

Phase 1: Definition

Team Leader: Preston Engstrom
Working Group: Tyler Kennedy Collins
Yucen Jin
Jaclyn Binch
Andrew Rooney
Jeff Yang

Abstract

This document describes the definition phase of Homestead. Homestead is a web application that provides services in the area of rental, sublet searching, and posting for tenants as well as landlords.

1 Context

As attitudes and economics change, more of the adult population is shifting away from home ownership towards renting. This reality combined with web applications becoming ubiquitous, create a market for Homestead.

Homestead provides an effective service by offering the landlords a simple and easy way to effectively list their properties. For tenants, Homestead allows for custom searching, alerts to mobile devices, reviews of landlords via a comments section, and a way to view past listings.

2 Objectives

The primary goal of the project is to create an attractive, stable, and scalable web service which upon acceptance can rapidly be turned into a functional business. The web application will present a platform to allow for the advertisement, searching and viewing of rental properties. These include student housing and sublets. The application will provide two interfaces to serve the two client groups, being Landlords and tenants. Tenants will be able to search for postings via criteria or using a map interface. Tenants will also be able to view properties from the site, as well as review comments from previous tenants regarding the landlord or property. Landlords will conversely be provided with an interface to create and edit postings, interact with and vet potential tenants and view metrics on other posting to stay competitive. Any user will be able to save searches for a particular area to monitor when a posting becomes available. The same user will be notified of postings meeting their criteria via SMS or email, options they will be able to select as part of their profile.

3 Features Supporting Objectives

- Mapping and Geo-location service.
- Video streaming service for viewing rooms.
- Street level view through the Google Maps API.
- Cross platform support.
- System to evaluate and rank postings given a set of criteria.
- Automated interaction with users via SMS and email.
- Persistent instant messaging service to support direct communication of tenant and landlord users via Homestead.
- Commenting and rating system for tenants and landlords.
- Simple, interactive, and easy to use searching interface for listings.

4 Requirements to Implement Features

- Web server with LAMP stack (Linux, Apache, MySQL, PHP) for testing and deployment.
- Google Developer account/Access to Google API's.
- Software as well as hardware infrastructure for sending SMS messages.
- Properties/Interior images to use as preliminary test data.
- Front and back end development frameworks to avoid creation of solutions to previously solved problems. **See Below**

4.1 Use of Frameworks in Development

Programming frameworks provide a set of pre-built tools and components to avoid reinventing the wheel. Many open source frameworks exist for many languages. For our purposes, we have selected two frameworks, discussed in sections 5.1 and 5.2.

5 Development Environment

Our development team will be working on a variety of platforms for local development, testing and production fixes. To solve this problem we have standardized the environments we are using. This section gives an overview of our frameworks and environments.

5.1 Front End Framework: Bootstrap

The front end of the application will be built by extending the Bootstrap front end framework. Bootstrap is an open source collection of CSS classes to be applied on website elements. The primary goal of Bootstrap is to allow developers the ability to create websites that are responsive based on the screen they are being viewed on. This solves the problem of users being unable to view our website on anything but a mobile device. They will instead be presented with a properly scaled version of it.

5.2 Back End Framework: Laravel 5.2

Laravel is a PHP framework that provides standard classes and stable systems for interacting with databases. Creating front end templates and many features that are otherwise simple but tedious to reimplemented is another main reason for our choice. By using a framework that provides standard features, such as user authentication through Google, Facebook, and Twitter, it allows the development team to focus on significantly more advanced problems. These include full support of mobile platforms, a more responsive website, and integration of SMS sending.

5.3 Local Development

Team members will be developing on a variety of platforms. To maintain consistency throughout the process, all development will take place on virtual machines. Each team member will have an identical machine, hosted using Oracle Virtualbox. This allows all team members to develop in an environment that matches the production set up. This also goes on to provides consistency regardless of the developers physical platform. This is advantageous as team members are developing on Microsoft Windows, Apple OSX and a variety of Linux OS distributions. This human resource complexity is removed by the use of virtual machines.

6 Source Control

All source code, documentation, business documents, meeting minuets, and any other pertinent information will be maintained via the Git version control system.

6.1 Git Project Name Selection

When first read, project names are often overlooked as an important part of a team development environment. As the software development industry has grown, large corporations and small startups alike have begun to converge on the idea of fostering a comfortable work environment. By generating project names that seem nonsensical, but memorable, teams also foster a more cohesive environment. To this end, the team has elected to randomly generate a project name. The selected name is **Barbaric Waffle**. This name, achieves the memorability and approachability that other companies, including Google, Facebook and Twitter, seek to achieve in their development projects.

7 Acceptance Criteria

The criteria found below make up our minimum acceptance level for the application to be considered completed.

1. Application must be at a point that it is ready to be deployed in a business environment.
2. The Application will present a usable, modern interface on all platforms.
3. Support for a mobile client must exist. This can take the form of a web application.
4. Users will be able to view listings made by landlords.

5. Information regarding listings will have criteria for location, amenities, and more.
6. Listings will be indexable for criteria, and on via a map interface.
7. Tenants will be able to view previously viewed listings.
8. User must be able to receive alerts regarding creation of new listings.

8 Personnel Roles

All team members will take up general programming and documentation roles, combined with their designated roles. All team members will participate in regular bi-weekly code reviews and interface reviews.

8.1 Team Leader

The team leader serves as the Chief Designer, Senior Programmer, and Chief Technical Manager. They are responsible for analysis and design, source control, data control, reviews, user documentation, writing critical source components, and assigning work to other team members.

8.2 Deputy Leader

The Deputy Leader shares all roles of the Team Leader. The Deputy Leader is also ready at any time to take over as the Team Leader.

8.3 Test Leader

Responsible for overseeing testing the application. Provides first review of all changes before moving from development to staging environment. Is prepared to take over at any time as Deputy Leader.

8.4 Test Programmer

Collaborates with test leader in developing test cases and testing builds.

8.5 Technical Librarian

Maintains documentation library. Prepares input data, cares for server infrastructure, and meeting minutes.

8.6 Technical Writer

Assists team leader in preparation of documentation, support documents, phase documents, and reports.

8.7 General Programmer

Responsible for detailed design and programming of front and back end components.

8.8 Team Testing Structure

Testing of features will be done by all team members, excluding the member responsible for the development of that feature. It should be noted that the goal of this structure is to allow members to receive constructive criticism from the group as a whole so that all members can learn from what is done well. This is to avoid repeated mistakes.

9 Team Members

- **Tyler Kennedy Collins** Deputy Leader
- **Jaclyn Binch** Technical Writer / Test Programmer
- **Yucen Jin** Technical Librarian / Test Programmer
- **Jeff Yang** General Programmer / Technical Writer
- **Alex Lawrence** Test Leader / Test Programmer
- **Andrew Rooney** General Programmer / Technical Writer
- **Preston Engstrom** Team Leader / Technical Writer

10 Team Organization

The project will follow the Agile SCRUM development methodology. Development will be broken into 3 groups; Front End, Back End and Database. Each group will deal with an equally critical segment of development. All three groups will meet together at bi-weekly meetings. Each group will then meet separately, directly after each team meeting.

11 Preliminary Design