

SW Engineering CSC648/848 Spring 2019

SFSURent

Team 11

Cory Lewis(Team Lead, clewis9@mail.sfsu.edu)

Xinyu Zou(GitHub Master)

Soheil Ansari(Back End Lead)

Junwei Liang

Poorva Rathi

David Dropping(Front End Lead)

Chintan Sanjay Puri

Milestone 1

3/1/19

History Table:

1. Executive Summary

Are you a student looking for affordable housing near SFSU? Do you own property in the bay area looking for quick renters? This website is the application for you! As a student, the last thing you want to worry about is finding a place to live. Especially in a city where two thirds of the housing market is renter owned. It becomes a struggle to dig yourself out of the avalanche of adverts and postings promising the "perfect place to live," let alone finding the one that's actually right for you. And the same can be said for the landlord looking to rent their house, when their ad becomes but a snowflake on top of the avalanche. SFSURent is an alternative solution geared for San Francisco State University students to find the apartment or house that's perfect for them.

SFSURent caters to SFSU students looking for housing that is inexpensive and close to school. There are many websites that offer housing options but they are too expensive, far from school, and don't cater to students specifically. Craigslist, Roomster, and Zillow are all common sources for housing but offer minimal opportunity for finding the right place for a typical SFSU student. Our platform does not charge to contact landlords, and we have an administration system in place that removes postings that might contain inappropriate content or does not meet our website guidelines.

SFSURent is a startup founded by SFSU students to better our community and offer a more streamlined source of housing opportunity.

This platform is new and is constantly being optimized and maintained by our team to fit the needs of the students!

2. Personae and main Use Cases

2.1 Key Personas

2.1.1 Tenant:

Most of the tenants are SFSU students. As college students, we don't have too much income. Many students have to work and study during the semester. After students receive their paychecks, they spend most of the money on housing, but at the same time, they also need to purchase textbooks and food. They don't have much money left in their pocket after they pay the house. Some of the students decide to live far away from school because they can't find cheaper house near school. They have to wake up earlier every morning and try to catch the bus, and sometimes they are still late for classes because of the traffic. The main goal for students is to find inexpensive houses that are close to SFSU. If students live close to school, they don't have to wake up at 7 a.m. in the morning, and they have more time to rest. Especially when the exams come, we don't want to be late because of the traffic. Students can manage their time better and save money if they use our application. Most of the college students are comfortable to use web application because sometimes they have to learn how to use web applications in some of their classes.

2.1.2 Landlord:

As landlords, they want to rent their houses quick and safe. Before the current tenant leave, they need to find a new tenant to take the position. If they can't find any new tenants, they earn less money every month. Landlords also want to know that who are living in their houses. They want to make sure that tenants don't mess up with their houses. And they also want to make sure that the tenants have clear background(not sure about this word), for example, are they doing drugs, or are they crazy party people. Tenants' schedules are also important for other renters. Gender is also need to considered, some of the tenants prefer living with all girls or all boys.

2.1.3 Admin:

As an administrator, the key responsibilities are to manage, modify, insert, delete data from the database so that users can have access to safe and useful information that is relevant to them. An administrator also helps in troubleshooting issues that users on the platform such as landlords and tenants face. He/she performs managerial tasks that help maintain the usefulness of the information on the platform and also prevents any malpractice or misbehaviour.

2.2 Use Cases

2.2.1 Commuting student

John is a SFSU student who lives in Hayward. He has to drive to school every day because he has classes Monday through Friday. Unfortunately, most of his classes begin at 9 a.m. in the morning. He has to wake up at 7 a.m. because the traffic is terrible in the morning. Sometimes he wake up early enough, but he is still late for class. He is afraid of be late on the exam date, so he wants to find a house nearby school. After he browses some websites, he still can't find the appropriate houses. Some houses are close to school, but they are too expensive for John. Some houses are cheaper, but they are far away from school. John is very frustrated, and he tries to find a solution.



2.2.2 Cheaper Living

Mary is a SFSU student who lives nearby campus. She is very busy every day because she has to work and study. She spends most of her money on housing; therefore, burger, pizza, and sandwich are her main food every day. She doesn't have enough time to cook or enough money to buy healthy food. Most importantly, she doesn't have enough time to study. Mary wants to find a cheaper house, so she doesn't have to work every day, If she can move to a cheaper house, she will have more time to enjoy her life and focus on her education. She tries to find a new house nearby school, but she can't find any appropriate houses on the Internet.



