CS 4750 Project Proposal and UI Design

Team members: Natalie Zhang (nyz7tc), Jennifer Liao (xl5cs), Xinyu Hou (xh4eu), Steve He(ch4gj)

Part 1: Project Proposal

Project Concept

Currently, many UVA students live on off-grounds housing around UVA, especially since UVA cannot accommodate housing for non-first years. Less than 20% of third years and fourth years live on grounds. Current leasing information is scattered by students in different Facebook and GroupMe chats. The goal of our project is a website that provides a universal database of housing in Charlottesville in a visual interface. This system allows students to search for companies and homeowners that currently have leases available, as well as details about the rent, amenities, and location for housing. The term “housing” does not include real-estate, since we will only focus on leases. The targeted users are the apartment and housing companies that typically rent to UVA undergraduate students or those who live close to grounds to provide a database for them to update their information and organize. These companies are the ones providing leasing information to the database.

Disclaimer: The user refers to the companies. The two terms may be used interchangeably throughout this document. A student user will be referred to as a customer throughout this document

Dataset information

The domain of our data will be leasing information in the Charlottesville community, specifically near grounds. The dataset will consist of companies and homeowners, rent, availability and leasing information, and amenities that come with housing. The attributes of a piece of property includes pets, water, trash, recycling, electricity, and others. Although the specific dataset that we plan on using has not been decided on yet, we plan that most of the dataset will be obtained directly from the website of companies that lease housing popular to UVA students and/or closest to UVA, such as University Apartments, as well as datasets that have information on real-estate, including Zillow. The size of our anticipated dataset will come from a range of five to ten companies. Each has multiple properties underneath them. By definition, we will refer to one property belonging to the company as one apartment unit or leasing option. We anticipate that the total amount of leasing options will be around twenty.

Functionalities

Our database will contain leasing/housing companies with their respective database information regarding leases that they currently have open, lease length, price, floor units, style (studio, 1B1B, flat, etc.), amenities offered, etc.). Utilizing this type of information, we will be able to complete the following actions on the database:

1. Retrieve data
    - ❏ Purpose and what it does:
      This will provide companies with a way to see their properties that they have submitted to the database, including price range, the location of the apartment, whether the apartment is fully furnished or not.
    - ❏ How users will use it:
      When the user visits the home page, we will retrieve all the apartment data for that company and display it onto the home page table (see UI mockups at the end of this document).
    - ❏ What users can expect from using it:
      The users can see their respective database on the home page.

2. Add data
    - ❏ Purpose and what it does:
      The company can add information about a new apartment that becomes available for renting into the database.
    - ❏ How users will use it:
      There will be an “add” option on the user interface. The user will be able to enter information about the apartment into a form and submit it. This information will be saved into the database.
    - ❏ What users can expect from using it:
      The user can expect that after adding to the database through this option, it will reflect in the home page screen.

3. Update data
    - ❏ Purpose and what it does:
      We will provide the companies a way to adjust the information about their listing apartments, for example, they may want to update the rental price or availability of their apartments.
    - ❏ How users will use it:
      Companies will have access to edit information about their apartments. They can pull up any specific tuple and make changes to it.
    - ❏ What users can expect from using it:
      Users do not need to remove and add new housing information to our database. They can simply update the currently available housing and account information.

4. Import data

- ❏ Purpose and what it does:
  Users will be able to import their property data via a JSON or CSV file (we are still deciding) to upload multiple properties for the company at once.
- ❏ How users will use it:
  There will be an “import” option on the user interface. While adding their property information to the database, instead of adding their property one at a time, users can import multiple properties via a JSON or CSV file.
- ❏ What users can expect from using it:
  This functionality allows users to add properties’ information more conveniently and quickly.

5. Delete data
   - ❏ Purpose and what it does:
     Users will be able to delete data from the database. Companies can remove housing information that is no longer available.
   - ❏ How users will use it:
     After logging in, users can remove the housing information from the database when the apartment is already rented.
   - ❏ What users can expect from using it:
     The user can expect that after deleting from the database through this option, it will reflect in the home page screen.

Requirements Fulfillment

We satisfy all project requirements:

- Relational database with realistic data

Our database will be built in MySQL, and it has relational data as a lot of the information has dependencies based on other data in the database. For example, pulling information about amenities for a piece of property based on the particular lease or leasing company a customer may be interested in. Our data will be realistic as we are going to be utilizing real data that we find using online leasing services, such as leasing company websites of popular student off-grounds apartments or Zillow (a real estate database). Zillow is also an API for gathering leasing data which will help us gain a large amount of usable data.

- Web-based application with dynamic behavior for UI

We are building three UI pages, one for user sign up or log in, another for adding properties, and the third for displaying information in the database with modification and deletion options. One of the web pages will reflect the information in our database, dynamically updating based on changes to the database from the user. See Part 2 for mock-ups of these pages.

- Use and interaction with the database

When a company signs up to put information into this UI, we will add the company to the database of users, and we will create a database for the company’s own database. Any retrieval, additions, deletions, and updates to the database will be allowed for the company.

