SW Engineering CSC648/848 Section 01 Fall 2016

Department of Computer Science

SAN FRANCISCO STATE UNIVERSITY

Milestone 1

Initial Proposal of David's List

Team 1

Nora Schinkel, Team Lead | ncschinkel@gmail.com

David Chau, Team Chief Technical Officer

Paul Derugin

Imran Irfan

Yansen Liu

Steven Nguyen

Submitted to—

Dr. Dragutin Petkovic**,** Marc Sosnick

Department of Computer Science, San Francisco State University

Document Version: 1.1

Last Modified: 2016 October 12

Contents

1. Executive Summary

Since its inception, the World Wide Web has been used with one major focus: exchange. This has evolved from the exchange of information to the exchange of goods, and in recent years, the exchange of real estate. With websites like eBay and craigslist in the top 40 most visited websites worldwide[1], e-commerce can be a fruitful market for a wide range of users. If we focus on just one subset of users, college students, we find a group who is a frequent user of the web, but who does not generally have much money to spend. Therefore, if you wish for them to use a service, it must be worth their time and money. There are already a few rental websites made with students in mind, but none of them offer the ability for students to verify themselves. This is where David's List comes in. David's List will be a place for students to *easily and securely* rent an apartment off-campus, something a majority of students need. This *ease and security* will also attract local homeowners who want to rent out their apartments to students for the following reasons:

1. Students can create an account with an official .edu email address and become *verified students*.
2. The contact information of the homeowner will be kept *private*. An email notification will be sent when a student is interested in renting.
3. Students can add bios and pictures of themselves to create a better *owner-renter match*.

The advantages for students using David's List over other rental websites include the following:

1. Students can verify themselves, which will attract more homeowners and gives them a *wider range of places* to pick from.
2. Apartment listings will be shown on a map to give students an idea of their surroundings. Each listing will be displayed with its distance to the university campus.
3. Students can browse posts based on *unique categories* such as "distance to campus" and "whole apartment / single room".
4. Owners can add bios and pictures of themselves to create a better *owner-renter match*.

To better optimize this website, it is best to test it first on a small group of students. The preference goes to San Francisco State University (SFSU) students. This university, like our company, is situated in the San Francisco Bay Area and has very limited on-campus housing options, which makes for a large demographic for David's List.

2. Use Cases

1. Renter, Unregistered User

Edward is currently attending SFSU and wants to find an apartment to rent. He is able to browse **apartment listings** on David's List without registering for an account. He has access to the same information for each **apartment listing** as a **registered user** except for the contact information of the **owner**. Edward finds an apartment he is interested in renting. When he tries to contact the owner of the apartment, he is prompted to register, and must do so before being able to proceed.

2. Renter, Registered Student

Allen is an incoming freshman to SFSU and needs to find a place to live for his first semester. He is able to register and view **apartment listings** near the SFSU campus based on a variety of search filters such as price, rooms, and distance from campus. Once he finds an apartment that he is interested in, he is able to communicate with apartment **owner** to arrange a time to see the apartment as well as to come to an agreement on the terms of the lease. Allen now has a comfortable room for his first year of study at SFSU.

3. Owner, Registered Student

Brenda is starting her senior year at SFSU and no longer needs her apartment because she has found other living arrangements. Brenda has registered an account on our website with her SFSU email and has a "**Verified Student**" status on her profile. She is able to list her apartment on the website by providing the address of the apartment, number of bedrooms, number of bathrooms, the price she is willing to rent for, and any other details that she thinks another fellow student **renter** should know. Once her listing is posted, she will receive email notifications if another student is interested in her apartment, and will be able to communicate with the other student to make any necessary arrangements.

4. Owner, Registered User

Candice is an elderly woman who has several properties that she would like to rent to SFSU students. She has very minimal computer skills but is able to register and post her **apartment listing**s to our website because the **user interface** is clean and intuitive. She is able to manage multiple **apartment listing**s on a detailed and organized **summary page**.

5. Administrator

Francine has been hired to keep David's List clean and friendly for students. She is able to edit and remove any **apartment listing** that she deems inappropriate for the website. She will be able to issue warnings to **registered users** if they violate the site's Terms and Conditions and ban **registered users** for repeat offences. She accomplishes this using the MySQL Workbench.

