# COMP 353: Databases

Instructor: Desai Bipin

By

| | |
|---|---|
| Marc-Andre Dragon(* Leader *) | 27721579 |
| Mihai Damaschin | 27177895 |
| Andres Kebe | 26638422 |
| Andres Nunnez | |
| Tom Chen | |

Demo on December 2nd 2016
Due Date December 9th 2016

# Project Description

Transportation or lack of is a common problem with the increasing number of responsibilities individuals of an evolving society shoulder. Team Alpha therefore presents you this web application designed to give its users an interface where they can easily plan their trips.

The users invited to join the application will register their information with the aid of a referral code supplied by another user or given by the founders of the software. Once registered in the system, the user can assume the role of either a rider or a driver.

The drivers form a community of individuals that have access to vehicles and provide users with possible trips from a certain origin, towards a specific destination at a certain time. The cost of the trip is remunerated using a specific formula available on the website that was the result of studies, researches and peer review.

The riders are a group of users that take advantage of the provided rides by the drivers to plan out their transportation needs. They are prompted to pay for the ride before hand and it is the responsibility of the rider to arrive on time and plan for possible luggage and food.

# The Assumptions

Multiple assumptions are made in order for the system to be function. This section will be separated into assumptions made for users, for drivers and for riders. Assumptions for users are a combination of the assumptions made for both drivers and riders and will often be in regards to the functionality of the website.

**User**:

- ➢ The user has an HTML5 compliant web browser capable of receiving data, loading CSS 4 and Javascript 1.8
- ➢ The user has a valid referral name as well as a corresponding referral code to be able to register into the system
- ➢ The user understands English
- ➢ The users possesses the hardware necessary to enter inputs and navigate the website.
- ➢ The users has an address, a name and information.
- ➢ The user possesses a paypal, a credit card or another form of online payment method or withdrawal method.
- ➢ The user has a valid email address.
- ➢ Users are above the age of 18.

**Rider:**

- ➢ The rider has enough money for the transaction
- ➢ The rider can get to the point of origin
- ➢ The rider has communicated what amount of extra luggage they have

**Driver**:

- ➢ The driver must have a functional vehicle
- ➢ The driver must have a vehicle that supports the number of riders to be
- ➢ The driver has enough room for the riders’ luggage
- ➢ The driver must have a valid insurance and permit
- ➢ The driver must show up on time at the origin spot
- ➢ The driver gets the riders to their destination safely

# The Limitations

Some limitations to the web applications are present. Some regarding hardware others regarding functionality due to a project description.

The user must have a piece of hardware capable of running a browser that is HTML-5 compliant with CSS 4 and Javascript 1.8. The user must have access to the internet to run the application.

Riders must also abide to the driver’s route. Special scenarios may happen at the users’’ discretion which is not covered by the company. The price of a route is decided by the system administrator and not the rider or the driver. These were designed as features for our application.

# The applications supported

Many applications are supported with the website regrouping 3 kinds of users:

**The Rider**:

- ➢ Can search for a ride using origin and destination search tools
- ➢ Can message the driver

