Crowdsourced Space Rental Web Application

| April 21, 2017

Project Purpose

To build and deploy a dynamically-scaling Airbnb clone to suit the needs of Neiybor, a local startup.

Project Importance

The rental space industry is a $32 billion dollar[1] industry and Neiybor is poised to disrupt it. The company has been in existence for over a year and has won $21,000 in business competition prizes. The founding team is composed of three business-savvy students, but they all lack technical expertise. They have a working minimum viable product (app.neiybor.com), but it is fraught with bugs, is built in a painfully antiquated framework, is expensive to maintain, handles payments poorly, and has heinous page load times. There is both great opportunity and great need that this project potentially can fulfill.

This project will be based upon various concepts that have been taught in my major and are in the information systems industry. Some of these concepts include Material Design, Responsive Design, Scalability, Relational Database Management, API integration, and MVC Web Application Development.

Project Overview

This project is a culminating effort that will test my knowledge of strategic technology decision making and app design, development, and deployment in every way. It's a big project. I've outlined (1) the core concepts that will drive the successful completion of the project and (2) the detailed specifications of the project below.

Key Concepts

Material Design-

Google Inc. compiled Material Design (what it calls a living framework) to bring order to the wild west that is website design. Its goal is to "create a visual language that synthesizes classic principles of good design with the innovation and possibility of technology and science[2]." The design philosophy has three core principles: material is the metaphor; bold, graphic, intentional; and motion provides meaning. Google has outlined best practices down to the miniscule details of a page, and I hope to infuse the website with material elements to "delight" the user (which is the goal of material design). These principles will inform all my design choices on the front end.

Responsive Design-

I cannot go into the front end design of this app with only a Material perspective. This website needs to be responsive. This means that the app will dynamically see if the viewport is a phone or a larger computer and adjust its design accordingly. This app will need to be accessible from a phone or a computer and therefore I will need to keep this in mind from the very beginnings of app design, or it will become a mess down the road as I try to make it compatible across all viewports.

[1] https://www.sparefoot.com/self-storage/news/1432-self-storage-industry-statistics/

[2] https://material.io/guidelines/material-design/introduction.html#introduction-principles

Scalability

Neiybor is designed to scale. Big time. And as the business will scale, the app will also need to scale. The scalable components of the app are (1) the code base and (2) the load capability of the information system.

1. The Code Base. I have chosen Ruby on Rails as the framework to build the app because it is a scalable framework. In Lehi, MX has proven that this can be successfully done with Rails, as demonstrated by the success of their products. I hope to improve upon their model by keeping my code *modular*, *abstracted*, and *readable*.
2. The Load Capability. The business cannot afford to have a site crash. That would be detrimental to the business' credibility and the user experience. Therefore, I will utilize services that cloud service providers have made readily available and are used by large enterprises worldwide. The infrastructure I will build the app upon will by dynamic—it will automatically scale to the load that the site is taking. It will spin up more computing power as needed.

Relational Database Management

Data consistency and availability is essential to the success of this app. Therefore, the database will be designed and implemented per proven industry practice. The schema will be designed in first, second, and third normal form. I'm strategically selecting, of all the database products out there, which will best serve the needs of this app, this company, and the company's future.

API Integration

There are a wealth of services and functions that are available on the web. Integrating these services can provide some awesome features for a cheap cost. I will strategically select and integrate payment services and geolocation services into this app that will provide the best return for benefit.

MVC Web Application Development

Related to scalability, this app will not succeed unless it follows strict procedures regarding the flow of data and the performance of server logic. Using principles of Model-View-Controller web development, I'll ensure each piece of the app is optimally abstracted. Additionally, I plan to separate as much as I can, thus ensuring that the app is modular and easy to adjust if necessary.

In short, the success of this project is dependent on my adherence to these powerful principles I've outlined above. I hope to draw on all of them to make a cohesive, slick web application that not only fulfills the needs of the business, but fulfills them well. This is a culminating work in my undergraduate experience, drawing on principles I've learned in every corner of my major.

Project Details

The proposed website needs 6 main components—user profile management, rental space search capability, an in-app messaging service, the ability to automatically perform monthly payments and customer charges as well as take Neiybor's cut out of each payment, a webmaster module to be used to manage users and listing approvals, and a property signup section of the site. Visually, the site will be built with Material Design principles as well as responsive design to provide a clean user experience. The website will be deployed to an auto-scaling infrastructure to account for the huge amount of growth the company expects to have. This will be done either through Amazon Web Service's Elastic Beanstalk product, or a similar product that Heroku may have. The app itself will be built with Ruby on Rails and probably run MySQL on the backend. This tech stack provides a nice balance between rapid feature development and modularity for future growth of the product. In a future day, the app will probably be built with web components on the frontend and a distributed NoSQL system on the backend, but this is simply a minimal viable product version 2, and time is of the essence. They plan on decking out the site after they raise a round of venture funding this upcoming Fall.

In the following sections, I'll briefly outline details the reader may be interested in regarding each core feature of the product.

User Profile Management

Signup, Login, Profile picture upload, profile details adjustment, viewing and editing of user listings, and viewing and editing of user rentings are all capabilities this module needs to have. This can easily be built with packages included in Ruby on Rails.

Rental Space Search

Most of this can be easily handled with Rails, but the geolocation function and the map-display function will be done through a Google Maps API. This is a big pain point on the current website that will be nice and customizable now that I'm building it from the ground up.

In-App Messaging Service

Users need to be able to send messages to hosts. This part of the app can be handled completely through the Rails framework.

Payment Management

This section of the project is a little more tricky. I don't have the resources to build a PCI compliant app from scratch, so the payment system will have to be outsourced. Stripe API has the exact functionality we need. It takes 3% of every payment, but we can exempt the monthly payments from this fee and reduce our cost to only 3% of the cut Neiybor receives. The API can take credit card payments as well, so I'll plug this into the site to give Neiybor the ability to seamlessly take and distribute payments, which is also a big pain point of the current system.

Webmaster

The site administrators need to be able to edit postings, user accounts, and approve listings. An optional feature of this module would be to view the in-app messaging of each user, but this is not an immediate need. This can all be done in Rails without a problem.

Property Signup

This part of the site needs the ability to upload photos of listings, to preview postings, to get verified and certified, and adjust the posting details. This can also be done in Rails without any complicated issues.

Thesis Advisors

Dr. Liddle is the head of the Rollins Center for Entrepreneurship and is an expert in the technologies I will be using. He is a self-proclaimed "propeller-head" in these technologies and has been working with them since he was very young. I was a high performing student in his *Web Development* class, and have a good relationship with him. (advisor).

Dr. Jenkins was the professor of my *Creating and Managing a Tech Startup* class and is the tech lead in a new venture himself. He was the one who built and validated his company's initial product, just as I am with Neiybor. Beyond discussions in class, I've met with him various times and he has given me & (reader) invaluable advice regarding this project and startups in general. He has a wealth of experience in exactly what I am doing with this project.

Project Timeline

This project will take an estimated 60-80 hours. The app will be split into two versions, the first of which should be deployed by the end of May, and the last deployed by the end of August. After that, within two months the project should be finalized and ready for the defense meeting in early November.