CSC 648/848 Fall 2017

Milestone 1: Use cases, High Level Requirements and Architecture

HouseShop

Franklin Henry Boswell, Team Lead (fboswell@mail.sfsu.edu)

Jesse Christiansen, Front End Dev

Nehemya McCarter-Ribakoff, Backend Lead

Khanh Nguyen, Front End Lead

Evgeny Stukalov, Backend Dev

Kevin Lay, Backend Dev

Revision History

Date	**Revision**
10/1/17	Rough draft submitted

HouseShop

Table of Contents

Executive Summary

The real estate business is notably one of the most stable markets worldwide, post 2008, the housing market in San Francisco alone has seen a steady increase of 9.5% a year. There is a huge market readily available and here at HouseShop we have identified the key issues homeowners and homebuyers face while shopping real estate. We understand that not everyone has a luxury of having a real estate agent find them the right deal for them, and that not everyone has the opportunity to have bids electronically submitted online, but this is where HouseShop really stands out. Our product puts everyone at a even level, so no one has an advantage over someone else. This really assures our users that they are playing on a fair battleground.

We believe that real estate should not only be accessible to all types of people, but it should also be effortless to them. HouseShop extracts the hardships and burdens of buying or selling a house and throws them out the window. Our principal mission here is to combine simplicity to our customers while hiding the complex fixtures working simultaneously, to provide a truly wonderful experience to our customers.

Our solution to the competition is simple, we're offering a effortless experience that other real estate websites just can't compete with. Other websites are bulky and contain too many things that overwhelm their users. We can solve this problem by just offering the right amount. We've planned user testing around optimizing a simple interface that both homebuyers and homeowners can feel comfortable using. Real estate is a near 1 trillion dollar industry and no real estate website has captured the market completely yet. This is the perfect time for HouseShop because the market is so heavily saturated between buyers and sellers.

Our team here at HouseShop is highly experienced and qualified in building cutting edge websites that combine simplicity at the front, with complex algorithms running simultaneously in the core. HouseShop users can trust that their experience will be unlike any other and that they are playing on a fair battleground when it comes to real estate.

Use Cases

Case 1. Non registered user

Phuong is a 45-year-old mother of three. All of her children have moved away for college and she is thinking of moving. She knows how to use a PC, but does not bother with that smartphone stuff. She finds HouseShop in an internet search and begins browsing listings. She wants to stay in the area, so she sorts **listings** by zip code. She also wants to save money for retirement, so she adds a max price **filter** to see what is in her price range. She sees several listings she likes, so she decides to save them to create an account and save them to her **watchlist**.

Case 2. Registered user (buyer)

Zach is a 23-year-old startup CEO. He lives and breathes tech, and he recently received a massive endowment from an angel investor. He is looking to move out of his hacker house into a nice million-dollar home. He logs into HouseShop and checks on his **watchlist** to see if any **listings** have changed. Two houses have gone down in price, and one has sold. This gives Zach a fear of missing out, a feeling he experiences regularly. He decides to pursue a home. He sends a message to the seller's **inbox** listed and schedules a tour.

Case 3. Registered user (seller)

Maggie is a 27-year old musician living in a condo in New York city. She recently signed a record deal with a big music company in Los Angeles and decided to sell her current place to move there. She visits the HouseShop website to request and schedule a listing consultant with an agent. She first signs up for an account to enter basic personal information including the current condition of her condo such as address, **square footage**, and **number of bedrooms**. She will be then contacted by an agent to discuss in more details of how she can proceed selling her place through her **inbox**. Maggie sets

the price based on her needs and estimation from the **listing agent**. Then, she hires professional photographers to take photos of her condo so she can upload **images**. Maggie can view how other people are viewing her place and what the agent is doing to sell it through her **dashboard**. Once the agent messages Maggie with different offers, she then decides and select which buyer to sell to and close the deal.

