**Analysis Document**

John Haney

Elizabeth So

Kyle Voon

September 24, 2019

**Table of Contents**

[1. Project Scope 1](#_gjdgxs)

[2. User Discussion 1](#_30j0zll)

[Users 2](#_1fob9te)

[Website Pages 2](#_3znysh7)

[Home/Main Page 3](#_2et92p0)

[About Page 4](#_tyjcwt)

[Hours and Locations Page 5](#_3dy6vkm)

[Online Takeout Ordering Page 6](#_1t3h5sf)

[3. Developer Discussion 6](#_4d34og8)

[Website Pages 6](#_2s8eyo1)

[Main Page 6](#_17dp8vu)

[About Page 7](#_3rdcrjn)

[Hours and Locations Page 7](#_26in1rg)

[Online Takeout Ordering Page 8](#_lnxbz9)

[Non-Functional Requirements 8](#_35nkun2)

[Software Development 9](#_1ksv4uv)

[Technologies 9](#_44sinio)

[Platform 9](#_2jxsxqh)

[Software Tools 9](#_z337ya)

[Software Development Process 10](#_3j2qqm3)

[Deployment 10](#_1y810tw)

[4. Cost 11](#_4i7ojhp)

[Analysis 11](#_2xcytpi)

[Design 11](#_1ci93xb)

[Coding 11](#_3whwml4)

[Testing 12](#_2bn6wsx)

[Cost Calculations 12](#_qsh70q)

[5. System Development Timeline 13](#_3as4poj)

[5 weeks 13](#_1pxezwc)

[10 weeks 13](#_49x2ik5)

[15 weeks 14](#_2p2csry)

[Additional Features beyond Timeline 14](#_147n2zr)

[Admin Page 14](#_3o7alnk)

[Account Creation 15](#_23ckvvd)

[Google Analytics 15](#_ihv636)

[Reservation Page 15](#_32hioqz)

# Project Scope

This document describes the analysis for a website that will be developed for a casual fine dining restaurant featuring Asian cuisine. The website will consist of four pages: home/menu page, about page, hours and location page, and online takeout page. The developers will utilize technologies such as HTML and CSS and modern web design techniques to provide a responsive and interactive experience for the user regardless of the platform it is accessed on. Open-source technologies will be used to provide functionality in the most efficient way possible so that more time can be devoted to providing a more customized experience for the customer. This would include scaling page and image sizes to accommodate differing screen sizes. The responsive user-friendly website shall be delivered around 15 weeks after the development team was formed to discuss the proposal for the website. The website will contain features described in the “User Discussion” and “Developer Discussion” sections of this document.

