride r, make_reservation m
        WHERE p.passenger_id = m.passenger_id
                AND m.ride_id = r.ride_id
                AND reservation_status = ‘accepted’
```

- List all the drivers who has rating more than 4.5 and created more than 20 ride:

```
CREATE VIEW best_driver AS
        SELECT driver_id, avg_rating
        FROM driver NATURAL JOIN ride T1, user T2
        WHERE T1.driver_id = T2.user_id
                AND avg_rating > 4.5
                AND 20 < (SELECT COUNT(*)
                           FROM T1)
```

5.3. Triggers

- When an user gets a review, update the average rating of the user:

```
CREATE TRIGGER update_rating AFTER INSERT ON review
```

```
REFERENCING NEW ROW AS nrow
FOR EACH ROW
WHEN EXIST (SELECT user_id
            FROM user
            WHERE user_id = nrow.reviewee_id)
BEGIN ATOMIC
      UPDATE user
      SET avg_rating = (nrow.rating + avg_rating
            * ( SELECT COUNT(*)
                  FROM review
                  WHERE reviewee.id = nrow.reviewee_id))
            / (   SELECT COUNT(*)
                  FROM review
                  WHERE reviewee.id = nrow.reviewee_id))
      WHERE user_id = nrow.reviewee_id
```

- If passenger changes the payment method of reservation, status becomes ‘waiting’:

```
CREATE TRIGGER status_change AFTER UPDATE OF payment_method ON
      (reservation)
REFERENCING NEW ROW AS nrow
FOR EACH ROW
BEGIN ATOMIC
      UPDATE reservation
      SET reservation_status = ‘waiting’
      WHERE (reservation.pessenger_id = nrow.pessenger_id)
            AND (reservation.ride_id = nrow.ride_id)
```

5.4. Constraints

- All users must be at least eighteen years old. This constraint can be checked when user enters its birthday. If (calendar date - this entered birthday) is greater than or equal to 18 years old, user can join.
