# Car Rentals

**Jacob Geddings and Stewart Rodger,**
**http://people.oregonstate.edu/~geddingj/cs340/final/index.php**

## 1. Introduction

Our website is a car rental site. From the site you can view the available cars, sorted by make, color, or location (all cars are located at a physical office). Anyone can create a members account, and as a member you can chose to rent a car (only one car per account can be rented at a time) and to leave a review for that specific car. Specific cars can be selected to view the various reviews previous renters have left for it.

Potential users would be people making money but have no car, and are only looking to rent one temporarily. Examples include people whose car is in the shop, travelers (people who came in by plane and are looking for something more reliable than public transport), and people who need a car but do not feel that now is the right time to buy.

## 2. Detailed Functionality & Requirements

Requirements:

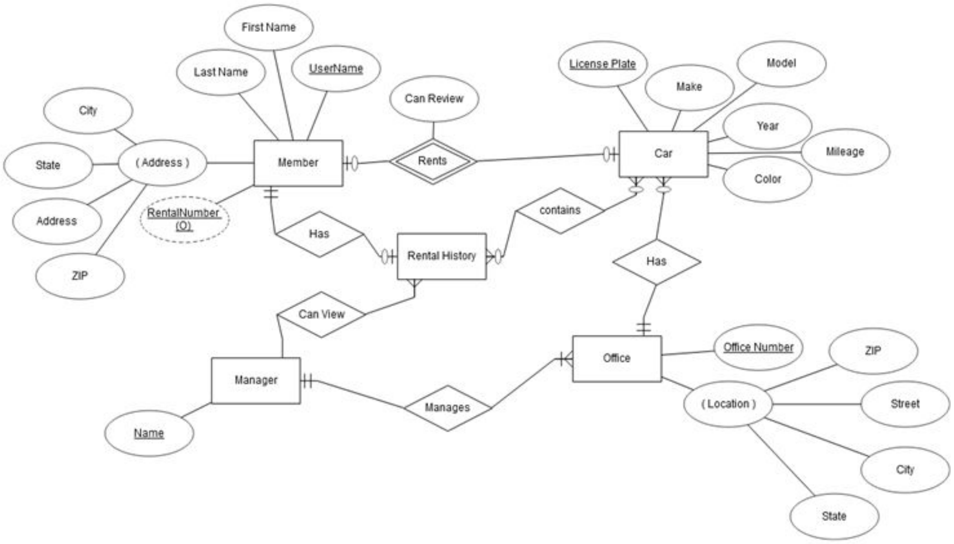
- Users can create and account and sign-in / sign-out
- Option to not log-in for viewing but no renting
- Locations of company listed
- Vehicles stored at each location listed
- Cars listed with information about them (color, mileage, year, make, model)
- Ability for user to register for a vehicle
- Ability for user to review their service

Business Rules:

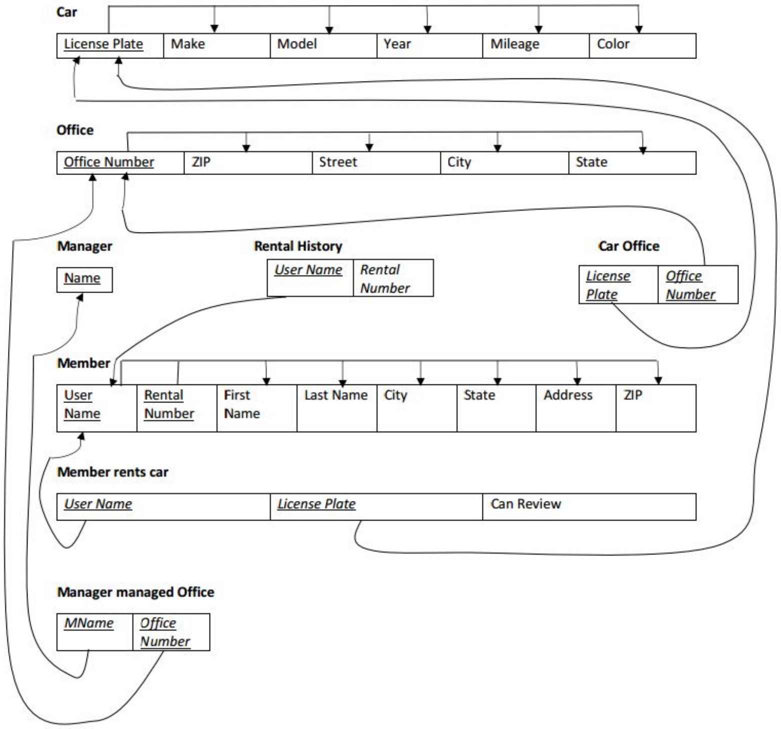
- User can only have one active rental at a time
- A car may only be rented out to one user at a time.
- Renting can only be done by user
- User can only review the vehicle they've rented
- No login necessary to view locations and inventory
- One review per vehicle per user
- All rental cars must be a part of an office.