Case 4. Listing agent

Mark is a **listing agent** at HouseShop. Mark logs onto his **account** as a listing agent to check his **inbox** for any messages from interested buyers or sellers. He can also view the **listing(s)** he posted to track the popularity of those properties on his **dashboard**. Once a user contacted Mark, he can reply to consult the seller or negotiate with the buyer. He can use the information provided by the seller to create and post a new **listing**.

Case 5. Admin

Jimela is the site **admin** and logs on to check how the site is doing. Jimela cannot buy or sell as an admin. She has a few priorities to check on: user reports of two kinds: **listings** that are inappropriate or suspicious, and **accounts** that have been flagged by other users as abusive or fraudulent. Jimela references the Terms of Use to determine if a report is valid. She may do any of the following for each report: warn the user via their contact info that they have violated the Terms of Use, delete the **listing**, and/or remove the user account.

HouseShop

Data Definition

Account Holder: Any person with a registered account; may be an admin, agent, or user

Admin: The site administrator; Individual who may remove listings, delete user accounts, and perform general site management

Listing Agent: A user who is a verified real estate agent. An agent may both post listings and assist other users in selling their own listings

Browse: the ability to view listings posted on the site

Dashboard: An account holder's homepage. The content of a dashboard depends on the type of account (individual vs. agent vs. admin)

Filter: a flag one may place on results while browsing to display listings in order of: price, zip code, number of bedrooms, number of bathrooms, square footage, etc.

Homeowner: A user that is trying to rent one of their houses. The user is not an agent of and is not verified.

Image: A picture of a listing, agent, or user.

Inbox: A list of messages a user may receive from other users or the admin.

Lazy Registration: the ability to browse listings without creating an account

Listing: an ad posting of a house or plot

Notifications: an alert of recent activity relevant to a user's interests; includes new messages, changes in watchlist, etc.

Number of Bedrooms: the number of bedrooms in a listing

Number of Bathrooms: the number of bathrooms in a listing

User: an individual who is buying or selling. A user is not an agent or an admin

Unregistered User: a guest on the site without a registered account

Square Footage: the square footage of a listing

Watchlist: An element of a user's dashboard where they may save listings to track if they change in price or are sold

HouseShop

High Level List of Functional Requirements

Users:

1. Users shall be able to navigate through all of our offered pages intuitively
    a. For example an easy, intuitive way to navigate from home page, to listings, to user's account etc..
2. Users shall be able to create an account, login, and manage their account
    a. To create an account user must type a few basic information, such as user name, password, email and security question. User will be able to login into this account using the information initially entered as well as password recovery.
3. Users shall be able to save homes they wish the view again later and access a list of saved homes.
    a. If the user is signed in, he/she can save listings that they want to refer back to later. These saved listings will be accessible at any time.
4. Users shall be able to sort listings based on a variety of categories.
    a. Sorting listings based on various categories ie. price, square footage, bedrooms, location and so on.
5. Users shall be able to find sellers contact information
    a. Once listing is clicked on, contact information of the seller will be easily visible.
6. Users shall be able to contact an agent if he/she wishes to sell their home.
    a. Users can request the ability to contact an agent if they navigate to their manage account options.
7. Users shall be able to access an inbox to receive and send messages
    a. There will be an inbox button next to the account login/register that allows users to send and receive messages.

8. Users shall be able to upload pictures or videos of their home they wish to sell
    a. Once user uploads listing, they will have an upload pictures / videos to attach to their listing.

Listing Agent:

9. Listing Agents shall be able to access a dashboard
    a. This dashboard will be the Listing Agent's account management page and will provide all the functionality the Listing Agent will need, including: inbox management, activity on each listing, etc.