- If a driver tries to enter its car whose number of seats exceeds a threshold, this data entry will not be allowed. This is due to health and traffic safety concerns. No vehicle is allowed to carry nine passengers. This threshold depends on the type of the vehicle.

5.5. Stored Procedures

- For this car sharing system, one of the most demanded feature is expected to be that passengers want to find the driver with highest average rating for the itinerary of their choice. A parameterized view having the inputs of arrival and departure points may yield this intended result.

- The review mechanism is the fundamental part of this system. When a user visits another user's profile page, visitor most likely would like to see the lowest rating reviews to have an idea about visited user. A function with the parameter of user_id that returns the reviews based on ratings ascending order should work for this purpose.

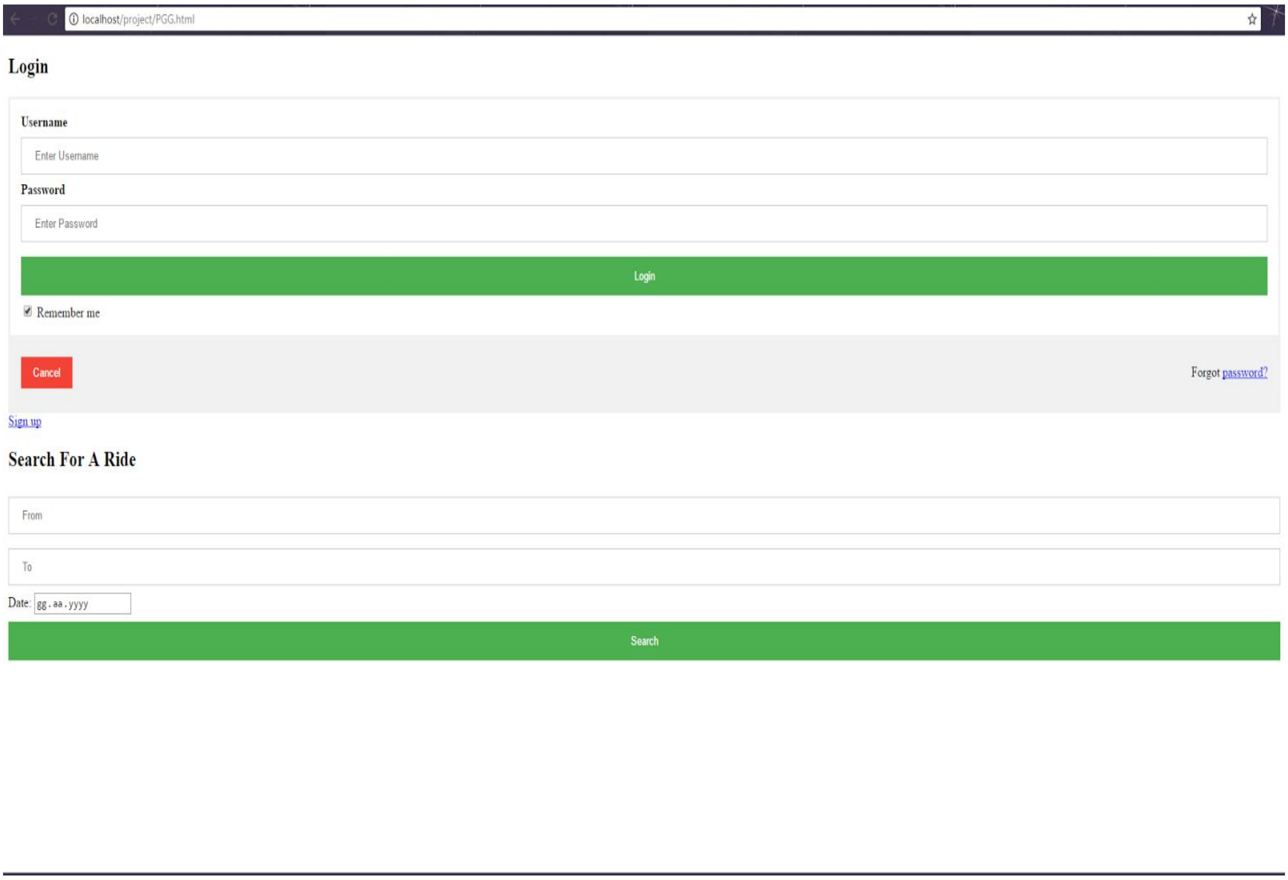
5.6. Secondary Indices

Considering the ride table, there will be many searches on the ride_id field. If a passenger wants to look for any rides, the system will look up on already created rides. To make on reservation on them, ride_id is needed. When drivers and passengers review system is heavily depending on ride table. Their historical data for the rides they created or participated basicly is those rides' ride_id. For these reasons, we construct a secondary index for this field in the following way:

```
CREATE INDEX ride_id_index USING BTREE ON ride (ride_id);
```

6. User’s Manual

6.1. Login Page



localhost/project/PGG.html

Login

Username

Enter Username

Password

Enter Password

Login

☑ Remember me

Cancel

Forgot password?

Sign up

Search For A Ride

From

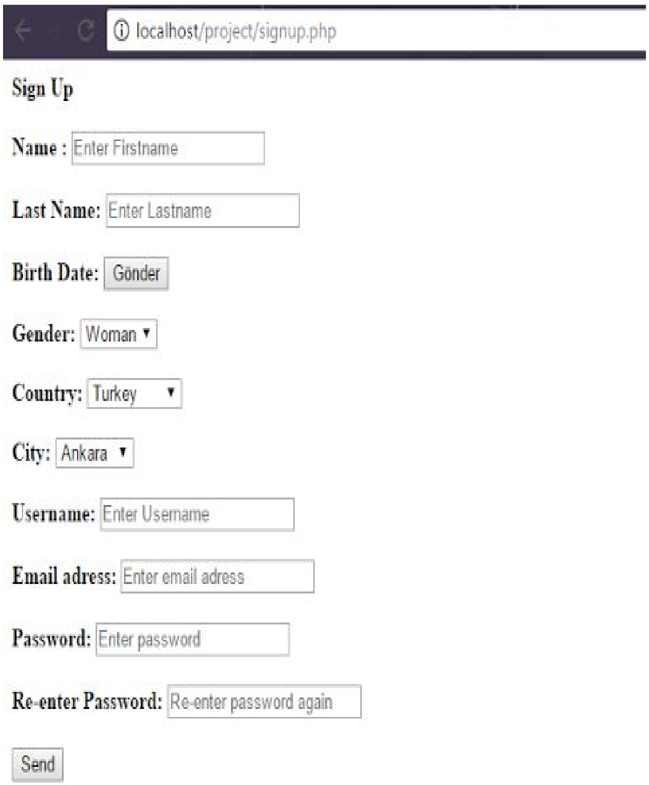
To

Date: gg . aa . yyyy

Search

The home page will include Login Form and Search for a ride form. If user has account already s/he can directly go through Home page. Or if they are new to Galaxy they can search for a ride and if they find one PGG will direct them to SignUp page.

