Raphael Shejnberg IT231 Final Documentation

**Purpose**

This application was developed with the intent of serving as a replacement website for a local Chicago leasing company. Taking the original website structure and purpose in mind, this site is meant to mirror that but with a significantly greater degree of functionality and effectiveness. Simultaneously, a primary reason for recreating the site was to increase the ease of navigation for both the public, and the designated users responsible for posting content.

**Organization**

This web app is structured primarily into three components, ‘main’, ‘apartments’, and ‘users’. The apartments portion is a comprised of a model, controller, and several views that display the database contents and provides users with the ability to alter the database contents. The database in its public view is transformed into a more aesthetic appearance with the ability to display only apartments corresponding to the amount of bedrooms. The users portion corresponds with the sessions controller which namely provide the authentication process for the website. The login page accepts email and password as means to authenticate and also makes the administrative portion of the site private while allowing the rest of the views to remain public. Lastly the main portion is not robust in ruby coding but rather focuses on the client-side designing aspect. Using JQuery, I was able to set up an interactive and responsive public portion to the site that allowed ease of navigation for anyone visiting. In the public portion, the home view, multiple sections are made available and accessible without changing the site by utilizing .hide() and .show() functions. However, the apartments tab does change the site to a view located in the aforementioned apartments category. This is the only public portion of that category. The other part of the main views in the admin page, which is the page you are directed to upon logging in and can access both users and apartments tables.

**Development**

The development process began with setting up the models and controllers for users and apartments. After doing this I realized that I would need a separate set of views for both the public and admin homepages, which caused me to create ‘main’. After implementing sufficient Javascript in these pages to be able to navigate around the site, it was time to connect the main views with that databases that created the structure for the site. For the users portion, I needed to modify the form to allow people to enter a password and confirmation rather than the digest itself, which would not be too useful. For the apartments portion I installed an image handling gem called rmagick which allowed me to add an image uploader to the form. This gem also allowed for resizing the images, so that a smaller version would appear in the menu view and a larger one in the show view. In order to standardize form input I set-up variables in the apartment model to create drop-down menus in the form. After this I created several instance variables relating to the apartments table using the find module in the apartments controller which would give the public apartment view the ability to show apartments only with the selected amount of bedrooms. Showing these different tables was again handled using JQuery animating functions.

**Instruction**

The homepage is set to the root\_url . After going to this page one may view the public portion of the site by navigating through the green tags at the top. In order to login one must click the png of the house at the bottom of the page and they will be redirected. In order to login one may enter ‘admin@email’ and ‘pass’ to gain entry. After logging in they will be redirected to the admin homepage where you can click on either users or apartments in order to view and modify tables. By clicking the green sign/image on the left of this admin homepage one may logout.

*Links*

The homepage may be accessed at <http://localhost:3000/> after running the rails server. The admin login page is simply that with /login at the end. I semi-successfully managed to upload it using Heroku to this link <http://smooth-dusk-9878.herokuapp.com/#5> , however several major errors occurred causing an alternative view of main#home to be displayed and do not allow access to the administrative portion. Not sure the cause, however, it may be related to issues with the ‘sqlite3’ gem. I did as follows in the class lecture and Heroku instructions but was still receiving errors during the master git command and had to look for alternative methods.

**Individual Contributions**

Me and Patrick collaborated on the project collectively. I handled the structural aspects regarding the models, controllers, and views along with the designing portion and JS coding. As part of the structural aspects I set up the authentication process. Patrick set up validators throughout the project and also created relationships between the tables so that each apartment would belong to a user and each user had many apartments.