# User Discussion

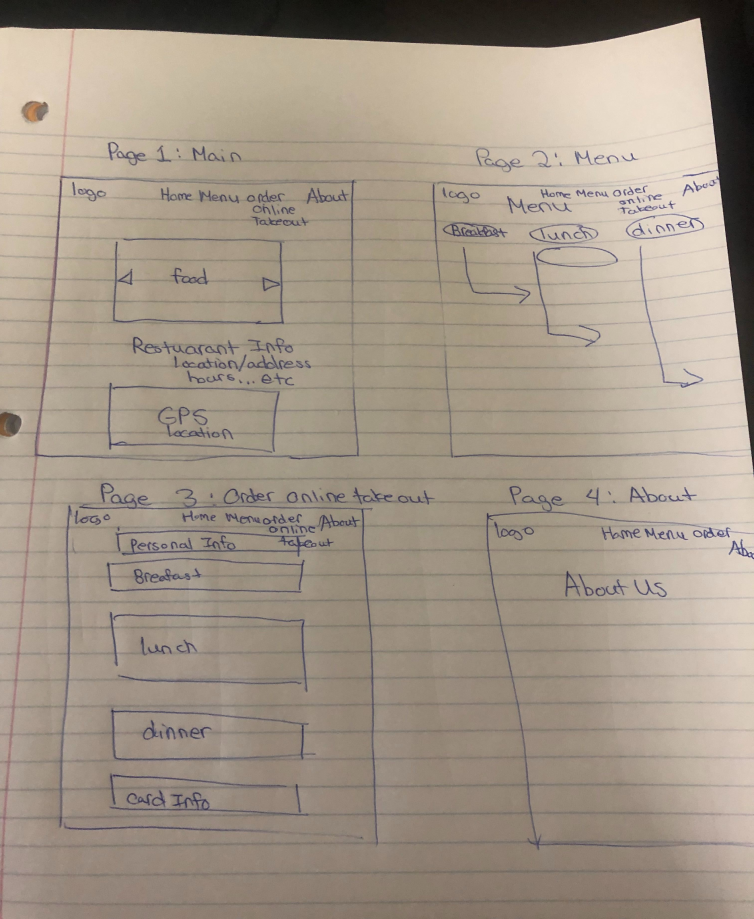
Due to the requirements specified by the users, it has been determined that a customized website would serve the users’ needs rather than an off-the-shelf product. The users will be able to select a layout and color theme to match the restaurant environment that the developers can implement. Engagement and information accessibility are the primary factors for an end user that interacts with the website. A well laid out interface and functionality are priority as user backgrounds will vary greatly. The most important pieces of information such as location and hours should be easy to find no matter which page the user is on. Pages should not overwhelm users with information. The information, whether it be regarding menu, mission statement, or navigation should be concise and easy to find so that it is accessible. Users should be able to navigate the site from any platform whether it be a computer, mobile device, or tablet. Pages should load quickly so the user isn’t bogged down when navigating.

## Users

The following users for the website are customers looking for a casual fine dining Asian cuisine restaurant. Targeted users shall range from first time customers to regular customers and from any age looking for any of the following information about the restaurant: menu items, pricing, and hours. Customers may range from a variety of backgrounds, ages, education levels, and technological know-how so an easily navigable page layout is necessary so that the end user may find what they are looking for. With such a wide range of users, the website needs to be as concise as possible. The site should be able to communicate the message of quality and accessibility to even those who may not be able to read English well or who may not be technologically adept.

## Website Pages

The following subsections include a description of the website requirements from a user perspective for the four pages of the website.



Initial mockup of pages to be implemented in the website[[1]](#footnote-0)

### Home/Main Page

The home/main page will contain the menu in the middle of the page and provide an entry point to the About, Hours and Locations, and Online Takeout Ordering pages on the website. All of the pages will contain a navigation bar at the top of the page with links to other pages and a section containing the address, hours, and phone number at the bottom of the page; in addition, the bottom of the main page shall contain the restaurant logo on the right side and all rights to the website for the restaurant.

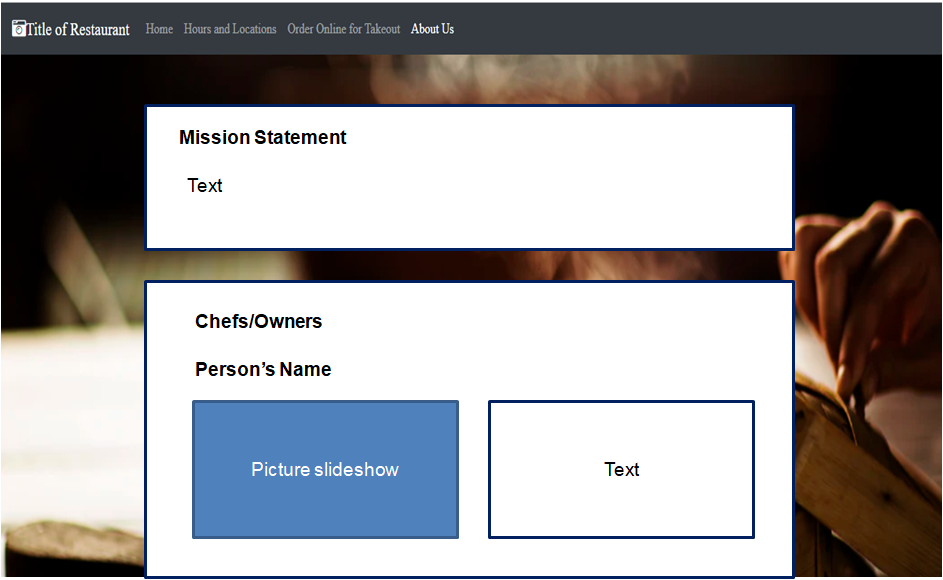
Clickable lunch and dinner icons will be displayed above the menu in the middle of the page. When users click on the icons, they will be directed to the point of interest. For example, when a user clicks on the lunch icon, the menu items for the lunch menu will be displayed. The items in the menu will be organized into different categories, such as appetizers, salads and soups, entrees, and cocktails. Individual menu items will be listed below each category and the text for the categories will be larger than the text of the individual menu items so that they are noticeable. Each menu item shall include the following: pictures, price, food sensitivities, and spice level.