6.2. Signup Page



localhost/project/signup.php

Sign Up

Name : Enter Firstname

Last Name: Enter Lastname

Birth Date: Gönder

Gender: Woman

Country: Turkey

City: Ankara

Username: Enter Username

Email adress: Enter email adress

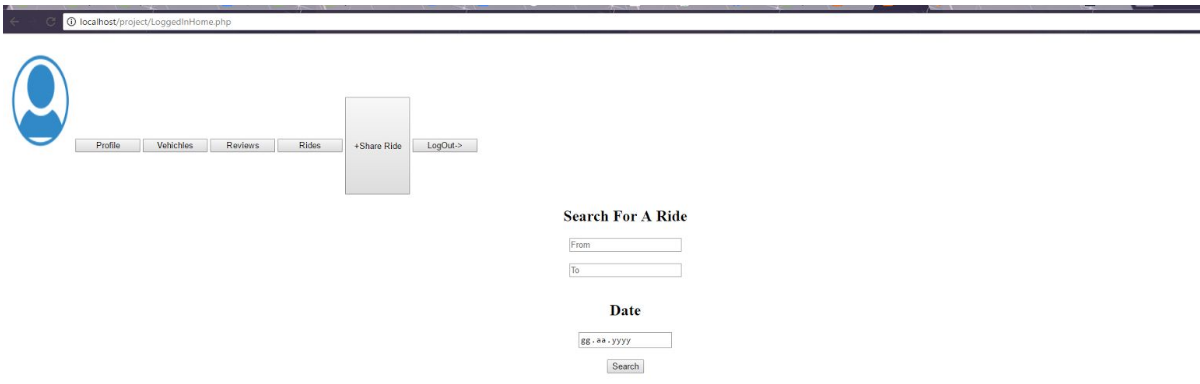
Password: Enter password

Re-enter Password: Re-enter password again

Send

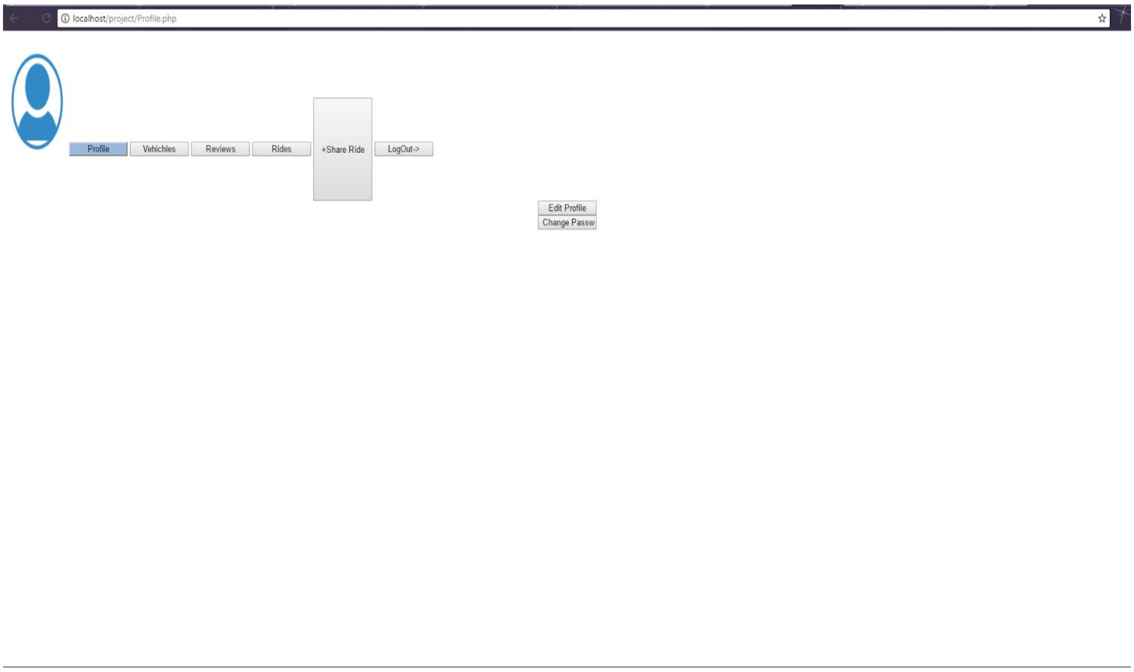
In signup page we take all the information about the use. Some parts are required like name exc. After user enters all information when they click save, they are directed to their new profile page.