Admin:

10. Admin shall have the ability to see flagged listings that may be in violation to the terms and services
    a. Admin will receive notifications when a user flags a listing, these notifications will appear as messages in the admins inbox.
11. Admin shall be able to access the terms and services to refer back to them
    a. The ability to quickly view the terms and services for the admin will be accessible.
12. Admin shall have the ability to delete or remove the user's account of a listing that violates the terms of services
    a. The admin will see an option to remove a listing. There will also be an option to remove a user's account and delete it from the database.

HouseShop

List of non-functional requirements

---

1. Application shall be developed and deployed using class provided deployment 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 must be explicitly approved by Anthony Souza 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 e.g. must render correctly on the two latest versions of all major browsers: Mozilla, Chrome.
5. Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile native app is to be developed
6. Data shall be stored in the MySQL database on the class server in the team's account
7. Application shall provide real-estate images and optionally video
8. Maps showing real-estate location shall be required
9. Application shall be deployed from the team's account on AWS
10. No more than 50 concurrent users shall be accessing the application at any time
11. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
12. The language used shall be English.
13. Application shall be very easy to use and intuitive. No prior training shall be required to use the website.
14. Google analytics shall be added
15. Messaging between users shall be done only by class approved methods and not via e-mail clients in order to avoid issues of security with e-mail services.
16. Pay functionality (how to pay for goods and services) shall not be implemented.
17. Site security: basic best practices shall be applied (as covered in the class)

18. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
19. The website shall prominently display the following text on all pages *"SFSU Software Engineering Project, Fall 2017. For Demonstration Only"*. (Important so as to not confuse this with a real application).

HouseShop

Competitive Analysis

Feature	Zillow	Trulia	Realtor	Our Product
Text Search	+	+	+	+
Browse	+	+	+	+
Video/Picture Previews	++	+	+	+
Register/Login	+	+	+	+
Save option	+	+	+	+
Filter options (i.e., by price, # of bedrooms, etc.)	++	++	++	++

+Feature exists; ++Superior feature; - Does not exist

Our planned product will have all listed features as our fellow competitors do. We intend to provide a more intuitive experience to our user. This will include a flushed out browsing and sorting option that allow the users to easily navigate to their ideal home. The ability to navigate to sign in and save homes will be as simple as needed and extremely user friendly, and quick to get started. Zillow's filter system offers to sort by newest, cheapest, and then a drop down menu of eight more options, trulia offers the user to sort by price, bedrooms, home types, if its for sale or not and a mores tab. Realtors offers the user to sort through mostly the same as the other two but on top of those, bathrooms and amenities as well.

High-level system architecture

HTTP server: nginx (lightweight web server, also handles request routing)
Front End layer: Bootstrap, EJS (Javascript template engine for Node.js)
Application layer: Express (web framework for Node.js), Node.js (server-side Javascript runtime environment)
Additional Node.js libraries: mysql2 (MySQL client for Node.js), sequelize (object-relational mapping tool for Node.js)
Node.js infrastructure: npm (package manager for Node.js), PM2 (process manager for Node.js)

Data layer: MySQL (relational database management system)

Deployment platform: Ubuntu (GNU/Linux distribution), GNU/Linux (Unix-like operating system), Google Compute Engine (virtual computing environment).

Supported browsers:
Google Chrome (two latest versions)
Mozilla Firefox (two latest versions)

Team

---

1. Franklin Henry Boswell, **Team Lead**
2. Khanh Nguyen, **Front-end Lead Developer**
3. Nemi McCarter-Ribakoff, **Backend Lead Developer**
4. Eugene Stukalov , **Backend Developer**
5. Jesse Christiansen, **Front-end Developer**
6. Kevin Lay, **Backend developer**

HouseShop

Checklist

- Team decided on basic means of communications
  Done
- Team found a time slot to meet outside of the class
  Done
- Front and back end team leads chosen
  Done
- Github master chosen
  Done
- Team ready and able to use the chosen back and front end frameworks
  Done
- Skills of each team member defined and known to all
  Done
- Team lead ensured that all team members read the final M1 and agree/understand it before submission
  Done