SW Engineering CSC 648/848 [2] - Fall 2016
“myplace” Web Application
Team 14

Sophia Amin - samin25@mail.sfsu.edu
Ilya Ivanenko
Daniel Nguyen
Patrick Aung
Jimmy He
Ulises Martinez

“Milestone 1”

October 6, 2016

Submission Date	History
10/06/16	First Revision
10/11/16	Second Revision

Table of Contents

I. Executive Summary

Myplace is an apartment rental web application that is built by San Francisco University (SFSU) students, for SFSU students. Myplace leverages itself over its competitors by limiting its rental services to SFSU students only. Myplace will include basic features such as search, filters, maps, apartment listing, and communication. However we highlight these features by basing the design similar to official SFSU websites; thereby creating a welcoming and native feel familiar to SFSU students as well as an extension of SFSU's online identity. By emphasizing simplicity and intuitive design, we can serve clients of all ages and focus on delivering the one desire of our SFSU students: finding a place to rent.

Guests and students shall be able to view apartment listings provided by landlords and filter results to better find exactly what they are looking for. After finding an appropriate living space, the student can contact the landlord to get more information and setup appointments.

Landlords have the ability to create apartment listings that can be seen by both guests and students. Landlords can provide information such as images, address, neighborhood, available times, and etc, which will allow students to better filter for their own apartment. In addition, landlords will have the option to adjust different visibility options for the information provided in their listings to protect their own security.

All postings shall be moderated by the sites administrators who can remove inappropriate or no longer relevant apartment listings. Administrators can also remove user accounts that are unproductive or toxic for the website's community.

Myplace's purpose is to deliver a simple and reliable apartment renting service to SFSU students. Although this type of service is not new, there has yet to be a renting service that targets SFSU students adequately.

As a small startup of SFSU students, we believe that Myplace will not just provide a better renting service, but also foster a better relationship between SFSU students, SFSU itself, and local landlords. We know the difficulties and obstacles in finding the perfect apartment and can provide competitive resources so that each user has the best overall experience.

II. Use Cases

2.1 Gary the guest - Gary is looking for apartments in San Francisco and stumbles upon this website. He is impressed at how easy it is to filter through the listings to find his ideal apartment. He clicks on the apartment only to find out that he must register on the website as a SFSU student in order to contact the landlord. Gary remembers that he has a niece named Sam who will be transferring to SFSU and decides to recommend the website to her.

2.2 Sam the student - Sam is a new transfer student at San Francisco State University. She will be attending the university to complete a bachelor's degree in Holistic Health. Since she will be attending school for two years, she wants to find an apartment where she can spend the semester then move back home with her parents during the holidays. As a student, she would like to search for apartments before every semester and find locations that are not too far away from campus. She enjoyed this site's ease of use and features including filters and secure ways of communicating with landlords.

2.3 Leah the landlord - Leah just found out on short notice that she is leaving on a seasonal business trip in a few days. She wants to rent her apartment next to SFSU temporarily to a student. Due to her limited time, she needs to post an apartment listing and find someone immediately. Thanks to Google Analytics, she found this site where she was attracted to the site's rapid process of registering and listing her place. She particularly liked how easy the website made it to describe what her apartment offers. This saved her time by giving her the option to know exactly what type of information she needs to provide.

2.4 Andy the Admin - Andy is logged in as the website admin. Andy is browsing through new apartment listings and notices that some images uploaded by the landlord contain inappropriate content. Andy then deletes the apartment and contacts the landlord to give him/her penalty. While this was happening, Andy receives a notification due to a flagged content by a user. Andy reviews the flagged content to determine whether or not it breaches the website's policy. If so, Andy will remove the apartment and contact the landlord to give him/her a penalty. If not, then Andy will simply remove the flag. Afterward, Andy will repeat these operations.

III. Data Definitions

3.1 **Guest:** An unregistered user who can browse through appartments without having to register. Has limited view of apartment information.

3.2 **Student:** A registered user that can browse through apartments and contact landlords for information. Has more access to information than guests.

3.3 **Landlord:** A registered user that can post apartment listings for rent by uploading basic information.

3.4 **Admin:** can moderate content uploaded by user and delete inappropriate images (must be logged on)

3.5 **Apartment:** A location that can be posted by landlords up for rent and viewed by guests or students.