6.3. Logged-in Page



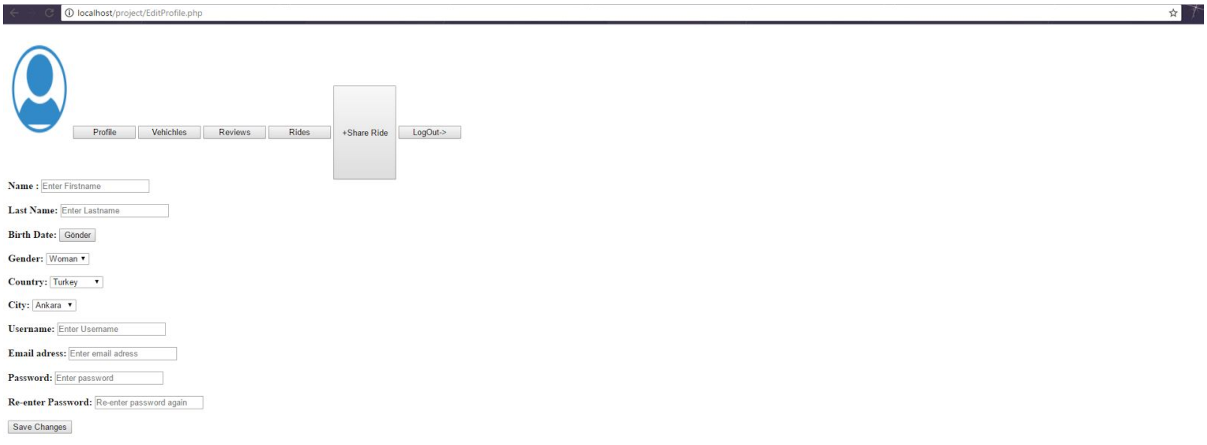
Logged in page is basically which user has options to see their profile information by clicking Profile button or their vehicles by clicking Vehicles button, same applies for rides and reviews as well. So mainly PGG lets user search for a ride in this section. If they want to see other properties they just us upper buttons bar.

6.4. Profile Page



In this part while user can see their information, additionally they have two option where they can perform editing profile informations and changing their password.

6.5. Edit Profile Page



Name : Enter Firstname

Last Name: Enter Lastname

Birth Date: Gönder

Gender: Woman

Country: Turkey

City: Ankara

Username: Enter Username

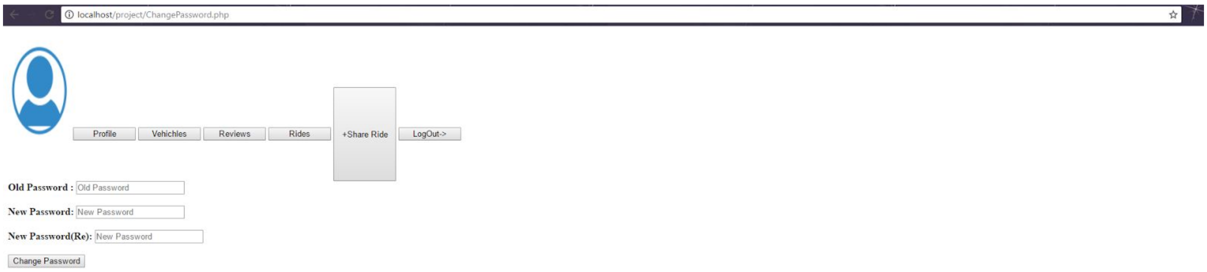
Email adress: Enter email adress

Password: Enter password

Re-enter Password: Re-enter password again

In this part user will fulfill all the information which they want to update. When they click save changes they will direct to Profile page again which will show updated information about the user.

