

March 4 2014

Housing4u Design Specification

Version 1.0

Prepared for CS189A
Chandra Krintz
Geoffrey Douglas

Members:

Site Mao
Jason Horsley
Austin Alameda
Alvin Tan
Trae Washburn

Table of Contents

1. Introduction
 - 1.1. Product Overview
 - 1.2. Definitions, Acronyms, Abbreviations
 - 1.3. Document Overview
2. Components
 - 2.1. Appfolio Property Data
 - 2.2. Postgres Database
 - 2.3. Ruby on Rails
 - 2.4. Heroku server
3. Design Specifications
 - 3.1. High Level Overview
 - 3.2. Account Creation and User Log-in
 - 3.3. Listing Properties for Owners
 - 3.4. Filtering Properties for Renters
4. Class Diagram
 - 4.1 UML Diagram
5. UI Mockups
6. Flow Diagrams
 - 6.1. General Use Flow
 - 6.2. User Authentication Flow

1. Introduction

1.1 Product Overview

Housing4u is a senior project being developed by 5 students at the University of California, Santa Barbara. The goal of the project is to provide a Web Application service in order to provide users with a means to search for rental housing, as well as for homeowners to list their property in search of tenants. Our application will provide a number of features such as selective filtering of house details, a personal watch list and several others with API integration of services such as Google Maps.

1.2 Definitions, Acronyms, Abbreviations

- API (Application Programming Interface) - A generalized interface used to allow Housing4U to interact with external software.
- Bootstrap - A front-end framework used to simplify user interface design efforts.
- CAPTCHA (Completed Automated Program to Tell Computers and Humans Apart) - Image-based authentication used to verify a human user and minimize spam.
- Front-end - The aspect of the application that the user interacts with.
- Back-end - The aspect of the application the user does not interact with. It is developed and maintained by development professionals.
- Gem - An external, packaged, 3rd-party tool used for back-end design.
- Heroku - An external, web-based platform used to deploy Housing4U on a live web server.
- HTTPS - encrypted communication protocol used to keep user sessions secure
- PostgreSQL - A back-end database tool used to store information.
- Ruby - The programming language Housing4U utilizes.
- Software Requirements Specification for Housing4U Page 2
- Rails - The web-based framework Housing4U utilizes, built on top of Ruby.
- UI (User interface) - Front-facing aspect of the Housing4U application.

1.3 Document Overview

This document will begin with overview of the components of our application. After, we will explain the different events that may occur in the system, including registration, login, authentication; we will also talk about to functionality of the system such as filtering and looking up listings. Then, we will show the class diagrams.

2. Components

Because we are using bootstrap to manage our front end, most of our product is the design and integration of the back end. The backend consists of using a Postgres database to store data that we acquired from Appfolio, and to create a Ruby of Rails application that interacts with the data so we can apply filters and transform the data in other ways.

2.1 Appfolio Property Data

The Appfolio property data is data on the customers of Appfolio that was provided to us. It is a large collection of information about houses with details such as photos, address and listing prices. This data is in XML format, and we parse this data and store it into a database. Because of this initial amount of information, we are able to develop and test our features even if our current user base is low.

2.2 Postgres Database

We use a Postgres database to store our property information. Initially, we populate this database using the Appfolio data. However, registered users on our website can also list their own property and be added to the database. Our back end will call the database to provide the user with the information needed to provide functionality.

2.3 Ruby on Rails

We use the framework Ruby on Rails to create our backend. It uses the programming language Ruby to create a web framework that is extremely versatile and simply to use. Rails is a modern web framework that uses add-ons called Gems that can solve many problems quickly and efficiently. It is very useful because it often allows one to skip doing redundant work that has been solved before.

Another benefit of Gems is that they are extremely easy to integrate into the system, and very difficult components of the Web Application can be completed very quickly so the developers can focus on parts deemed more important.

2.4 Heroku server

Heroku is a web service that provides hosting for web applications. Rather than using our own server to host our application, we decided to use heroku because not only do they streamline a large portion of pushing production code to a website, they provide a large amount of support for applications such as ours.