Here is an example of how the menu items could be organized.

### About Page

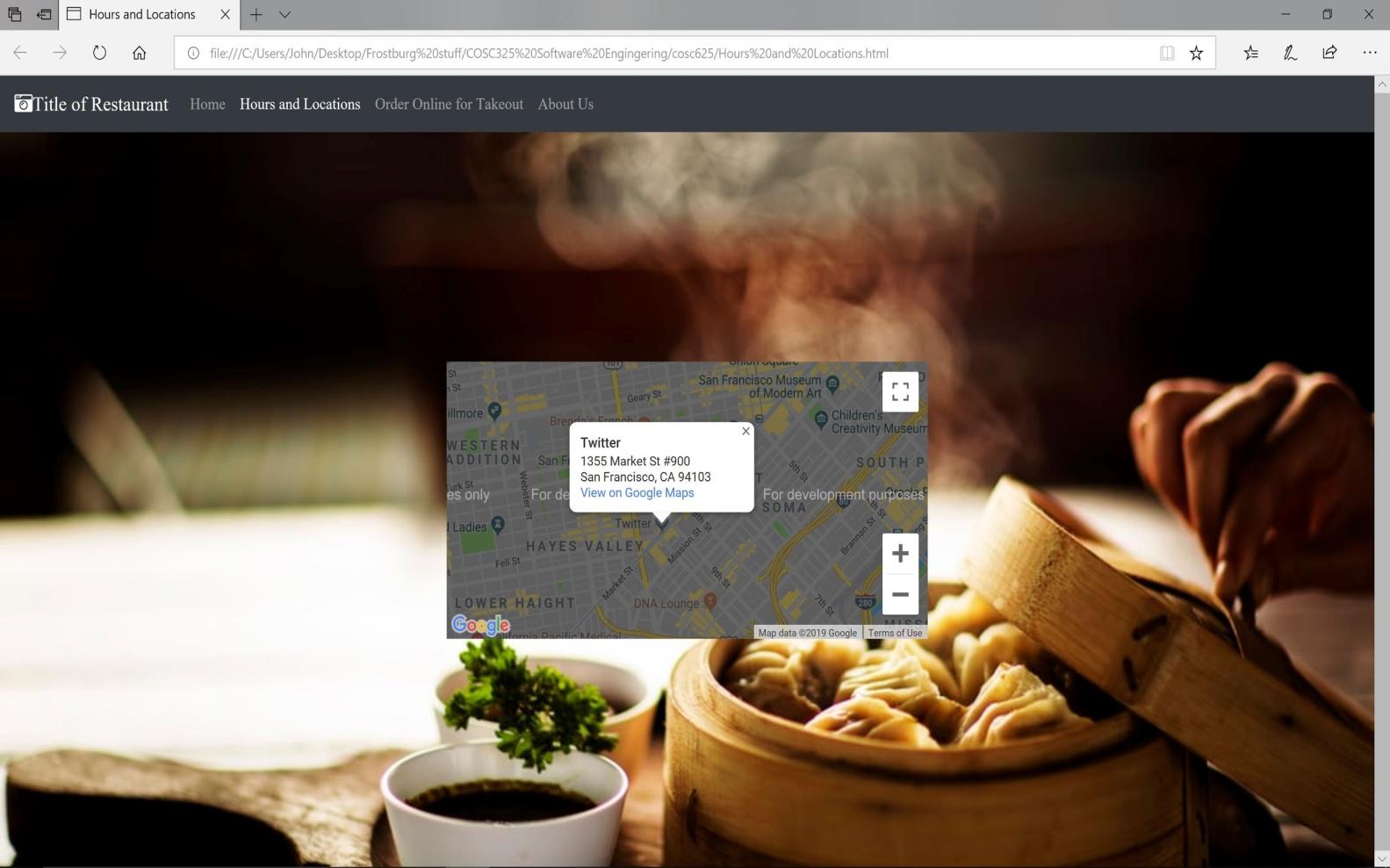
The about page will contain a mission statement and a picture slideshow of chefs/owners with additional biographical info of each person. Users should be able to scroll up and down the page to view and read the content in the different sections of the page. Each person should have a cascading section with corresponding pictures when clicked on while any other open section is closed in turn. The slideshow is designed so that the page isn't cluttered with information. Blurbs should be accompanied with a timed slideshow of pictures that first shows the person of interest followed by other related images. Blurbs should be concise and contain no more than a few paragraphs.