6.6. Change Password Page


localhost/project/ChangePassword.php

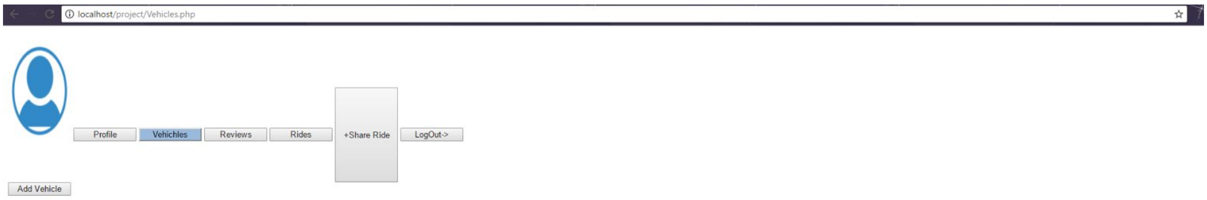
Old Password : Old Password

New Password: New Password

New Password(Re): New Password

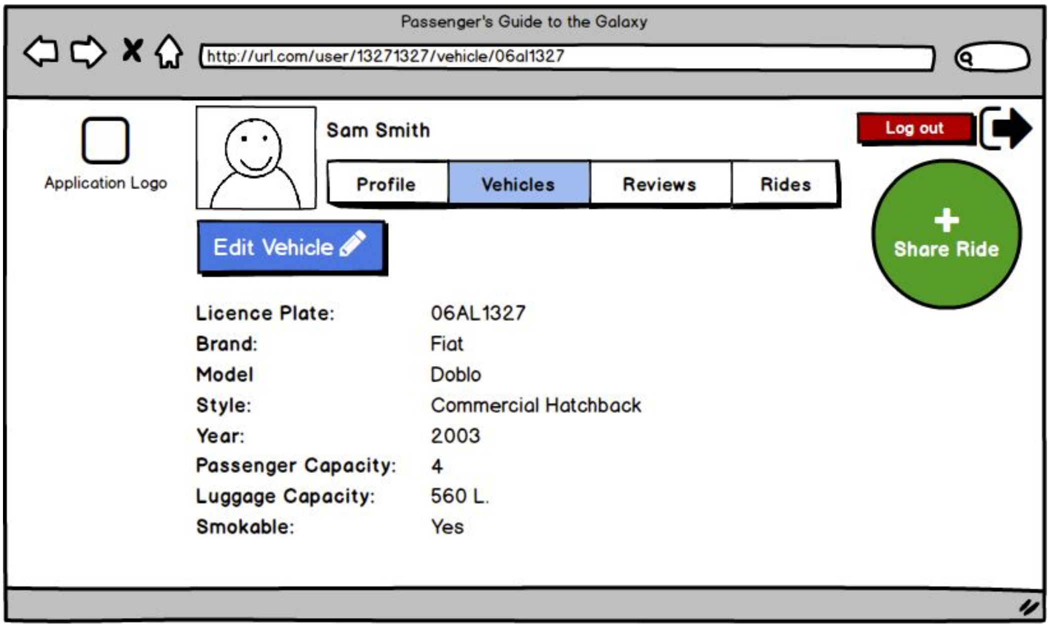
User can basically enter old password and new password here. Than need to re-enter new password again to check the correctness.

6.7. Vehicles Page


localhost/project/Vehicles.php

In this part user can see their vehicles. They can update their vehicles or they can add a new vehicle.