- Includes required functionalities (retrieve, add, update, delete, import)

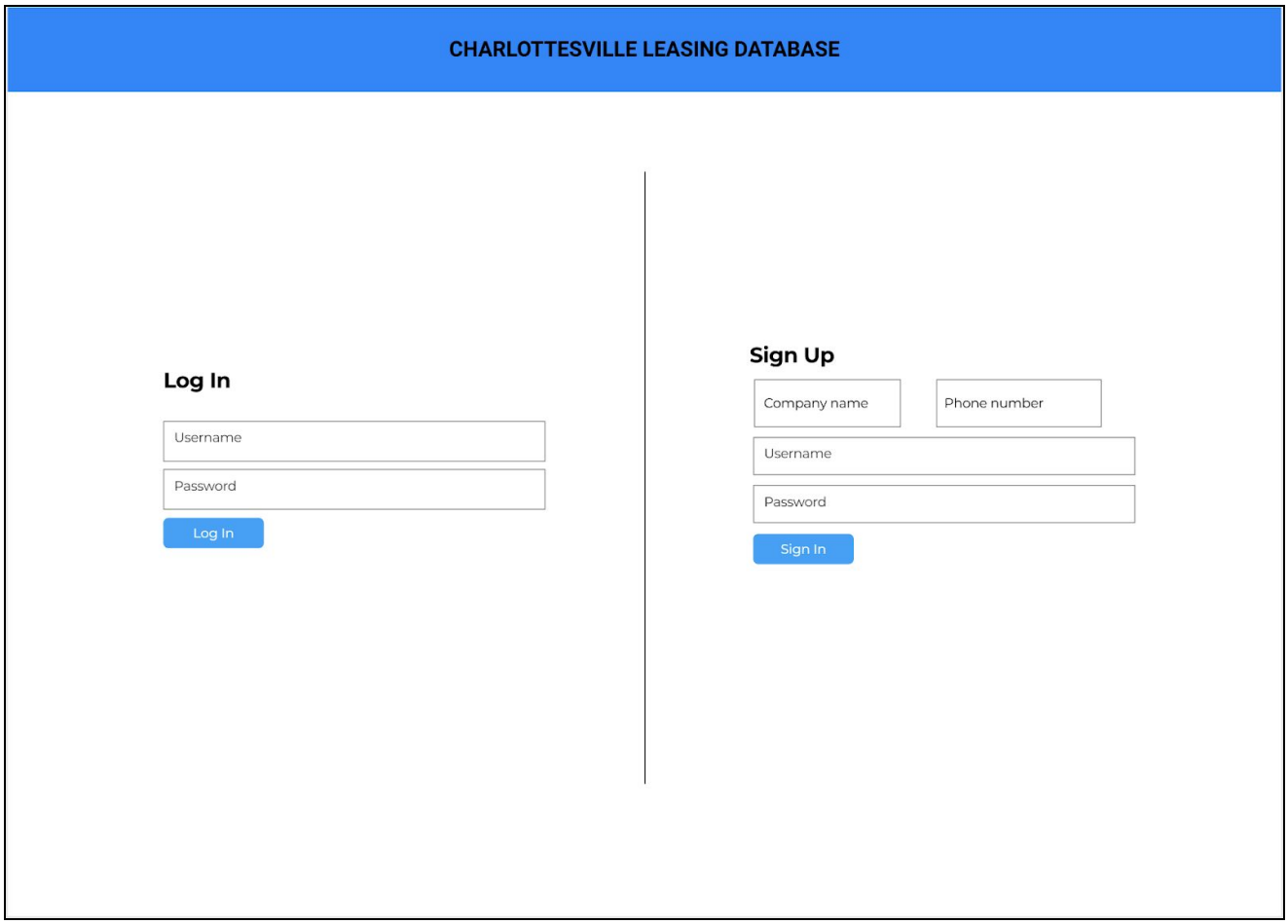
The user (company) will be able to retrieve, add, update, and delete data from the database. Please see the UI mock-ups. The user will also be able to import their property data via a JSON or CSV file to upload multiple properties for the company at once.

- Support multiple and returning users

Our database and UI will be able to support multiple users by allowing there to multiple users/companies in the database. When the user revisits the database, all their previous interactions with the database will be retained in the database.

Part 2: User interface design

Login/Sign Up Page:
The user can sign up or log in to access their data. If they are returning, they can access the database and add/delete/retrieve/change data as shown later on. Here, users can either log in if they are already in the system to support multiple return users or sign up if they are a new user. Returning users only need a username and password. New users need to add more information such as their company name and phone number in addition to the previous ones.



Home Page:
This page will display all the retrieved data for the user that is currently logged in. This is all the data that the company has added thus far. We allow for modification and deletion of entries on this page. For modification, we are anticipating allowing the user to directly modify table values.

Add Page:
This page will give the users options to either manually add their property information to the database or import their property information into the database via JSON or CSV. The import functionality will allow users to add multiple properties' information at once.