Draft layout of About page

### Hours and Locations Page

The hours and location page will contain a text block of the operational hours and physical address of the restaurant. It will also contain an interactive map that shows the restaurant location. The hours will be organized by weekday lunch and dinner hours, and weekend lunch and dinner hours. The physical address of the restaurant will feature a link that will open the Google Maps application. Parking information will be displayed on the interactive map, which will contain parking icons. The map will utilize a Google Map application to show a pin of the location.



Draft layout of Hours and Locations page displaying a Google map with a location

### Online Takeout Ordering Page

The online takeout ordering page will provide the functionality of ordering takeout online. It will contain input elements such as name, phone number, a menu that allows quantity and additional options (i.e. extra sauces, suggested sides, etc), and a field to allow a customer to add special requests, and an element that will take in the customer’s credit card information. Output elements will be incorporated as well. Features such as a cart that will hold the order, a breakdown of the pricing, time completion estimate, and if the order can be filled at the time (i.e. a message will be displayed if the restaurant is not open) will be present as well.

Once the user fills out all of the required fields and fills out fields using the specified formats, the user can click on the Submit button to place an online takeout order. If the user does not complete the form in the expected manner (e.g., omitting a value for a required field, not entering all digits of the credit card number), errors will be displayed on the page that informs the user of what they have missed or entered incorrectly so they will clearly know which fields need to be corrected. The online takeout ordering form can be completed without requiring a customer to login.

# Developer Discussion

The “Developer Discussion” section describes the functionality of the website pages in technical detail, non-functional requirements of the website, and decisions related to the software development process which the developers plan to use when developing the website.

## Website Pages

The following subsections include a description of the website requirements from a developer perspective.

### Main Page

Menus will be displayed in the center of the main page in several columns that will responsively adjust themselves according to the screen size. Menu items should be categorized with the price on the right side and excerpts of flavors/ingredients. There should be a text footer on all pages that displays the logo, location and hours. The navigation bar for the menu displayed at the top will be implemented using Bootstrap to have a consistent look-and-feel with the rest of the webpage. The navigation bar will contain the links for each of the pages. There will also be a sub-navigation bar in the menu section that will sort both lunch and dinner menus and then be able to filter out options according to allergies or dietary preferences. There will be several filters for common allergies/dietary preferences such as gluten, shellfish, and vegan.