We will be doing our development locally and when necessary, pushing our production code onto the Heroku for demonstrations.

3. Design Specifications

3.1 High Level Overview

3.2 Account Creation and Log-in

In order to gain full functionality of the website, a user must first create an account and log in with it. We used a Ruby on Rails gem Devise to deal with user login and registration.

3.2.1 Successful User Registration

Users seeking to register for an account will need to enter a unique email, a unique username and a password. Successful user registration occurs when the email and username entered is unique and the password follows proper formatting. A confirmation email will be sent to the user and after following the instructions the user will be properly registered.

From this point, the user will be able to login, as they are validated and the information is stored inside the database.

3.2.2 Failed User Registration

Registration will fail if the email or username is non-unique, or if the password does not meet the minimum length requirements. When this occurs, a notification will appear telling the user what had failed and asking them to try the registration again.

3.2.3 Successful User Login

Users who want to log in must visit the login page and enter in their information. The application will compare the entered information to the user information stored inside the database. If the data matches correctly, the user will login and be able to access his account and features such as messaging and his watch list.

3.2.4 Failed User Login

Upon submitting a username that is not in the database or by entering an incorrect password, the user will be informed that their login has failed and they will be redirected back to the login page if they would like to try to log in again.

3.3 Listing Properties for Owners

Property owners are able to list properties under their account and provide information and pictures for their property. They are able to post a new listing as well as edit a listing that has changed or become unavailable.

3.3.1 Listing a new property

Property owners are able to add a new listing. They would be brought to a page that they would fill out with information about the listing such as price, number of rooms, amenities and so forth. After this information is filled out and deemed enough to create a listing, a new entry is created in the database under the user with this information, and it becomes a candidate to be searched for.

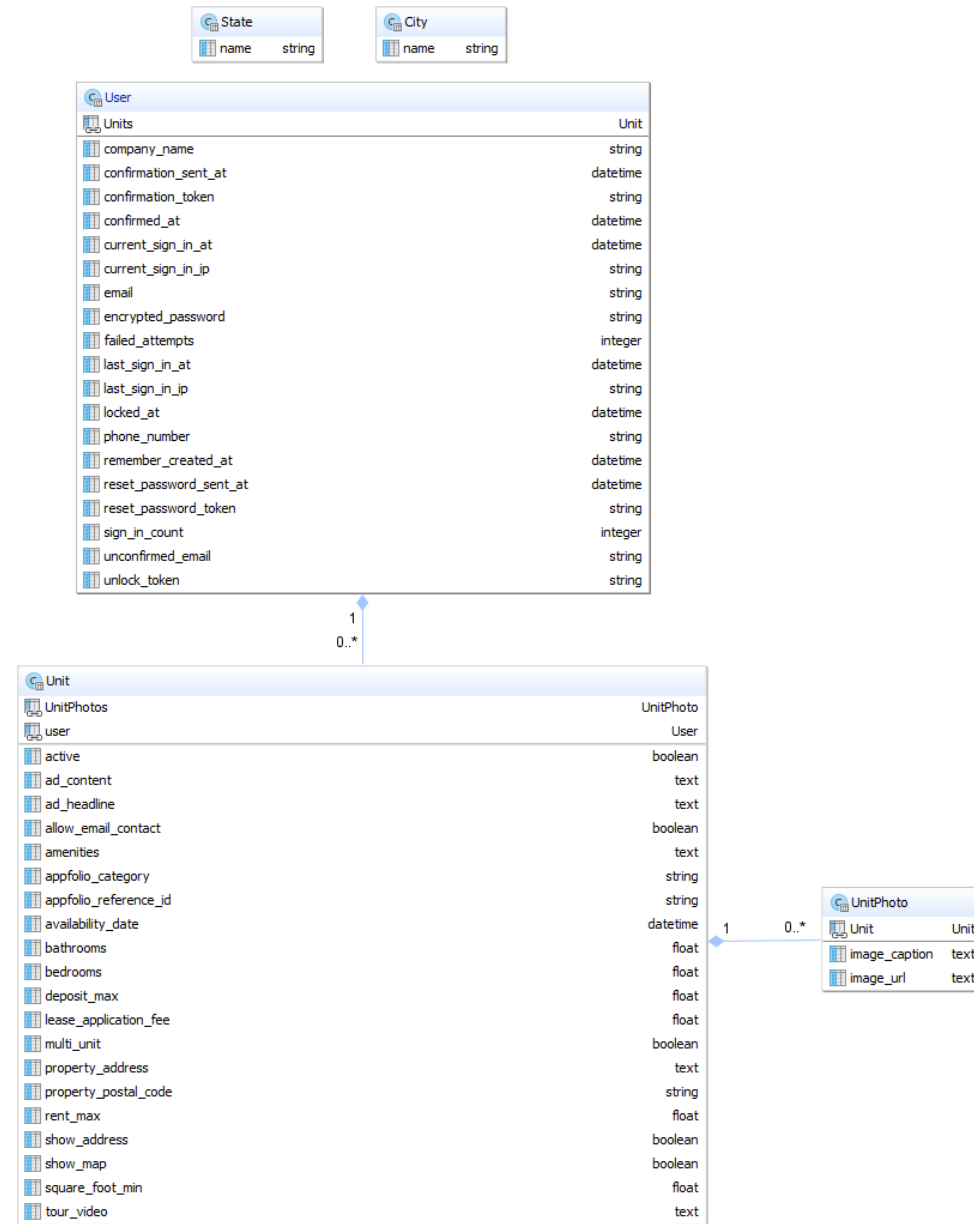
3.3.2 Editing/Deleting an existing property listing