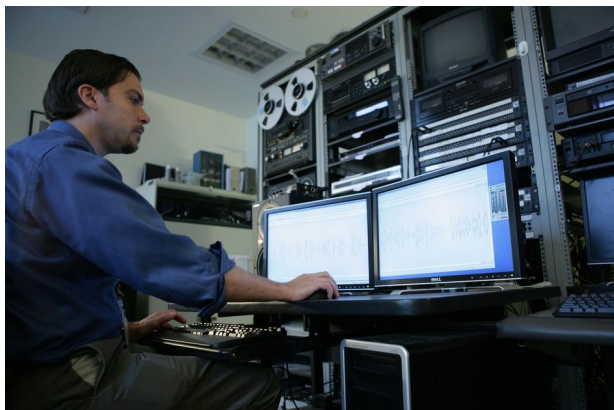
2.2.3 Renting out properties

Adam is a retired employee with multiple properties around the SFSU campus. He wants to post these properties up for rent but is unable to find tenants. He suffers from age related diseases so he can not spend a lot of time trying to find people who would want to rent his properties. He uses this website to conveniently search for and find students who are in need of an apartment that is close to the university campus.



2.2.4 Managing the information and preventing restricted content

Shane is an administrator who has access to the database and has rights to manage that data. A potential landlord is having an issue trying to sign up as he/she is not proficient in his/her computing abilities. Shane helps this potential landlord and walks them through the sign up process. A user on the website comes across restricted content and contacts Shane. Shane immediately follows up and takes down this unsafe content. Shane helps another user that has forgotten their password to reset their password so that they can regain their access to their account.



3. List of main data items and entities

- 3.1. *Unregistered User* (can browse the website but won't be able to contact the Landlord and post listings)
- 3.2. *Registered User* (can perform all actions of an Unregistered User and contact the Landlord and post listings)
 - 3.2.1. *Admin* (can access, modify, edit and delete all the necessary data; can block the fraud user; login required)
 - 3.2.2. *Landlord* (can post the images and details of the house; login/register required)
 - 3.2.3. *Tenant* (can view the public images but need to register to see private images and to communicate with landlord)
- 3.3. *Landlord Posted History* (contains basic details like rent and address posted by landlord in their account)
- 3.4. *Landlord Posted History Detail* (contains detail history with images, rent, address, furnishing etc. posted by landlord in their account)
- 3.5. *Tenant Search History* (history of the search houses by the tenant to simplify the user experience)
- 3.6. *Favourite item* (tenant's list added to the cart which helps in shortlisting the house)
- 3.7. *Approved item* (all the house details which are approved by Admin)
- 3.8. *Disapproved item* (all the house details which are disapproved by Admin)
- 3.9. *Blocked user* (details of user's blocked by the admin)
- 3.10. *House Listing* (address and rent details of the house)
- 3.11. *Subject/Title* (main part of the renting information)
- 3.12. *House Images* (Images of the house posted by Landlord and approved by Admin)
- 3.13. *Type of Commuting Mode* (type like walking/biking/car/bus)
- 3.14. *Distance from University* (each commuting mode distance and time taken details from house to university)
- 3.15. *Type of House* (type like 2BHK/3BHK, apartment/ bungalow, floor)
- 3.16. *House Features* (features like furnished/not furnished, has pool, has air conditioner, pet allowed, carpeted or not, number of bathrooms, other roommates information and many more)
- 3.17. *Renting Information* (leasing details between landlord and tenant; and their feedback)

4. Initial list of functional requirements

4.1. Unregistered User

- 4.1.1. Shall browse postings by simply visiting the website and being presented with a list of postings sorted from recent to oldest.
- 4.1.2. Shall sort/filter listing by location, distance to SFSU, cost, utilities included.
- 4.1.3. Shall read the description of postings,
- 4.1.4. Shall check commute time of posting to SFSU via Google Maps (walk time, bike time, bus time, bart time, driving time)
- 4.1.5. Shall view images of postings by clicking through a carousel of images that give the user an idea of the living circumstances
- 4.1.6. Shall search postings by typing into the navigation bar at the top of the website
- 4.1.7. Shall register for an account by completing a form that will prompt general login credentials for more accurate post listings.