## 3. Database Design

- ***ER Diagram of Database***



- ***Relation Schema***



**Car**

| License Plate | Make | Model | Year | Mileage | Color |
|---|---|---|---|---|---|

**Office**

| Office Number | ZIP | Street | City | State |
|---|---|---|---|---|

**Manager**

| Name |
|---|

**Rental History**

| *User Name* | *Rental Number* |
|---|---|

**Car Office**

| *License Plate* | *Office Number* |
|---|---|

**Member**

| User Name | Rental Number | First Name | Last Name | City | State | Address | ZIP |
|---|---|---|---|---|---|---|---|

**Member rents car**

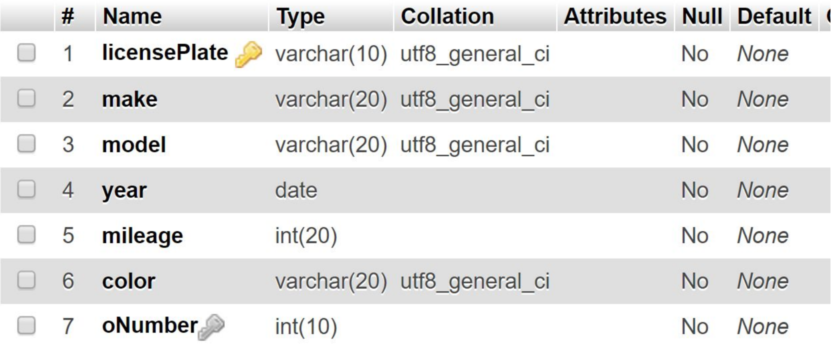
| *User Name* | *License Plate* | Can Review |
|---|---|---|

**Manager managed Office**

| *MName* | *Office Number* |
|---|---|

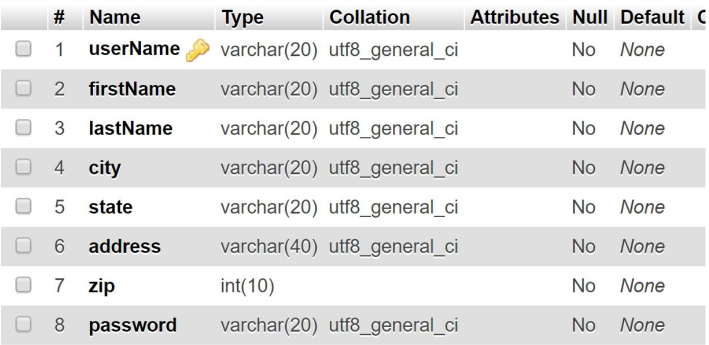
- ***Database Tables***

Car:



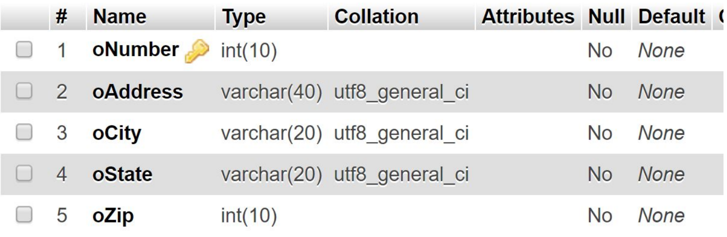
| | # | Name | Type | Collation | Attributes | Null | Default | ( |
|---|---|---|---|---|---|---|---|---|
| □ | 1 | **licensePlate** 🔑 | varchar(10) | utf8_general_ci | | No | *None* | |
| □ | 2 | **make** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 3 | **model** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 4 | **year** | date | | | No | *None* | |
| □ | 5 | **mileage** | int(20) | | | No | *None* | |
| □ | 6 | **color** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 7 | **oNumber** 🗝 | int(10) | | | No | *None* | |