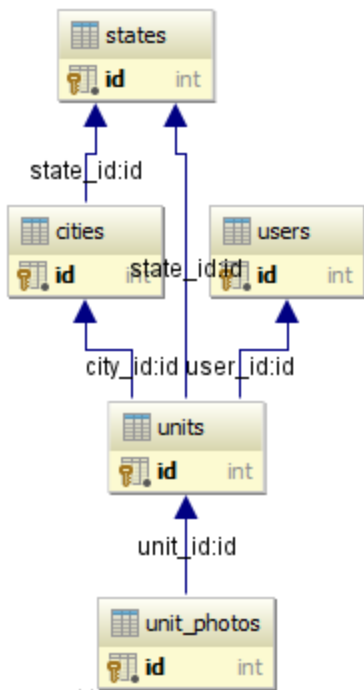
Property owners are able to edit and delete their listing as circumstances change. Property owners will be able to edit details about their listing, and these will be recorded and changed in the database. However, deleting a listing will simply change a flag on that listing in the database. This will make it so the listing will not become a candidate for the searching and filtering, but it will still remain in our records.

3.4 Filtering Properties for Renters

Renters are able to use the application to filter through houses based on certain criteria. Examples include filtering within a certain City, filtering within a certain price range. The database will be searched for houses that match the user's search options and subsequently displayed to the users.