**The Driver**:

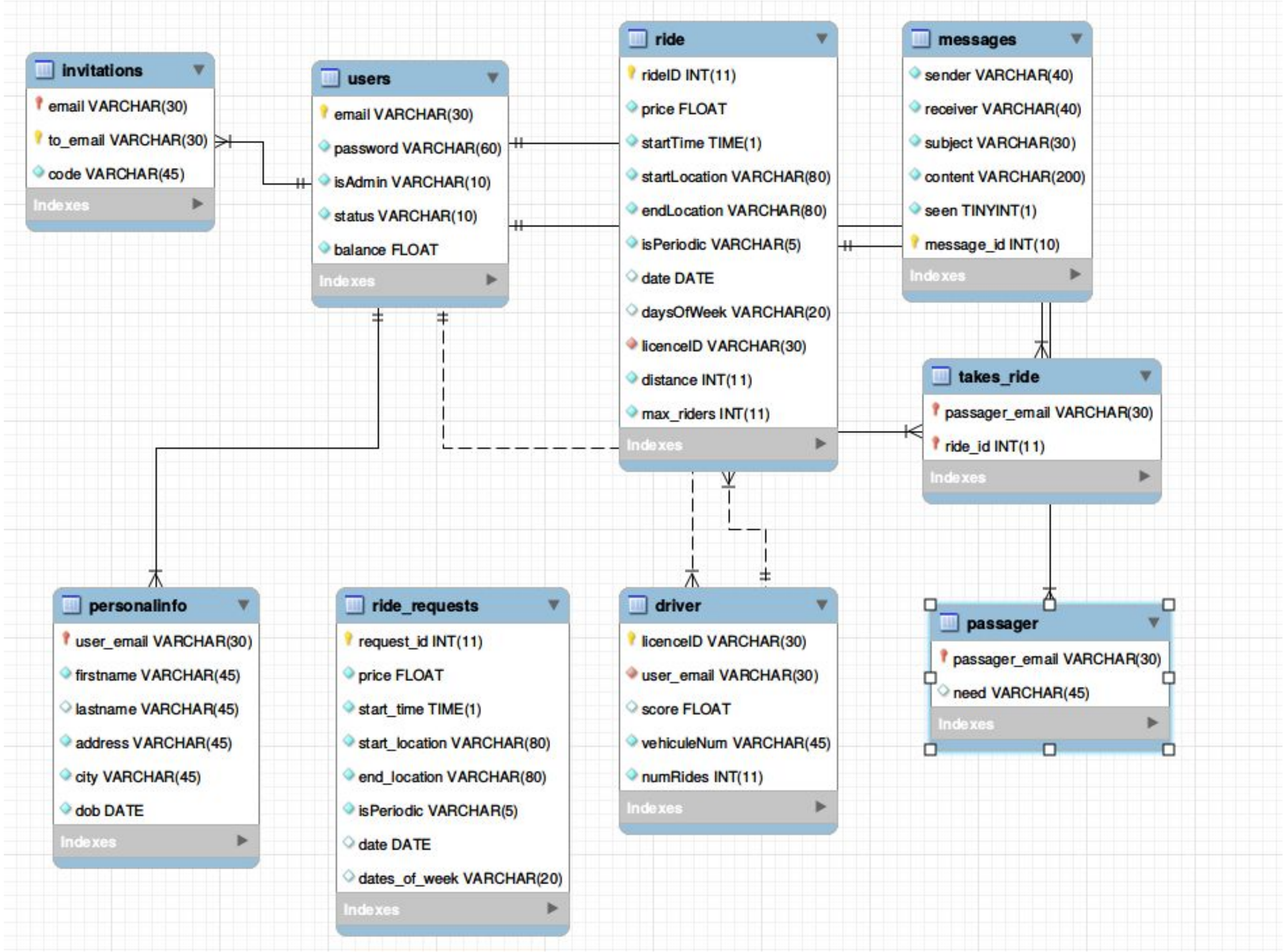
- ➢ Can establish a ride
- ➢ Can delete a ride (can he?)
- ➢ Can message the rider(s)
- ➢ Can

**The Admin**:

- ➢ Can delete a ride
- ➢ Can modify a ride
- ➢ Can suspend users
- ➢ Can modify users’ information

# Architectural Design

# E-R Diagrams



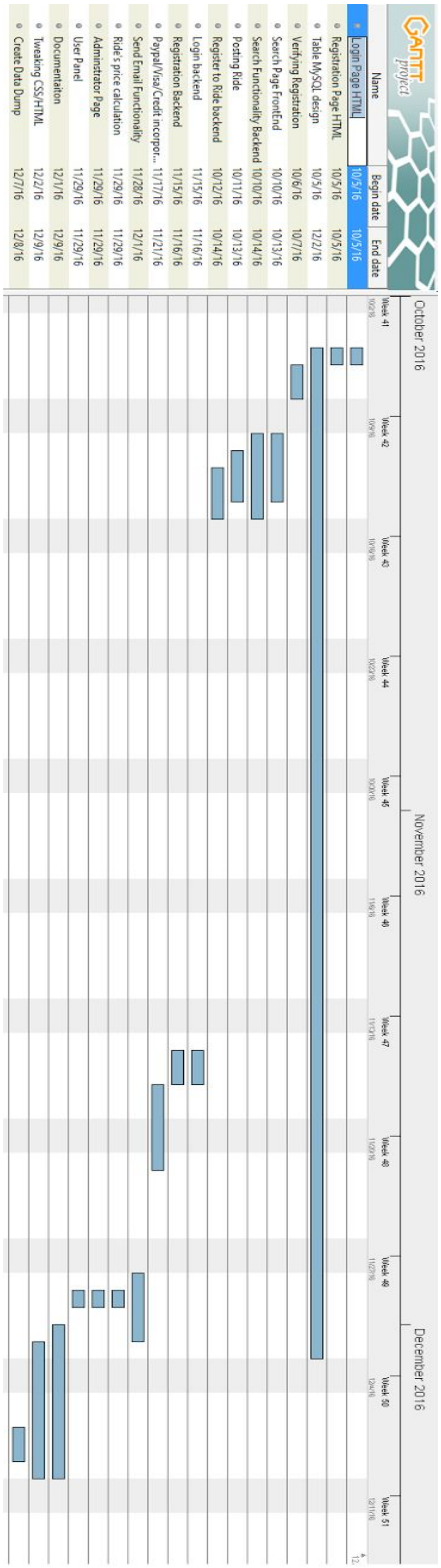
# Relational Database Design (I’m not sure what’s the difference between this and E-R Diagram tbh - I think it might be the table you guys generated above this)