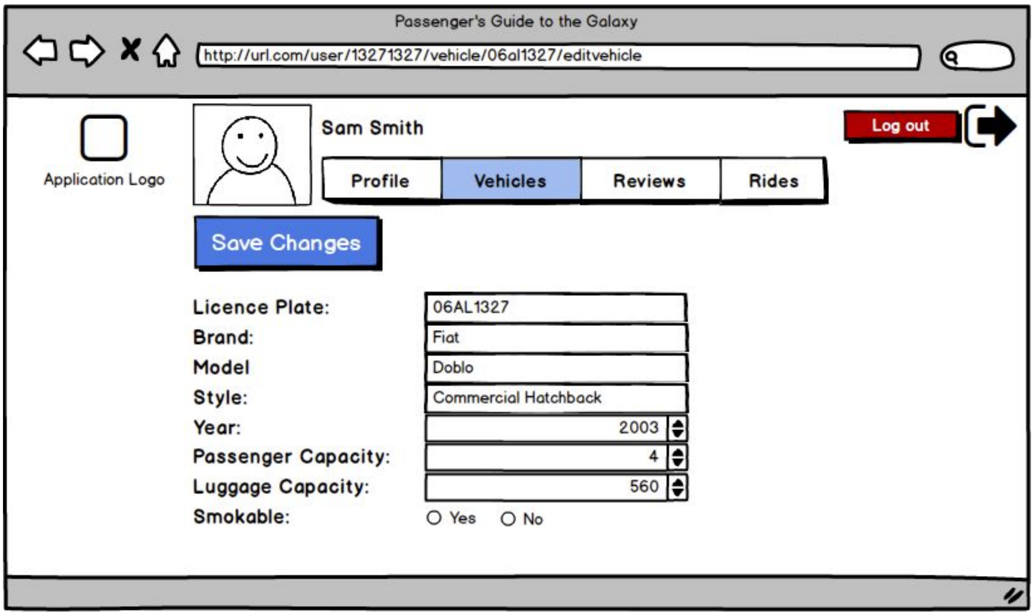
6.8. Single Vehicle Page



Licence Plate:	06AL1327
Brand:	Fiat
Model	Doblo
Style:	Commercial Hatchback
Year:	2003
Passenger Capacity:	4
Luggage Capacity:	560 L.
Smokable:	Yes

Same properties like update a vehicle, add a vehicle, delete a vehicle applies for this page as well. Only difference if user has one vehicle than PGG shows it directly with its' properties.

6.9. Edit Vehicle Page

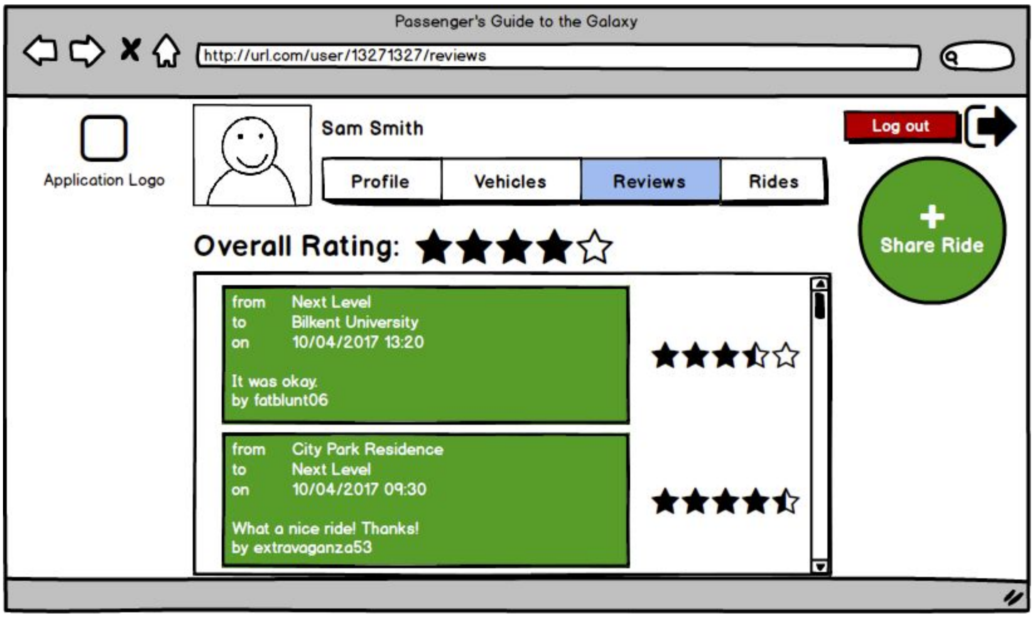


Passenger's Guide to the Galaxy
http://url.com/user/13271327/vehicle/06al1327/editvehicle
Application Logo
Sam Smith
Log out
Profile
Vehicles
Reviews
Rides
Save Changes

Licence Plate:	06AL1327
Brand:	Fiat
Model	Doblo
Style:	Commercial Hatchback
Year:	2003
Passenger Capacity:	4
Luggage Capacity:	560
Smokable:	○ Yes ○ No

User will be able to update the properties they want about the chosen vehicle.