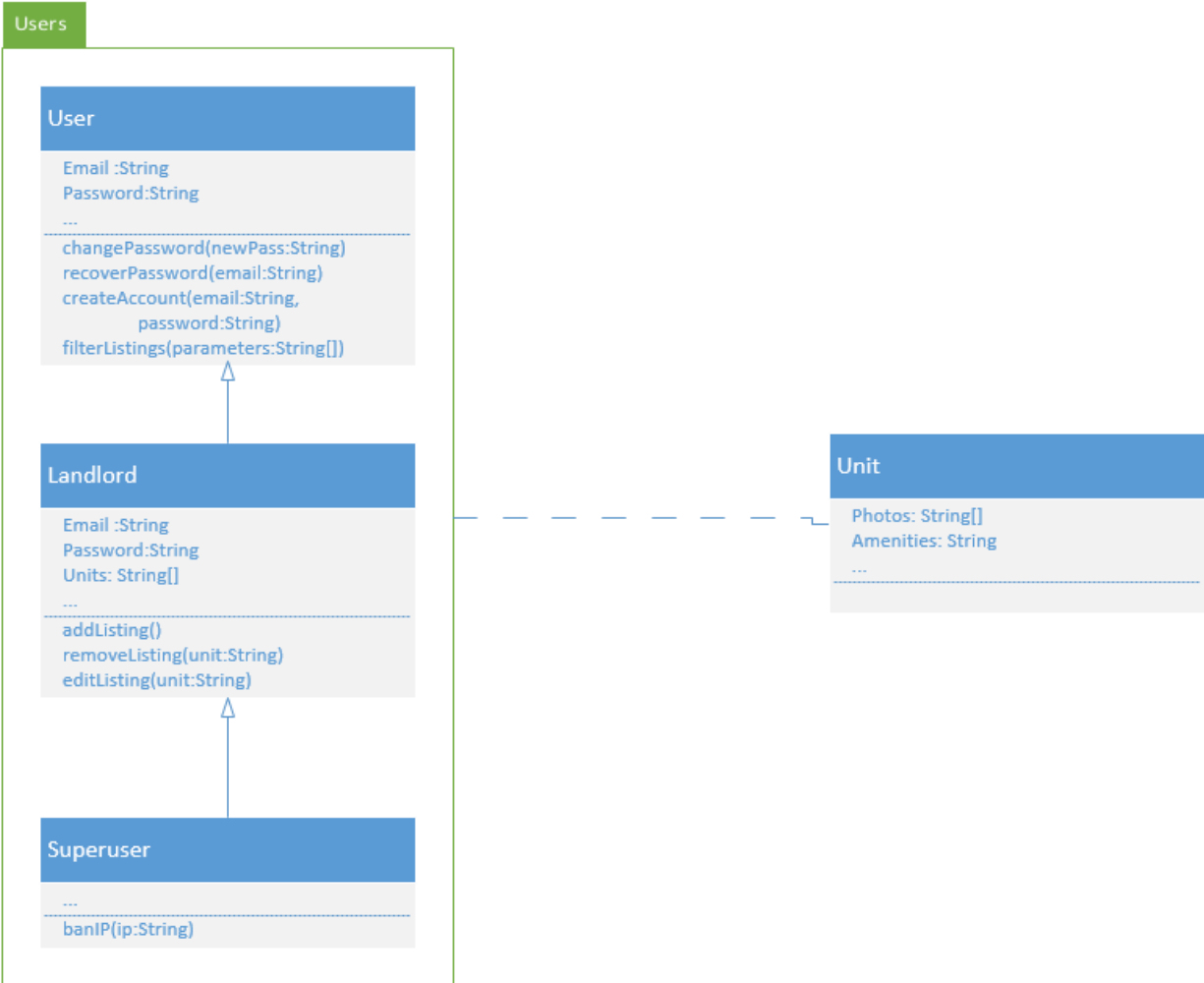
4. Class Diagram





Powered by yFiles

4.1 UML Diagram



5. UI Mockups

5.1 Log-in and Authentication

5.1.1 Log-in Page

Please sign in

 Remember Me

[Sign up](#) [Forgot your password?](#)

5.1.2 Registration Page

Please register below

[Sign in](#)

5.1.3 Password Recovery Page

Forgot your password?

Send instructions

[Sign in](#) [Sign up](#)