# 3NF Solution

# Member  Responsibility

| Members | Responsability |
|---|---|
| **Mihai Damaschin** | Javascript incorporation, front end programming aid, documentation supervision |
| **Tom Chen** | Front End Programming |
| **Marc-Andre Dragon (*Leader*)** | Database Architect, Team Leader |
| **Andres Nunnez** | Database aid, back-end implementor of the front end, PHP programming |
| **Andres Kebe** | Front End Programming |

# Gantt Chart (see document for photo not restrained by margins)



| Name | Begin date | End date |
|---|---|---|
| Login Page HTML | 10/5/16 | 10/5/16 |
| Registration Page HTML | 10/5/16 | 10/5/16 |
| Table MySQL design | 10/5/16 | 12/2/16 |
| Verifying Registration | 10/6/16 | 10/7/16 |
| Search Page FrontEnd | 10/10/16 | 10/13/16 |
| Search Functionality Backend | 10/10/16 | 10/14/16 |
| Posting Ride | 10/11/16 | 10/13/16 |
| Register to Ride backend | 10/12/16 | 10/14/16 |
| Login backend | 11/15/16 | 11/16/16 |
| Registration Backend | 11/15/16 | 11/16/16 |
| Paypal/Visa/Credit incorpor... | 11/17/16 | 11/21/16 |
| Send Email Functionality | 11/28/16 | 12/1/16 |
| Ride's price calculation | 11/29/16 | 11/29/16 |
| Administrator Page | 11/29/16 | 11/29/16 |
| User Panel | 11/29/16 | 11/29/16 |
| Documentaiton | 12/1/16 | 12/9/16 |
| Tweaking CSS/HTML | 12/2/16 | 12/9/16 |
| Create Data Dump | 12/7/16 | 12/8/16 |

# Detailed Analysis of coding the website

## Front End Programming:

**Programming Languages:**

- HTML5 (Hypertext Markup Language)
- CSS 4 (Cascading Style Sheets)
- Javascript 1.8

**Framework:**

- Foundation 5

## Back End Programming:

**Programming Languages:**

- PHP 7.1 ( Previously, Personal Home Page, now “Hypertext Preprocessor”)
- SQL (Structured Query Language)

**Relational Database Management System:**

- MySQL aided with the tool PHPMyAdmin

## Explanation of Files by Folders:

- core.php:
- DeleteUser.php:
- GoogleMapsTest.html: This was made as a test to see how to pull google’s  API for maps
- header.php:
- home.php:
- index.html:
- index.php:
- login.html: This was a prototype of the login page that was discarded during further iterations.
- README.md: Readme of Github repo
- Register.html: This was a prototype for the registration that was discarded during further iterations.
- register.php:
- registerVerification.php:
- stylesheet.css:
- super_db.sql:
- test.php:

- userpanel.html: Userpanel serving as profile and control hub for the user.
- /backend/*
  - dbconnection.php:
  - dbFunctions.php:
  - logout.php:
  - postRideController.php:
  - setup.php:
- /css/*
  - animate.css: Open source library used to animate classes inside of HTML files.
  - app.css:
  - foundation.css:
  - foundation.min.css:
  - styleMihai.css: Styles made for the prototype login and registration page.
- /Documentation/* is self-explanatory
- /images/*
  - Anonymous.jpg: Image showing profile picture by default of users.
  - photo_bg.jpg: Background picture of old prototype of login and registration page.
- /info/*
  - DB design.vsdx:
  - Super_db.sql:
- /js/*
  - /vendor/*
    - foundation.js: Foundation imported javascript library.
    - foundation.min.js: Minified Foundation imported javascript library.
    - jquery.js: JQuery library.
    - what-input.js:
  - app.js
- /style/*
  - car.png:
  - large_car.png:
  - pedo.jpg:
  - small_car.png:
  - style.css:
- /vendor/*
  - /composer/*:
  - /stripe/*:
  - autoload.php:

## The interface design rational

## Queries

## User manual

## General information

The system was made with a combined effort to provide a smooth and intuitive interface for the Suber web application. The riders and drivers interact with each other through our inbox system while our system sets the price automatically.

## Authorized use Permissions

Here displayed are a list of preconceived users with or without administrative power that were tested on our web application:

Please see previous chapter “application supporter” to understand the power of the admins compared to the members.

| Admins | | Members | | |
|---|---|---|---|---|
| **Username** | **Password** | **Username** | **Password** | **Rider/Drive** |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

| | | | | |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

## System Summary

## Overview of the user interface

## Home page

## VISITORS

## MEMBERS

## ADMINISTRATOR

## Git Log

## Database

## Skip to create the database and populate it

# References