SW Engineering CSC 648/848, Fall 2017

Real Estate Website -
California State Real Estate

Team 08 - Section 2 - global team

Team Lead: Falco Becker (falcobecker@freenet.de)
Members: Richard Robinson, Karan Gupta, Vidit Manglani

Milestone 1

4th October 2017

History Table

Date	**Revisions and Feedback**	**Page**
10/04/2017	First draft	
10/08/2017	Revised, frozen	1 to 12

Content:

1 – Executive Summary

Have you ever visited a real estate website and ended up more confused than when you started? Does your current real estate portal try to do much and yet achieve nothing at all?

‘California State Real Estate’ is a Real Estate Portal designed from the ground up to be a simple and effective way to search properties facilitate communication between buyer and agents.   It stands apart from its competitors by only delving in the important things, and doing them well. An interface that comforts the user, familiar like a fireplace. Yet detailed enough to be customizable to be the correct tool box for the job.

Offering only the essentials in a seamless and synergized interface appeals to essential ends of the web-portal, home buyers and property agents. We recognize the differing requirements of the our two clients, neither of whom we can exist without, and we also recognize the ways in which the resemble each other, the points their interests intersect with the requirements of the real estate marketplace and we make sure do those well, and most importantly implement the differences in design using subtle changes which require little cost to implement and provide the required distinctions.

With an interface approved by real estate professionals in the United States and the Database designed by a team in Germany. It enables the buyer to hone in on exactly what it is they are looking for and for the seller to work with a buyer, not a browser. Fluid in all the right places, with a modular design that is ready to change with the times and yet since it is built on the sound logic of the that is minimal and yet scalable.

With a homepage that encourages buy-in from our prospective clients: buyers and agents by displaying features that are inherently attractive to each sect of the differing perspectives and circumstances of our market(audience). A specialized interface for each of the three clients that will consume our product. Property buyers and shown a interface that suits their outlook.

The interface is seductive to buyers, it gives them the information they need, it helps them imagine their future home. The organized interface appeals to the prospective seller to make them feel that it will make their job easier. And encourages them to complete sales.

With a reliable, modular interface that works, it allows for changes in the future and further implementation of monetization strategies.

The messaging system ensures integrity of users.
And ultimately our portal aims to impress both buyers, and agents, not by brainwashing them with features but by delivering a tool that makes real estate interactions a simple, authenticated, uncomplicated affair.

2 – Use Cases

Use Case Nr. 1: unregistered user searches
A unregistered user, which could be a future customer, visits the web site with the intention to find a proper property. He/she has a imagination which characteristics the property shall have.
At first he/she wants to search for properties in certain locations. So the unregistered user can search with a offered search-function on the index-site for either a proper name of a city or a zip code.
With this information the system can show every registered property-offer out of the database.
The system also offers a bunch of filter-options, such as filter by price-scale, distance and number of bed/bath-rooms.
If the unregistered user is interested in an offer he/she can select it and will be leaded to a site with all the information needed, like google-maps location, price, bedrooms, bathrooms, owner, description and pricing.

Use Case Nr. 2: registered user wants to buy a bookmarked house
After a registered user found a proper offer with a suitable price he/she wants to buy it. At first the user needs to log in on the index-site with username and password. After this the user will be led to his own user-information-dashboard with the bookmarked houses. Those bookmarks can also be filtered by price-scale, distance and amount of bed/bath-rooms. After the wanted offer was found he/she can select it and will be led to the profile page of the house where the user can start the buying-transaction.

Use Case Nr. 3: owner offers house
If a owner of a property wants to sell his/her house on the website he/she needs to sign up at first. After the sign-up where the owner needs to give personal information the owner will be led to his/her dashboard.
This dashboard shall display all offered properties and incoming requests and buying requests.
For getting the new offer online, he/she needs to create a new offer. Here the owner needs to describe the property correctly with uploaded pictures, maybe a video-tour, short description, price, location and the number of rooms. After the owner confirmed the information the offer can be uploaded to the database.

Use Case Nr. 4: real estate agent wants to sell
If a real estate agent wants to sell a property he/she needs to log in at first. After the log in on the index-site the real estate agent will be led to a agent dashboard. This shall display a history of all transactions, active offers and incoming requests.
The real estate agent can select the request which he/she is aiming to accept and start the selling-transaction.