5.2 Listing Properties for Owners

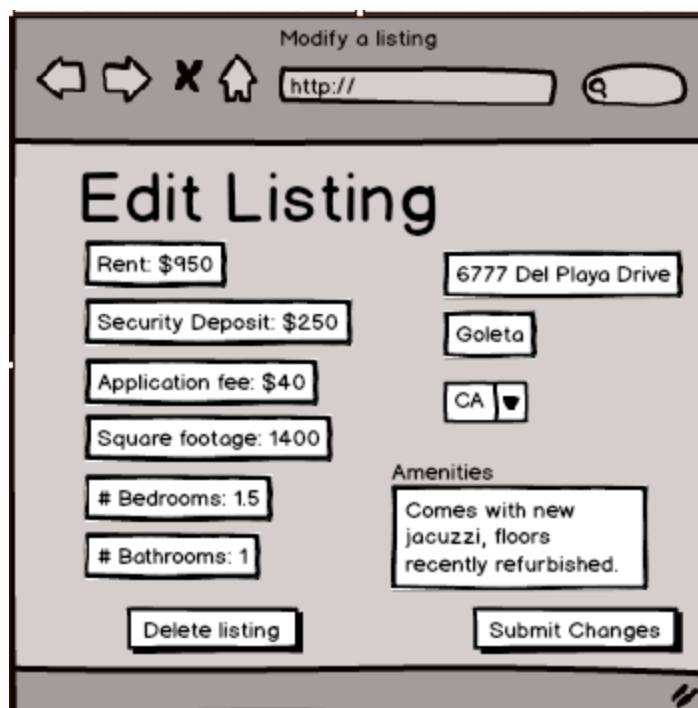
5.2.1 Creating a new listing



The screenshot shows a web browser window titled "Create a listing". The address bar contains "http://". The main heading is "Create New Listing". The form includes the following fields and controls:

- Rent
- Address
- Security Deposit
- City
- Application fee
- State (dropdown menu)
- Square footage
- Amenities (text area)
- # Bedrooms
- # Bathrooms
- Submit

5.2.2 Modifying an existing listing

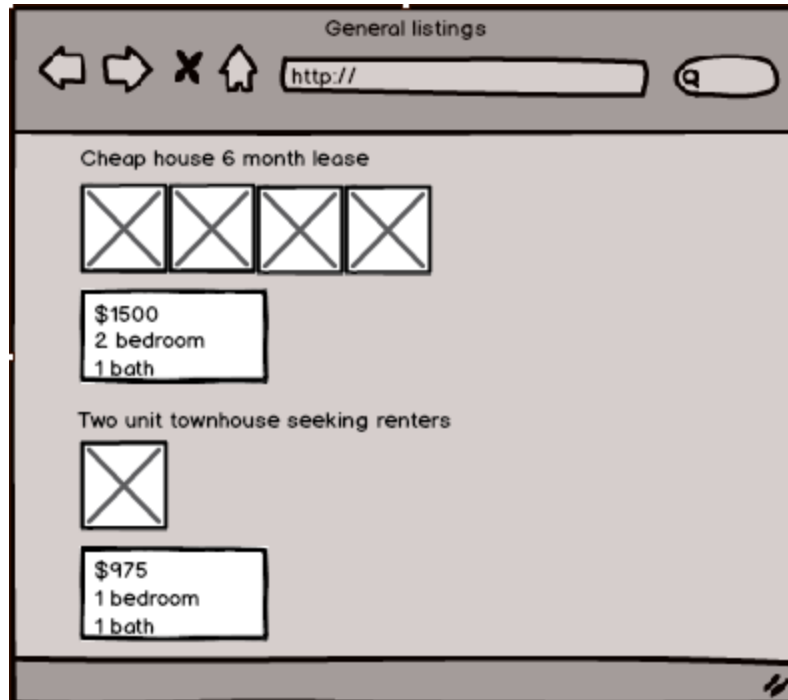


The screenshot shows a web browser window titled "Modify a listing". The address bar contains "http://". The main heading is "Edit Listing". The form displays the following information and controls:

- Rent: \$950
- Address: 6777 Del Playa Drive
- Security Deposit: \$250
- City: Goleta
- Application fee: \$40
- State: CA (dropdown menu)
- Square footage: 1400
- Amenities: Comes with new jacuzzi, floors recently refurbished.
- # Bedrooms: 1.5
- # Bathrooms: 1
- Delete listing
- Submit Changes