IV. Initial List of Functional Specs

1. **Webpage** shall have:
   - **1.1** Use of filters (e.g. distance, price.. etc.)
   - **1.2** Registration
   - **1.3** Login
   - **1.4** Preview/List of apartments
   - **1.5** Featured apartments (Updated Daily)
   - **1.6** Use APIs suggested in High-level architecture
2. **Guest** shall be able to:
   - **2.1** Register for an account (only as SFSU student or Landlord)
   - **2.2** Have limited view of apartments
   - **2.3** Filter apartments results
3. **Student** shall be able to:
   - **3.1** Login account with mail.sfsu.edu email
   - **3.2** Search apartment with or without filter
   - **3.3** Save favorite apartments
   - **3.4** Communicate with landlords (by email)
4. **Landlord** shall be able to:
   - **4.1** Offer multiple apartments for rent
   - **4.2** Edit their apartment (ie. description, hide, delete, etc.)
   - **4.3** Set visibility level of apartment attributes
   - **4.4** Communicate with interested students (by email)
   - **4.5** Upload images of apartment
   - **4.6** Option to bump listing to the top of the list of apartments (after 7 days)
5. **Admin** shall be able to:
   - **5.1** Delete user accounts and/or apartments but not edit them
   - **5.2** Send messages to students/landlords (ie. warnings)
6. **Apartment** shall have **data** including:
   - **6.1** Posted date
   - **6.2** Neighborhood (optional address)
   - **6.3** Tags
   - **6.4** Description
   - **6.5** Price range
   - **6.6** Rental Term (Availability)
   - **6.7** Parking availability
   - **6.8** Distance from SFSU calculated using Google APIs
   - **6.9** Optional images (max 10)
   - **6.10** Map + Bus Routes to SFSU using Google APIs
   - **6.11** Rating System (for apartments frequently rented)
   - **6.12** Thumbnail for preview (if no image upload, default is map)

V. List of Non-functional Specs

1. Application shall be developed using class provided LAMP stack
2. Application shall be developed using pre-approved set of SW development and collaborative tools provided in the class. Any other tools or frameworks shall be explicitly approved by Marc Sosnick on a case by case basis.
3. Application shall be hosted and deployed on Amazon Web Services as specified in the class
4. Application shall be optimized for standard desktop/laptop browsers, and shall render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome. It shall degrade nicely for different sized windows using class approved programming technology and frameworks so it can be adequately rendered on mobile devices
5. Data shall be stored in the MySQL database on the class server in the team's account
6. Application shall be served from the team's account
7. No more than 50 concurrent users shall be accessing the application at any time
8. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
9. The language used shall be English.
10. Application shall be very easy to use and intuitive. No prior training shall be required to use the website.
11. Google analytics shall be added for major site functions.
12. Messaging between users shall be done only by class approved methods to avoid issues of security with e-mail services.
13. Pay functionality (how to pay for goods and services) shall be simulated with proper UI, no backend.
14. Site security: basic best practices shall be applied (as covered in the class)
15. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development, and only the tools and practices approved by instructors
16. The website shall prominently display the following text on all pages "SFSU/FAU/Fulda Software Engineering Project, Fall 2016. For Demonstration Only". (Important so as to not confuse this with a real application).

VI. Competitive Analysis

Features	places4students	sfstate.renthello	Our Future Product
Map	+	+	+++
Search	+	+	-
Member Login	+	+	+
Ease of Use	+	+	+++
Security	+	+	+
Filters	+	+	+++

+ Feature exists; +++ superior; - does not exist

myPlace's closest competitor is sfstate.renthello.com (RentHello). RentHello is another apartment renting web application that caters to SFSU students. However, anyone may register to RentHello without verification. Myplace will verify each renter as an SFSU student. Also, RentHello follows the design of the official SFSU website, but are convoluted due to its dynamic content and excessive styling. Myplace will display only what is needed to give SFSU students a familiar feel to the official SFSU main website and serve their needs.

VII. High-Level System Architecture

The following tools, APIs, and frameworks will be used to assist the development of this web application.

1. **LAMP Stack**

   The LAMP stack is a very popular web platform that allows for easy setup and maintenance of web applications. It's acronym stands for Linux, Apache, MySQL, and PHP which are used together to provide all the tools needed for developing and hosting web applications.

   - **1.1** **Linux** is an open source operating system on which the rest of the LAMP stack runs.
   - **1.2** **Apache** is the component of the LAMP stack that acts as a web server. It processes all incoming/outgoing HTTP requests/responses and, like Linux, is open source as well.
   - **1.3** **MySQL** handles communication with the database for the server. It allows for multiple users to access, create, and modify database entries and uses SQL for these communications.
   - **1.4** **PHP** is the programming language we will use to develop our webpages. PHP can be embedded easily into HTML which makes it easy to use for web development.

2. **APIs and Frameworks**

   APIs and frameworks go hand in hand when developing web applications. They provide a set of tools and libraries to utilize for the creation of web applications.

   - **2.1** **Mini** is a simple open source framework made specifically for PHP. It follows the traditional MVC (Model - View - Controller) architectural pattern.
   - **2.2** **Google Analytics API** will allows us collect, configure, and report on user interaction.
   - **2.3** **Google Maps API** contains methods to display the apartments on a map so users can visually see where apartments are located. In addition, "Directions Service" will be enabled with the mode set to TRANSIT to offer possible public transportation routes to SFSU.
   - **2.4** **jQuery** is an open source JavaScript library designed for the client-side scripting of HTML. This will allow for the creation of dynamic web pages.
   - **2.5** **Bootstrap** is a front-end framework for designing websites and web applications. It combines HTML, CSS, and JavaScript into this one framework for ease of use.
   - **2.6** **Supported Browsers** that our website will support are the latest two versions of Safari, Chrome, and Mozilla.

VIII. Team

Member	Role
Sophia Amin - samin25@mail.sfsu.edu	Team Lead, Front-End
Ilya Ivanenko	Team Tech Lead, Controller
Daniel Nguyen	Back-End Developer
Patrick Aung	Back-End Developer / Database
Jimmy He	Back-End Developer
Ulises Martinez	Front-End Developer