3 – Data Definition

Entities:

Name	**Description**	**Data**
User	Any entity that accesses the website	
Buyer	Any entity that is interested in purchasing a property	Can be registered or unregistered -> inheritance
Unregistered Buyer	A buyer who has not registered with the site	
Registered Buyer	A buyer who has registered with the web site	● ID ● Name ● Location ● Contact Info ● Bookmarks ● Mailbox
Seller	Any entity that is interested in selling a property, needs to be registered	Can be a owner or a real estate agent -> inheritance
Owner	A private person who wants to sell a property, registered	● ID ● Name ● Location ● Contact Info ● Description ● Bookmarks ● Postings ● Mailbox
Real Estate Agent	A professional seller of properties, registered	● ID ● Name ● Company ● Location ● Contact Info ● Description ● Bookmarks ● Postings ● Mailbox
Property	A house that has been listed	● ID (Posting) ● Size ● Location ● Bedrooms ● Bathrooms ● Price ● Owner/Seller ID ● Description

4 – Functional Requirements

1. **Login System:**
   User gives a Username and Password, which shall be checked by the system to give proper access
2. **Multiple property management:**
   Manage a database which handles all information about the offered properties
3. **Dashboard Pages:**
   Registered buyers and sellers shall be able to organize their properties and respective communications on a single page.
4. **Featured Properties:**
   A page area that shall display interesting properties on the Homepage, selected by a criteria such as popularity, and/or for paying registered sellers.
5. **Search Function**:
   Shall be implemented, in minimum, as a text based way to query the property database.
6. **Filter Functions:**
   Users shall be able to filter their searches and browsing by price-scale, distance and number of bed/bath-rooms.
7. **Browse Page**:
   A place for buyers, and a separate place for sellers, to see related properties using keywords, categories or schema.
8. **Messaging System:**
   A way for buyers to express interest and inquire about certain properties and for sellers to manage each communication, even ones from unregistered users shall be integrated into the interface.
9. **Location on Map:**
   A map shall be visible on every property's opening page. With links to walking score, stores nearby and relevant tie-ins.

Functional Requirements - Format Option B

1. Search Function shall be implemented to access the property database.
2. Users shall be able to filter their searches
3. The system shall auto-populate home pages with featured properties
4. The website shall allow unregistered users to message the agents
5. A map shall be visible on every property's opening page. With links to walking score, stores nearby and relevant tie-ins.
6. System shall require login to give to access the advanced features
7. Registered users and agents shall be able to organize their properties and respective communications on a single page.

5 – 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, Safari, 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

6 – Competitive Analysis

WEBSITES	**Map Split**	**Search with Filters**	**Sign up with email**	**Top properties displayed**
CSRE	*Yes*	*Yes*	*Yes*	*Yes*
zillow.com	Yes	Yes	Yes	Yes
realestate.com	No	No	Yes	No
realtor.com	No	No	Yes	Yes
redfin.com	Yes	No	No	No
climbsf.com	No	No	Yes	Yes
mcguire.com	No	Yes	No	No
trulia.com	Yes	Yes	Yes	Yes

WEBSITES	**Split Buy/Sell option**	**Floorplan**	**Agent Finder**	**Property Improvement**
CSRE	*Yes*	*Yes*	*No*	*No*
zillow.com	Yes	Yes	Yes	Yes
realestate.com	No	No	No	Yes
realtor.com	Yes	Yes	Yes	Yes
redfin.com	No	No	Yes	No
climbsf.com	No	No	Yes	No
mcguire.com	No	No	No	No
trulia.com	No	No	No	No

Additional overview of the Competitive Analysis

Name/Feature	Design	Communication on the site	Description of offers	Usability	Number of Features
CSRE	+	+	+	+	-
Zillow	+	+	+	-	+
Realtor	-	-	-	-	-
Trulia	+	-	-	+	-