4.2. Registered User (Can perform all tasks of Unregistered User)

- 4.2.1. Shall contact landlord through the platform.
- 4.2.2. Shall favorite postings so they can easily be found in the future.
- 4.2.3. Shall logout of session by clicking logout.
- 4.2.4. Shall be able to edit their profile.
- 4.2.5. Shall change password for login by viewing profile and clicking "change password".
- 4.2.6. Shall use forgot password to recover login.
- 4.2.7. Shall share website with friends via Social Media through referral links.
- 4.2.8. If an user A shares the website through their referral link and a new user B registers through their referral, user A shall be given priority status which enables them to view the listings 24 hours earlier.

4.3. Landlord (Shall perform all tasks of Registered User)

- 4.3.1. Shall create listings.
- 4.3.2. Shall view postings.
- 4.3.3. Shall delete own postings.
- 4.3.4. Shall edit own postings.
- 4.3.5. Shall read through registered users' profiles.

4.4. Administrator (Shall perform all tasks of Landlord)

- 4.4.1. Shall delete any posting.
- 4.4.2. Shall block accounts.
- 4.4.3. Shall approve postings of Landlords
- 4.4.4. Shall disapprove postings of Landlords

5. List of non-functional requirements

- 5.1. 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).
- 5.2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
- 5.3. Selected application functions must render well on mobile devices
- 5.4. Data shall be stored in the team's chosen database technology on the team's deployment server.
- 5.5. No more than 50 concurrent users shall be accessing the application at any time
- 5.6. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
- 5.7. The language used shall be English.
- 5.8. Application shall be very easy to use and intuitive.
- 5.9. Google analytics shall be added
- 5.10. No email clients shall be allowed
- 5.11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.
- 5.12. Site security: basic best practices shall be applied (as covered in the class)
- 5.13. Before posted live, all content (e.g. apartment listings and images) must be approved by site administrator
- 5.14. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
- 5.15. 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).

6. Competitive analysis

	Zillow	FB Market	Craigslist	Hotpods	campus cribz	SFSU RENT
Search	B	D	B	A	A	A
Map View	B	D	B	A	A	F
Ease of use (UI)	B	A	C	A	A	B
Ease of contacting landlords	D	A	B	B	A	B
Ease of listing	D	A	C	B	A	B
Tailored for students	F	F	F	F	A	A
Marketed to SFSU students	F	F	F	F	B	A+

Ratings: A=Outstanding, B=Good, C=Acceptable, D=Poor, F=Not a feature

To assess the current available student housing websites in the market, five different housing websites were studied. Majority of the competitive systems do not offer any tailored features for the students. CampusCribz is one example that is specifically targeting the students. For example, it shows walking, biking and driving distances to the user's selected campus. Some of the competitive systems offer easy navigation and filtering tools for the search results which improve the user experience. SFSURent first priority is to create great user experience for the SFSU students. Moreover, it will combine most crucial features in the other competitive systems such as filtering, search and ease of listing for the landlords.

7. High level system architecture and technologies used

- 7.1. **Server Host:** Amazon Web Services, 1 vCPU, 1gb RAM, 15gb SSD
- 7.2. **Operating System:** Ubuntu Server (v18.04.1 LTS)
- 7.3. **Database:** MySQL (v8.0.13)
- 7.4. **Web Server:** AWS EC2
- 7.5. **Server-Side Language:** JavaScript (v1.8.5)
- 7.6. **Web Framework:** Node.js (v10.15.1 LTS)
- 7.7. **Additional Technologies:**
 - 7.7.1. **IDE's:** Visual Studio 2017, Visual Studio Code (v1.31.1)
 - 7.7.2. Bootstrap (v4.3.1)
 - 7.7.3. jQuery (v3.3.1)
 - 7.7.4. Popper.js (v1.14.7)
 - 7.7.5. Express.js (v4.16.4)
 - 7.7.6. Lint (v5.14.1)

8. Team Members

Team Lead	Cory Lewis
Front End Lead	David
Back End Lead	Soheil Ansari
GitHub Master	Xinyu Zou
Front End	Chintan
Back End	Poorva Rathi
Back End	Junwei Liang

9. Checklist

- 9.1. Team found a time slot to meet outside of the class - **Done**
- 9.2. Github master chosen - **Done**
- 9.3. Team decided and agreed together on using the listed SW tools and deployment server - **Done**
- 9.4. 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 - **On Track**
- 9.5. Team lead ensured that all team members read the final M1 and agree/understand it before submission - **Done**
- 9.6. Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.) - **Done**