5.3 Searching through listings for Users

5.3.1 General Listings Page



5.3.2 Filtering Page

Search listings

http://

Filter Listings

Price Range

min max State ▾

Square footage range

min max Low Traffic

Bedrooms

min max Low Noise Levels

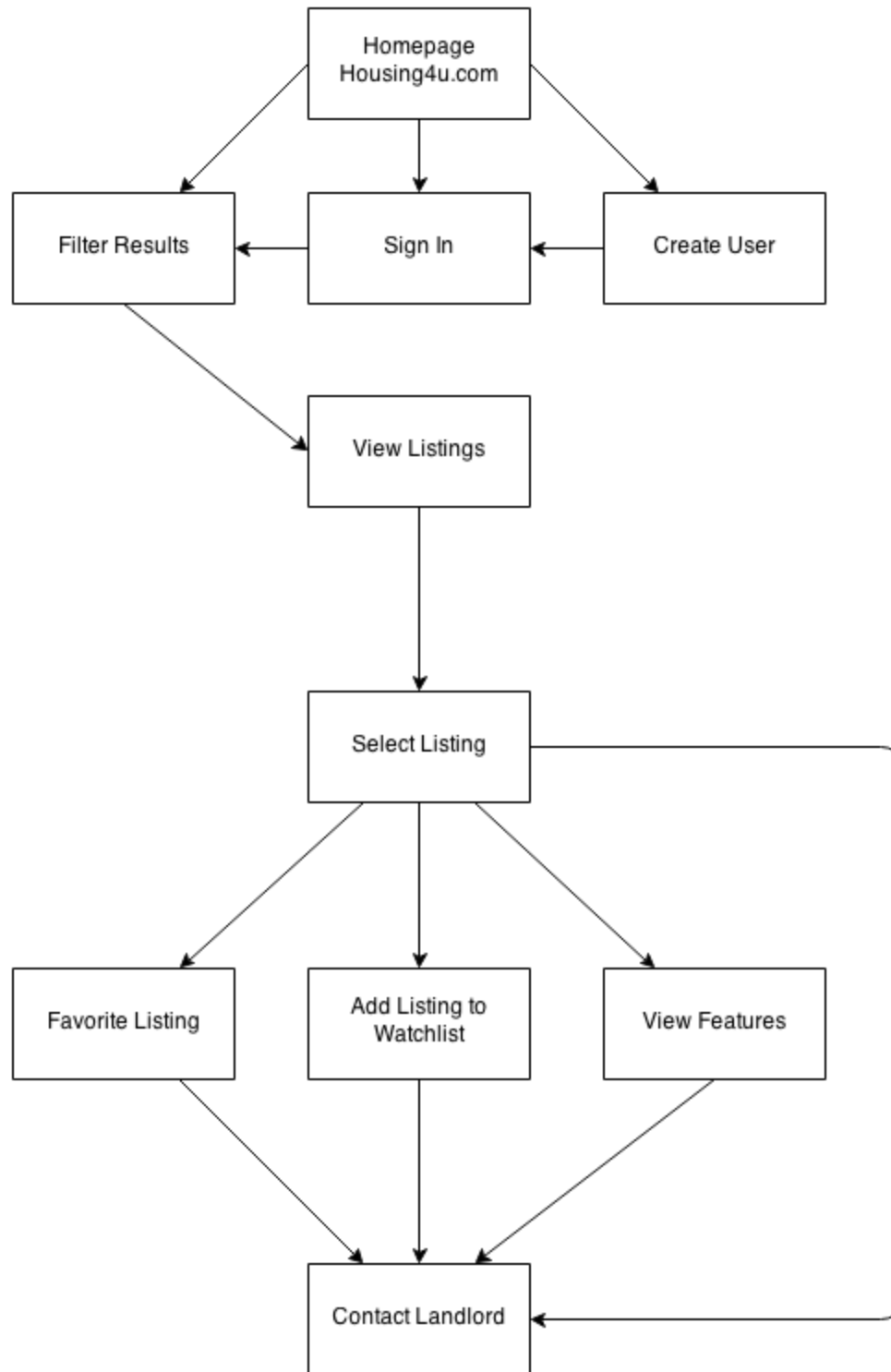
Bathrooms

min max Recent Listing(< week)

Search

6. Flow Diagrams

6.1 General Use Flow



6.2 User Authentication Flow