Additional Description

Name of Website	**Features**	**Advantages**
CSRE	● Buy ● Sell ● Rent ● Filter ● Maps ● Searching ● Favorite properties	● Unregistered buyers may request contact with sellers ● Easy Usage ● Good Performance
Zillow	● Buy ● Rent ● Sell ● Mortgage ● Find an Agent ● Home Design ● Blogs and Forums	● mildly animated ● relaxing picture on main page ● sign in and join buttons are separate ● forums for the particular website ● Design ideas ● Can find good school districts ● zillow-specific terms that are catchy
Realtor	● Buy ● Rent ● Sell ● Find Realtors ● Take out loans ● News ● Able to look at current home ● Feedback option ● Careers link ● Shows recently sold houses	● Looks more modern, with news, your home estimate +Encourages lending by making it look easy ● Logging in and registering as a user is very clean and clear. The separation between different account types is likewise clear. ● Encourages action
Trulia	● Buy ● Rent ● Mortgage ● Sell ● News	● Saved searches and homes is extremely easy to access. ● Simple looking ● Search is always visible on main page

7 – High Level System Architecture

HTML5

- The standard language for web-applications. It is capable of being formatted. Objects, images, videos, and other media may be embedded.
- Usable on most platforms HTML5 is the standard for web development, and as such this project should implement it.

JavaScript

- Supported by most modern web browsers --Supports event-driven, functional, and imperative programming
- Can be run client or server side.
- JavaScript is a standard for web development. It is extremely supported and widely used. JavaScript is diverse enough to develop anything from simple arithmetic operations to entire games.

CSS

- Eases formatting for HTML components
- Changes can be made in CSS files and applied automatically to all HTML files that implement the CSS files
- Allows for consistency, maintainability, and saves numerous headaches over using HTML without CSS
- CSS is yet another of the standards for web development. It's essential for working efficiently; saving time from refactoring numerous files whenever a change in formatting desired.

NODE.JS (Express)

- open source
- asynchronous input and output
- platform independent
- dynamic events can be run on server before sending to a user
- Node allows for the running of dynamic components on a server smoothly and without interference with input or output.

PUG

- Templating language
- Cleaner and less cluttered than HTML, more readable.
- Is converted into HTML
- Pug was chosen as a templating language because the syntax used for it is more readable than HTML. HTML contains a lot of clutter with the angled brackets and it can be time-consuming to sift through components. Pug should allow for more malleable and maintainable code due to it being easier to read.

Bootstrap

- Many prebuilt components for web formatting and scripting
- Allows for easy scalability on mobile devices
- Can break components down into grids, allowing for precise and scaled positioning with formats
- Eases resizing, when a window's size is changed the components will generally shift in a readable fashion
- Bootstrap allows for easy scaling, supports grids, and makes navigation bars incredibly simple. Dropdown bars, footers, carousels, forms, and many other components can also be easily accessed. There is a ton of templates that utilize bootstrap, many of which are usable by anyone. If there is something already implemented that does exactly what is needed there is no reason to reinvent that component.

SQL

- Query language
- Update, add to, delete, and retrieve data from databases
- Can structure tables such that items that do not satisfy prerequisites cannot be added
- May cascade on certain operations
- SQL is an easy query language that has a lot of optimization that occurs behind the scenes. A user could make a request in an inefficient manner and it will be somewhat optimized. SQL is easy to learn, easy to apply, and extremely useful.

Google Maps

- May imbed google maps into a web-environment
- Simple to set up Google maps will be extremely useful for the creation of a real-estate website.

Google Analytics

- Can be used to track web traffic
- Can report findings, which can later be interpreted or further analyzed

Workbench

- User interface for administrators/database management.
- Facilitates working with a database
- May automatically fill and reuse certain queries. Workbench makes maintenance and usage of the database easier.

PM2

- Process manager for Node.js
- Allows for easy running, and monitoring of the server PM2 is the process manager that is running on our server.

SASS

- Eases writing of CSS
- Can use variable options, allowing for components to be modified in the future

Supported Browsers (latest two versions)

- Firefox
- Chrome
- Opera

8 – The Team

Surname	First Name	Position	Location	Mail
Becker	Falco	Team Lead	USA	falcobecker@freenet.de
Robinson	Richard	Back Lead	USA	richardjohnrobinson@gmail.com
Manglani	Vidit Joy	Front Lead	USA	viditjoy@gmail.com
Gupta	Karan	Member	USA	karangupta.k9@gmail.com

9 – 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
  - ON TRACK

- Skills of each team member defined and known to all
  - ON TRACK

- Team lead ensured that all team members read the final M1 and agree/understand it before submission
  - DONE