Members:



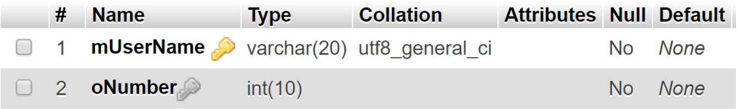
| | # | Name | Type | Collation | Attributes | Null | Default | C |
|---|---|---|---|---|---|---|---|---|
| □ | 1 | **userName** 🔑 | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 2 | **firstName** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 3 | **lastName** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 4 | **city** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 5 | **state** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 6 | **address** | varchar(40) | utf8_general_ci | | No | *None* | |
| □ | 7 | **zip** | int(10) | | | No | *None* | |
| □ | 8 | **password** | varchar(20) | utf8_general_ci | | No | *None* | |

Office:



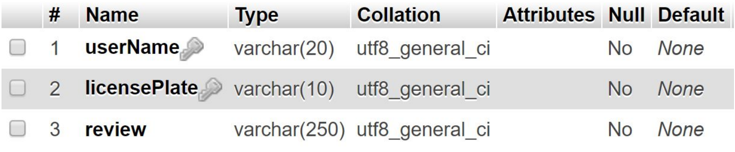
| | # | Name | Type | Collation | Attributes | Null | Default | ( |
|---|---|---|---|---|---|---|---|---|
| □ | 1 | **oNumber** 🔑 | int(10) | | | No | *None* | |
| □ | 2 | **oAddress** | varchar(40) | utf8_general_ci | | No | *None* | |
| □ | 3 | **oCity** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 4 | **oState** | varchar(20) | utf8_general_ci | | No | *None* | |
| □ | 5 | **oZip** | int(10) | | | No | *None* | |

Manager:



| | # | Name | Type | Collation | Attributes | Null | Default |
|---|---|---|---|---|---|---|---|
| □ | 1 | **mUserName** 🔑 | varchar(20) | utf8_general_ci | | No | *None* |
| □ | 2 | **oNumber** 🗝 | int(10) | | | No | *None* |

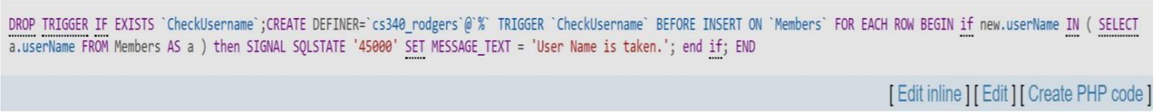
RenterHistory:



| | # | Name | Type | Collation | Attributes | Null | Default |
|---|---|---|---|---|---|---|---|
| □ | 1 | **userName** 🗝 | varchar(20) | utf8_general_ci | | No | *None* |
| □ | 2 | **licensePlate** 🗝 | varchar(10) | utf8_general_ci | | No | *None* |
| □ | 3 | **review** | varchar(250) | utf8_general_ci | | No | *None* |

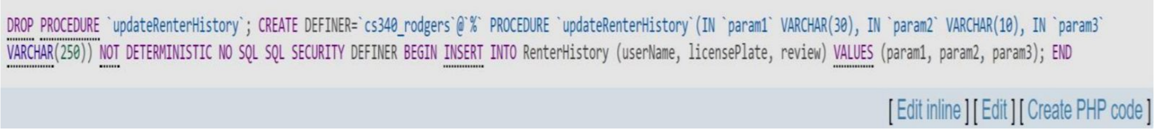
Functions/triggers/procedures:

Trigger that prints a message telling the user if they tried to create an account with a taken username:



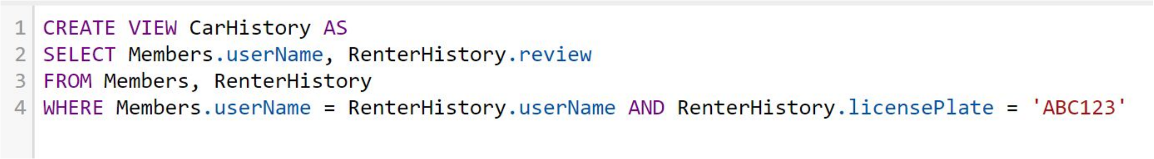
```
DROP TRIGGER IF EXISTS `CheckUsername`;CREATE DEFINER=`cs340_rodgers`@`%` TRIGGER `CheckUsername` BEFORE INSERT ON `Members` FOR EACH ROW BEGIN if new.userName IN ( SELECT
a.userName FROM Members AS a ) then SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'User Name is taken.'; end if; END
```

Procedure that can be used to add the relevant data to RenterHistory when
the user rents a car:



```
DROP PROCEDURE `updateRenterHistory`; CREATE DEFINER=`cs340_rodgers`@`%` PROCEDURE `updateRenterHistory`(IN `param1` VARCHAR(30), IN `param2` VARCHAR(10), IN `param3`
VARCHAR(250)) NOT DETERMINISTIC NO SQL SQL SECURITY DEFINER BEGIN INSERT INTO RenterHistory (userName, licensePlate, review) VALUES (param1, param2, param3); END
```

View (for the reviews of a specific car):

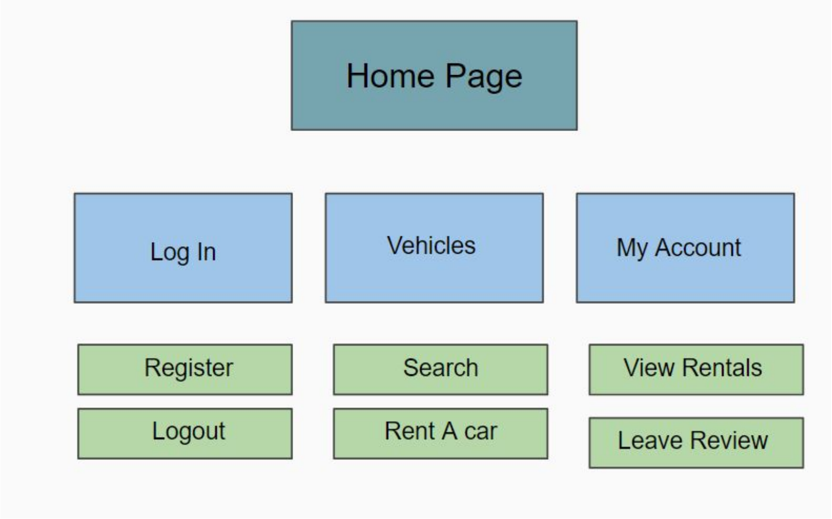


```
CREATE VIEW CarHistory AS
SELECT Members.userName, RenterHistory.review
FROM Members, RenterHistory
WHERE Members.userName = RenterHistory.userName AND RenterHistory.licensePlate = 'ABC123'
```

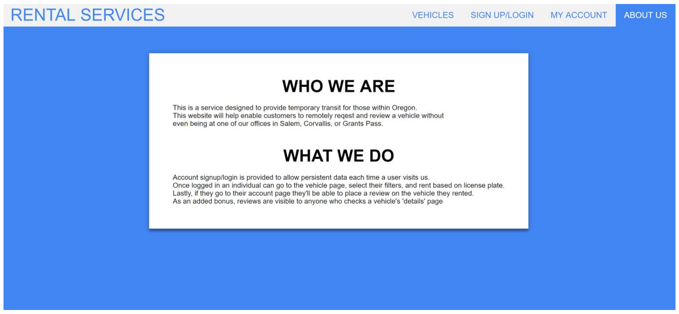
## 4. Website Design

The user is first directed the the home page, with an ‘About Us’ section detailing the website's functions and purpose. From there, the user Can navigate freely between the 3 other pages and access their respective functionalities.