### About Page

The about page will be implemented using an open-source JQuery plugin that will provide the functionality for the picture slideshow containing the chefs/owners with additional biographical information of each person. There will be span buttons with each person’s name that will send the page down to the corresponding text excerpts. Collapsible dividers could be used to provide the functionality for hiding and displaying sections on the page to reduce the amount of information visible on the page at one time.

### Hours and Locations Page

The text block containing the operational hours on the hours and location page will be organized by weekday lunch and dinner hours, and weekend lunch and dinner hours. An HTML heading tag will be applied to each heading to alter the heading style and different font colors will be applied to the headings for the hours using CSS. For example, the “Lunch” heading can have a heading tag of <h2/> and a CSS class associated with the heading can contain a specific color value for the color property associated with the text.

The interactive map displaying the restaurant location will be presented using the Google Maps API (<https://developers.google.com/maps/documentation/>). The API requires the use of an API key that can be generated from the Google Developers Console ([https://console.developers.google.com](https://console.developers.google.com/)). Usage of the API is free as long as the number of API requests is below the number as determined by Google, based on the type of map that is used. Additionally, the map should have the ability to open the Google Map application on mobile devices.

The Google Map API has functionality to add in custom legends which will be used to mark the parking areas nearest to the restaurant. As parking will not be handled by the restaurant but rather a non-affiliated organization, there will be no need to go into detail of parking availability or pricing.

### Online Takeout Ordering Page

The online takeout ordering page will include a form to be filled in by the user. Validation on the form can be performed using JQuery. For example, if a user does not fill out a required field, a validation error will be triggered within the code that will be displayed on the page in a noticeable way so that the user will be able to correct the errors before re-submitting the form. The online cart that will capture the items the customer selects for takeout could be implemented using an open-source JQuery plugin.

Once the user clicks on the Submit button and no validation errors occur, the data in the form is submitted to an endpoint using a POST request where the data will be processed, logged, and then saved to a server. A response will be displayed to the user based on the result of the form submission processing at the endpoint which will include a receipt. User authentication will not be implemented for online ordering during this phase of development, but may be implemented as an additional feature.

## Non-Functional Requirements

The non-functional requirements of the website describe attributes of the website that are not related to specific functionalities or behavior. Examples of non-functional requirements include performance, availability, usability, security, and maintainability. Performance indicates how well the website responds, availability is a measure of the time during which the website can be accessed and is functional, usability describes how easy it is for a user to learn how to use the website, security reflects how difficult it is for an unauthorized user to be able to intercept or modify data that is used as part of the website, and maintainability is a quality associated with the website code and its life beyond the project timeline. The developers will also keep in mind the non-functional requirements while determining the best way to address the functional requirements in order to provide a quality website for the customer.

The website should be responsive and handle multiple users at a time, without resulting in a lag in response time. Users should be able to access the website during all hours, except for scheduled times when the website is undergoing maintenance. The website should use elements like labels and alternative text for images that are designed to provide an interface that is accessible to all users. The form present on the online takeout ordering page should be developed in a way that will validate form input to reduce security vulnerabilities, such as SQL injection. The code should be written in a way that is easy to understand, reuse, and modify by other developers. The use of maintainable code reduces the amount of time developers will need to spend refactoring code to fix issues or add new features in the long-run.

## Software Development

### Technologies

Open-source technologies will be used to develop the website. HTML and CSS will be used for the front-end technologies. Bootstrap 4 (<https://getbootstrap.com/>) will be used for the design layout, Font Awesome (<https://fontawesome.com/>) will be used for the icons, Google Fonts (<https://fonts.google.com>/) will be used for more appealing fonts and JQuery (<https://jquery.com/>) will be used to provide additional dynamic functionality to the website. Minimized/compressed versions of these resources will be accessed using content delivery networks (CDNs) to improve website performance.