6.10. Reviews Page



from Next Level
to Bilkent University
on 10/04/2017 13:20

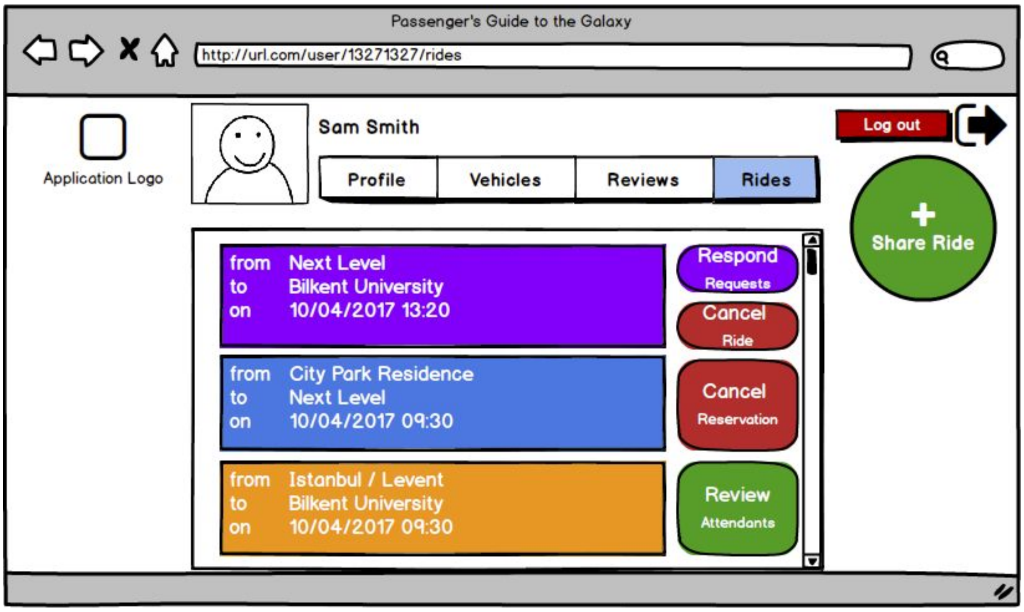
It was okay.
by fatblunt06

from City Park Residence
to Next Level
on 10/04/2017 09:30

What a nice ride! Thanks!
by extravaganza53

User will be able to see their reviews in this part. Or if a user searches for a drive after they go to chosen driver profile they can see their reviews in this part.

6.11. Rides Page



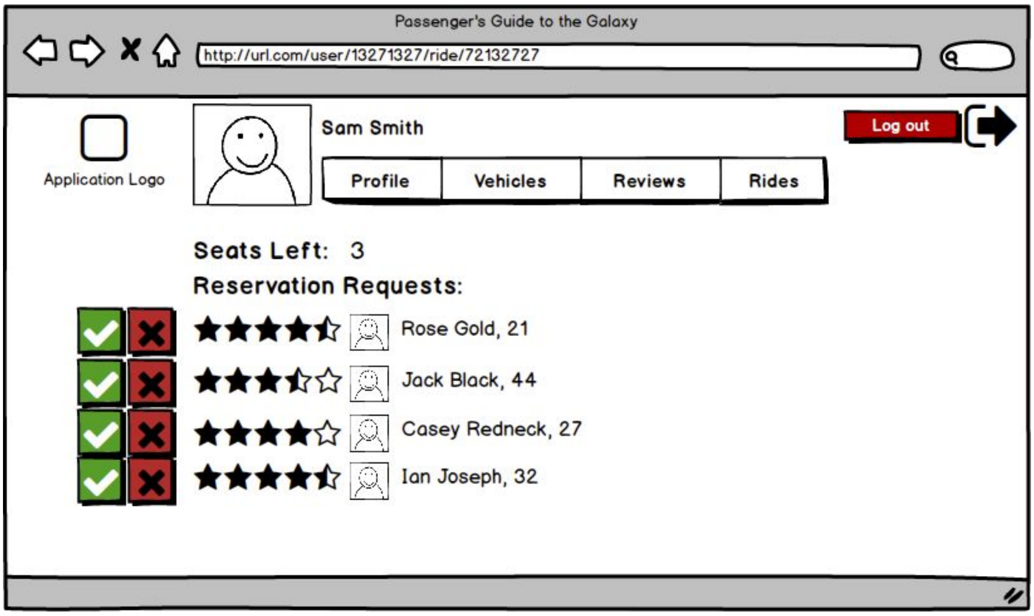
from Next Level
to Bilkent University
on 10/04/2017 13:20

from City Park Residence
to Next Level
on 10/04/2017 09:30

from Istanbul / Levent
to Bilkent University
on 10/04/2017 09:30

Driver by own or another user can check the rides the specific driver has. And the owner of the profile can update, delete or add rides in this page.