3. Data Definition

1. **Registered User –** User with an account; can be renter or owner.
    a. **Email** – Email address associated with account.
    b. **Phone** – Optional phone number associated with account.
    c. **Bio** – An optional user description.
    d. **Image** – An optional image.
2. **Guest User** – User without an account; can browse site for apartments; cannot rent or rent out.
3. **Renter** – Registered User seeking apartment.
4. **Owner** – Registered User posting apartment.
5. **Listing** – An apartment/room/share for rent.
    a. **Number of Bedrooms**
    b. **Number of Bathrooms**
    c. **Internet** – Capability, not whether or not it is provided.
    d. **Pet Policy**
    e. **Elevator Access**
    f. **Furnishing**
    g. **Air Conditioning**
    h. **Address**
    i. **Description**
    j. **Image**
6. **Administrator (Admin)** – User with special privileges; has the ability to remove listings from the site, issue warnings and bans from the site, and generally enforce the Terms and Conditions of the site.

4. Initial List of Functional Specifications

1. Guest users shall be able to browse through the site, but not have access to post items or contact registered users.

2. All users shall be able to create accounts and login.
3. All users shall be presented with the Terms and Conditions and Privacy Policy.
4. All users shall be able to search for apartments.
5. All users shall be able to filter the search based on whole apartments or just rooms.

6. Registered users shall be able to post create and post listings.
7. Registered users shall be able to be verified based on SFSU email.
8. Registered users shall be able to reset their password should they forget it.
9. Registered users shall be able to create a biography.
10. Registered users shall be able to save listings to their Favorites.
11. Registered users shall be able manage their accounts; to be able to remove or edit their listings.

12. Listings shall display at least one picture of the apartment or room to be rented, the price, the address.
13. The site shall show recent apartment postings.
14. The site shall use Google Maps integration to display the approximate location of the apartment.
15. The site shall provide directions between SFSU and the apartment and give the time it takes to travel from the apartment to SFSU.
16. The site shall show a rating of a registered user, based on votes from other users.
17. The site shall display nearby restaurants on Google Maps.

18. Admins shall be able to delete accounts.
19. Admins shall be able to delete postings.

5. List of Non-Functional Specifications

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).

6. Competitive Analysis

	David's List	Rent.com	Trulia	Craigslist
Search Filters	+	+	+	+
Map Usage	++	+	+	++
Ease of Renting Out	+	+	+	++
Student Verification	++	-	-	-

KEY

+	Feature exists
++	Application excels at this feature
-	Feature missing or ineffective

All competitors provide many filters for apartment searches, including number of bedrooms, number of bathrooms, pet policy, and other details. In our plan, the default search result will show listings on both a map and in a list, simultaneously. Users of Rent.com, Trulia, and Craigslist must switch to the map view in order to view results on a map. Craigslist also allows search results to be displayed by title, thumb, and gallery. Like all of our competitors, our plan provides users the option for renting out apartments. Craigslist provides apartment-seekers the ability to create a post indicating that they are looking for an apartment. Our plan's biggest advantage is in the area of owner-renter trust. Since SFSU students are able to verify themselves as students, both owners and renters will benefit from the inherent layer of trust provided when one sees that the other is a verified student.

7. High-Level System Architecture

Web Stack – LAMP

- **L**inux Operating System
- **A**pache Web Server
- **M**ySQL Database
- **P**HP programming/scripting language

Deployment – Amazon Cloud

- Great reliable service

Dynamic UI – JavaScript/jQuery

- Interactive front end
- Dynamic coding scripts

Database – MySQL

- Industry standard for database
- Used in L.A.M.P software bundle

UI – Bootstrap

- Supports popular browsers like Chrome, Firefox, and Internet Explorer
- Built-in scaling
- Minimum HTML needed
- Contains HTML, CSS, and JavaScript based templates

MVC Framework – Mini Framework

- Minimalistic MVC framework in PHP
- Low learning curve

Google API – Google Maps

- Allows users to determine how far the apartment is from the school
- Allows users to find directions from school to apartment

8. Team

Team Member	Role
Nora Schinkel	Team Lead
David Chau	Technical Lead
Paul Derugin	Back-end Developer
Steven Nguyen	Front-end Developer
Yansen Liu	Front-end Developer/UI/UX designer
Imran Irfan	Back-end Developer

References

[1] Professional website data analyzing company SimilarWeb: https://www.similarweb.com/top-websites