- ***Website Layout***



- **User Interface (website wireframe/diagram)**

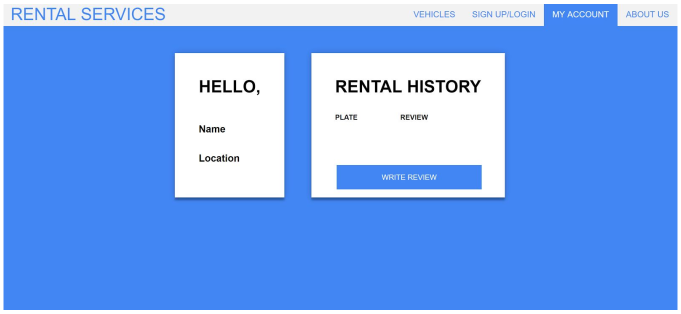


## WHO WE ARE

This is a service designed to provide temporary transit for those within Oregon.
This website will help enable customers to remotely reqest and review a vehicle without
even being at one of our offices in Salem, Corvallis, or Grants Pass.

## WHAT WE DO

Account signup/login is provided to allow persistent data each time a user visits us.
Once logged in an individual can go to the vehicle page, select their filters, and rent based on license plate.
Lastly, if they go to their account page they'll be able to place a review on the vehicle they rented.
As an added bonus, reviews are visible to anyone who checks a vehicle's 'details' page



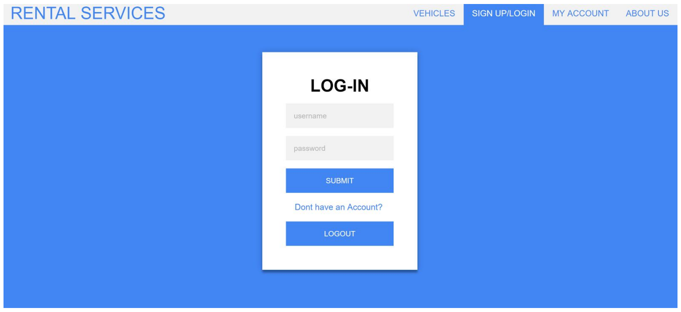
## HELLO,

Name

Location

## RENTAL HISTORY

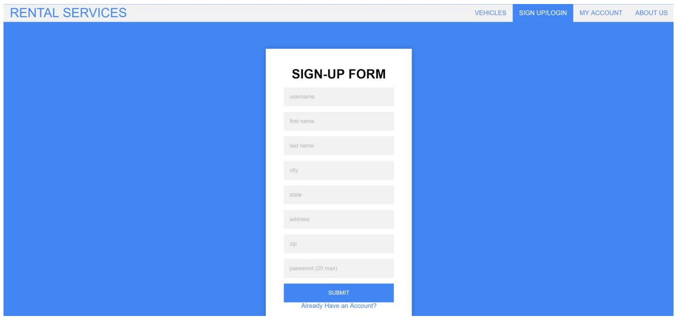
| PLATE | REVIEW |
|---|---|



## LOG-IN

username

password



## SIGN-UP FORM

username

first name

last name

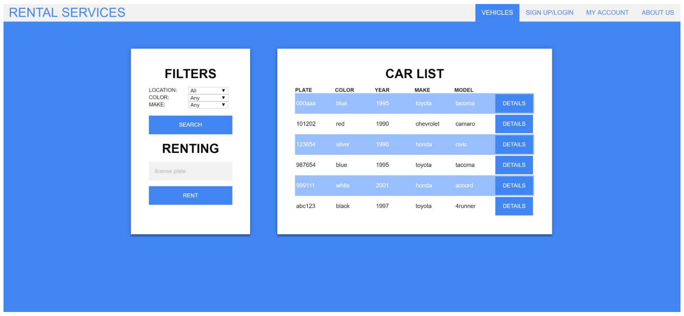
city

state

address

zip

password (20 max)



## FILTERS

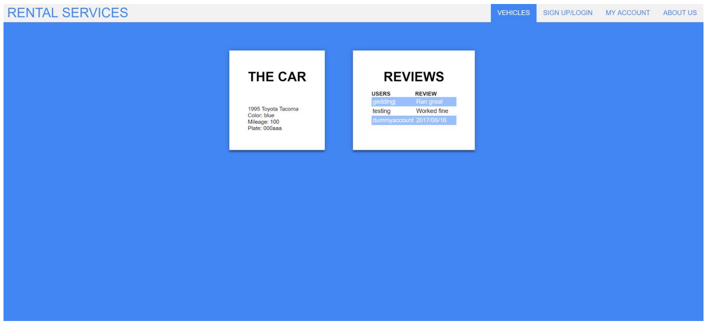
LOCATION: All ▼
COLOR: Any ▼
MAKE: Any ▼

