

SW Engineering CSC 648/848 Spring 2019
"Stay"

Team 02

Jaren Lynch
Chris Rosana
Monali Mirel Chuatico
Thanh Le
Shan Kwan Cho
Russelle Pineda
KaHo Lee

Milestone 1

History Table

Progress	Date
First Draft	March 4th, 2019
Revision 1	March 6th, 2019

Table of Contents

Table of Contents

I. Executive Summary

II. Use Cases

III. Data Definitions/Summary

IV. List of functional specs

V. List of non-functional specs

VI. Competitive analysis

VII. High-level system architecture

VIII. Team Roles

VIII. Checklist

I. Executive Summary

There are many companies that offer online marketplaces connecting renters and rentees. However, there are none specifically tailored to serve housing needs of university students - until now. Stay is a an application for San Francisco State students that serves to significantly decrease the stress and difficulty of finding affordable housing near campus. Stay makes it easier for SFSU students to find housing tailored to their needs. The user interface is visually appealing and simple to use, so people of all ages and skill levels can use it with ease. Stay aims to make every students life easier by connecting them to housing specifically tailored to their needs as students, efficiently and affordably.

II. Personae and main Use Cases



- A. About Sam (Tenant):
- a. Sam is a freshman SF State Student
 - b. He is proficient in using WWW
 - c. He is originally from southern California.
 - d. He doesn't have a car
 - e. He is new to the area and does not know anyone
 - f. He wants to look for a place to stay that is accessible and is walking distance to SFSU
 - g. He is looking to connect with other roommates who are willing to split the rent



B. About Darnell (LandLord):

- a. He is an old grandpa
- b. He is less experienced with using the WWW
- c. He doesn't like loud noises
- d. He lives in a 3 bedroom home
- e. He wants to rent out 2 bedrooms in order to make profit because he is retired and is no longer able to work



C. About John (Administrator)

- a. John is a Software Engineer
- b. He is an SFSU Alumni
- c. He is from the Bay Area and he understands that it's not easy to find an affordable place to stay
- d. He wishes to solve the students' problem of finding a place to "stay"
- e. He manages "Stay" and keeps the application running for the students in the Bay Area.

III. List of main term and definition

1. Unregistered user: The user who comes to our site and looks for a room to rent.
2. Registered users: This user is a member of “Stay” and shall receive more benefits than unregistered users such as quickly receiving information they has been looking for or easily finding a home or for rent out. To be a member of “Stay”, this person must provide a valid contact information such as phone, email, and/or optional profile photo. These requirements not only to ensure that the user isn’t scam but also make them easily connect to the website.
3. Administrator: In general role, this person mainly interacts with users. This person has the most authority to track all users’ activities. In order words, this person is a supervisor of all customer transactions.
4. Developers: A person who analyzes and develops the website. This person shall continue working on the site to maintain and debug the page when needed
5. Listing: different and connected items that users can view to manage their searches
6. Login: identifier user’s information on Stay’s system. It requires User ID and password.
7. Sign-up: authorize a user to be a member on this site.
8. Search: the results returned from querying for listings.
9. Saved listing: listings that users want to check back later on
10. Tenant: a registered user who looks for a place to rent.
11. Landlord: a registered user who looks for a person needing a place to rent out.
12. Transaction: The booking of a rental property.

IV. List of functional requirements

1. Stay
 - a. Briefly introduce and explain what this site is about
 - b. Ability to make transaction corrections, cancellations, and adjustments
 - c. Shall collect and store all users' activities to databases
 - d. Shall have internal and external interfaces
2. Unregistered
 - a. Unregistered users shall be able to login or register.
 - b. Unregistered users shall be able to search information they need
 - c. Unregistered users shall be able to search by location, price and type.
 - d. Unregistered users shall be able to sort listings.
 - e. Unregistered users shall be able to search and browse listings.
3. Registered user
 - a. Registered user shall be able to do anything unregistered user can do.
 - b. Registered users shall be able to upload a listing on the site
 - c. Registered users shall be able to communicate directly to other registered users about a listing.
 - d. Registered users shall be able to live chat with administrators
 - e. Registered users shall receive information they have been looking for
 - f. Registered users shall be able to view their own historical activities
 - g. Registered users shall be able to favorite a listing and save it for later.
 - h. Registered user shall be shown listings based on their location on the main page upon logging in.
4. Navigation drawer:
 - a. Shall allow users to navigate to different sections of the application.
5. Administrators:
 - a. Shall be able to set up rules for users interacting with the application.

- b. Shall be able to ban registered users
 - c. Shall be able to view historical data such as searches, making payment, listings, etc. from all registered users
 - d. Shall be able to check and approve or remove registered users' posts.
 - e. Shall be able to track each transaction.
- 6. Developers
 - a. Developers shall be able to edit the site
 - b. Developers shall be able to view databases and maintain the server
 - c. Developers shall be able to view internal and external interfaces

V. List of non-functional requirements

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO).
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
3. Selected application functions must render well on mobile devices
4. Data shall be stored in the team's chosen database technology on the team's deployment server.
5. No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
7. The language used shall be English.
8. Application shall be very easy to use and intuitive.
9. Google analytics shall be added
10. No e-mail clients shall be allowed
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.
12. Site security: basic best practices shall be applied (as covered in the class)
13. Before posted live, all content (e.g. apartment listings and images) must be approved by site administrator
14. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development

15. The website shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2019. For Demonstration Only" at the top of the WWW page. (Important so as to not confuse this with a real application).

IV. Competitive Analysis

Tentative Features	Airbnb	Roommate	Trulia	"Stay"
Search bar (location)	++	+	++	++
Sort (Price, Location)	+	-	+	+
Rating	+	+	+	++
Room information page	++	+	+	+
Reviews	+	-	+	+
Unique User Interface	+	+	+	++
Search by SFSU Housing Program	-	-	-	+

Every house-renting market has different approach to customers. The general idea of technology we will use is covering most of the standard technologies that exist in the industry and our product services is quite similar to the global companies on the market such as Airbnb, Roommate and Tulia, etc. Whatever approach, the key part of our platform is that especially as fast communication as possible with SFSU students who looking for a room to stay and the landlord of the room.

VII. Architecture and Technologies Used

1. Supported Browsers: Latest versions of Safari and Chrome
2. Framework: React, Redux
3. Software Tools: Atom, Terminal, Visual Studio Code
4. Web Server: Google Cloud Platform
 - a. Server Tools: Docker, Node.js

5. Database: MySQL

VIII. Team

Team Members	Roles
Jaren Lynch	Team Lead/Github Master
Monali Mirel Chuatico	Frontend
Thanh Le	Frontend
Chris Rosana	Frontend
Shan Kwan Cho	Backend
Russelle Pineda	Backend
KaHo Lee	Backend

VIII. Checklist

1. Team found a time slot to meet outside of the class: **DONE**
2. Github master chosen: **DONE**
3. Team decided and agreed together on using the listed SW tools and deployment server: **DONE**
4. Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing: **DONE**
5. Team lead ensured that all team members read the final M1 and agree/understand it before submission: **ON TRACK**
6. Github organized as discussed in class (e.g. master branch, development branch): **DONE**