6.12. Reservation Request Page


Passenger's Guide to the Galaxy
http://url.com/user/13271327/ride/72132727
Application Logo
Sam Smith
Log out
Profile
Vehicles
Reviews
Rides

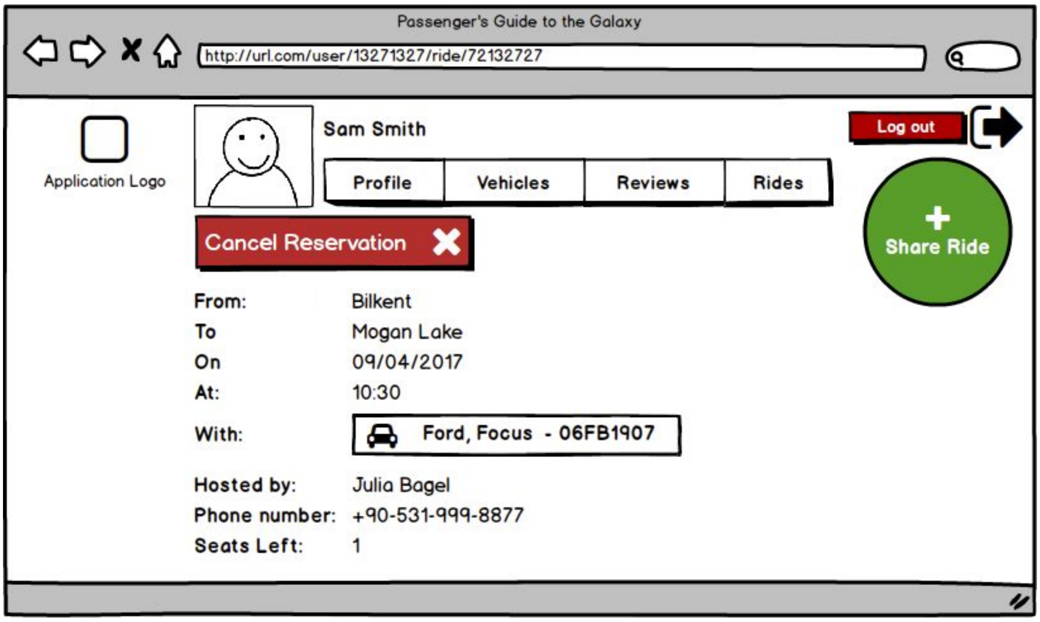
Seats Left: 3

Reservation Requests:

- ✓ ✗ ★★★★⯪ Rose Gold, 21
- ✓ ✗ ★★★⯪☆ Jack Black, 44
- ✓ ✗ ★★★★☆ Casey Redneck, 27
- ✓ ✗ ★★★★⯪ Ian Joseph, 32

This will be basically reservation request page which user has options between the different drivers for same ride.

6.13. Reserved Single Ride Page



From:	Bilkent
To	Mogan Lake
On	09/04/2017
At:	10:30
With:	Ford, Focus - 06FB1907
Hosted by:	Julia Bagel
Phone number:	+90-531-999-8877
Seats Left:	1

This page shows the reserved ride if there is only 1 ride reserved