## RENTING

license plate

## CAR LIST

| PLATE | COLOR | YEAR | MAKE | MODEL | |
|---|---|---|---|---|---|
| 000aaa | blue | 1995 | toyota | tacoma | DETAILS |
| 101202 | red | 1990 | chevrolet | camaro | DETAILS |
| 123654 | silver | 1990 | honda | civic | DETAILS |
| 987654 | blue | 1995 | toyota | tacoma | DETAILS |
| 999111 | white | 2001 | honda | accord | DETAILS |
| abc123 | black | 1997 | toyota | 4runner | DETAILS |



THE CAR

1995 Toyota Tacoma
Color: blue
Mileage: 100
Plate: 000aaa

REVIEWS

| USERS | REVIEW |
|---|---|
| geddingj | Ran great |
| testing | Worked fine |
| dummyaccount | 2017/06/16 |

- ***User Manual or Help page***

1. From the About Us page, select the “Sign up/Log In” tab at the top of the page.
2. If you have an account, sign in now, if not select “Don’t have an account?” below the submit button.
3. If you created a new account, select the “Sign up/Log In” tab again and enter your new account information.
4. Once you're logged in (you will see a welcome message if you logged in correctly) you can proceed to the “Vehicles” or “My Account” tabs.

FAQ:
Q1: Why can’t I rent a car?
A1: You must be logged in first, and you must enter a valid license plate number.

Q2:My account page isn’t showing any information?
A2: Make sure you are logged in.

## 5. Application Implementation

- Describe your use of HTML/PHP/CSS/JavaScript/….

The website is made entirely with PHP, using SQL to access the database and a simple CSS to provide a coherent style to the different pages. The back-end of the project has 4 distinct types of files, 1 connecter file (included in nearly every page, which simply connects to the database), 5 web pages (the code for the pages the user will connect to), 3 ‘helper’ files (used to handle DB requests more complicated than simply displaying a table), and 1 CSS file to handle the overarching visual design of the site. Of these, only the CSS file isn’t a .php

- Discuss your SQL queries

SQL queries are used to display the renters history on the “MyAccount” page, and to display the car lists with the user specified search constraints on the “Vehicles” page.

- Which of your application requirements the DBMS provides.

The DBMS is critical to all of the requirements, especially the login requirement and the ability to rent/review cars, which requires the use of the foreign key constraints in the DBMS.

## 6. Evaluation

To test the application, we were able to use equivalence partitioning in our tests. Namely, we were able to walk through all aspects of the website (both logged in and logged out) and manually test that the features were working as intended. We were able to skip testing every entry in a table, settling for just one or two, because all entries were equivalent for the tests.

We used traditional, common website design practices (nav links along the top and to the right, all important buttons or entry fields placed about center). This payed off as other people who had not seen the site before were able to navigate it easily and did not have any issues locating the various functions.

## 7. Future Work & Lessons Learned

One additional functionality that was intended to be implemented was a manager login and special manager account page, another was the ability to search reviews based on the user who reviewed it. The biggest challenge was time, with neither of us particularly experienced in SQL and with limited memory of PHP, solving problems was a learning process, and that can take up a lot of valuable development time. The best way to address these issues in the future would be to simply allow for more development time.

## *Appendix – Team Report* *If you worked in a team summarize the division of labor.*

Jacob Geddings:

All CSS styling, horizontal navigation bar, all pages excluding the ‘existing-user login’ page. All SQL queries except for ‘renting’ and ‘login’ where done by me, joint effort on ‘renting’ and ‘login’. This includes filter functionality, viewing individual details of a vehicle and their subsequent information/reviews, ability to create a new account, viewing of personal information and past rental history under ‘My Account’, and ability to write reviews on past rented vehicles under ‘My Account’.

Stewart Rodger:

Setting up the database structure, with appropriate foreign key constraints and primary keys, creating the visualizations of the database (both the ER diagram and the relational schema). Most of the ‘renting’ back-end, joint effort on ‘login’. Final testing and bug fixing, documentation, and writing up most of the peer review and project report.