### Platform

The website should be able to be viewed on the following supported web browsers: Google Chrome, Explorer, Firefox, and Safari. Any operating system and any tablet and mobile device should be able to open the website. The variety of platforms supported for the website will increase the number of potential customers who can access the website.

### Software Tools

Git (<https://git-scm.com/>) will be used as the version control system to store the code used for developing the website and as a way for the developers to collaborate. Git is a distributed version control system that allows developers to work on code locally and then commit and check-in their code once it is ready for integration into the master code branch. Developers can pull the latest available code in the master code branch into their local clone of the code. Developers must resolve conflicts between what is available in the master code branch and their local repository so that the code available in the repository is consistent.

GitHub (<https://github.com/>) will be used as a free Git service provider for hosting a private version repository of the code, providing a user-friendly, interactive website for the developers to be able to manage the code. GitHub provides features for viewing different versions of the same file, merging code changes, and tracking issues.

The Google Map API will be used to add location functionality to the site. The API allows the development team to utilize the Google Maps app to set a marker to show the location to end users. The API will also be utilized to set additional markers or *legends* to mark nearby parking areas.

Amazon Web Services (AWS) will be used to host live iterations of the website. AWS uses a pay as you go plan with a free tier for small applications. For this configuration, this hosting service is the best option for the site as it launches because it will likely have a small initial user base. AWS allows for elastic federated hosting services, allowing the site to be publicly available and scalable for growth.

### Software Development Process

An agile software development process will be used so the end user may utilize the system as early as possible to extract value from it immediately. There will be continuous communication between the customer and developers so as new requirements are discovered they may be prioritized in development. Therefore, refinement of the system will happen in iterations as new requirements are discovered. This process is well-suited for the scope involved for the development of the website and the small development team, which will be able to adapt to changes in requirements during the project duration. Furthermore, the design and coding activities of the software development process will be critical and integrate the analysis and testing tasks throughout the iterations.

### Deployment

The website will be developed and tested on the developers’ computers. A possible deployment setup could involve setting up a t2.micro virtual instance on AWS Elastic Cloud Compute (EC2) (<https://aws.amazon.com/ec2/>) that can be used for hosting the site.

A development version of the site could be accessed at a publicly available URL on AWS so that customers can view and test the website as new features are developed. The AWS Free Tier provides free usage of a t2.micro Linux or Windows virtual instance for 750 hours per month for up to one year. So long as an AWS account that is used is eligible for the Free Tier services, there should be no cost for using this technology. The website should be optimized to reduce the overhead so that during times of high traffic, the site will not go over the free tier instance limits.

# Cost

The cost for the restaurant website will include the cost of the analysis, design, coding, and testing tasks as well as additional costs due to unforeseen circumstances. This section describes the tasks associated with the cost and the calculations and decisions that were used to estimate the total number of hours and cost for the website.

## Analysis

The analysis will include gathering and documenting requirements from users, discussing the system requirements among the developers to obtain enough technical detail so that the website can be implemented, deciding upon the technologies, tools, and methodologies to be used to develop the website, the costs, and the timeline.

## Design

The design will involve using the information gathered from analysis during the user and developer discussions to create a potential layout and/or iteration of the website that can be used to visualize the website before it is actually implemented. This task will involve the most hours because the design forms the foundation of the website and needs to be carefully created to meet user requirements and ensure a stable product in the long-run.

## Coding

Coding will involve implementing the pages of the website using technologies decided by the developers. The implementation involves converting the design into markup and scripting languages that can dynamically respond to user input. A website framework will be used to provide a foundation for the layout. Stylesheets will be used to define and customize aspects of the design and layout such as font family, font size, and color.

## Testing

Testing will involve performing usability testing on the different supported browsers, such as Google Chrome, Explorer, Firefox, and Safari. Load impact testing will also be used to ensure stability of the site during high traffic hours. If the data shows that traffic is higher than predicted, there may be additional costs relating to server, database, and or Google Map API service usage.

## Cost Calculations

The cost will be based on $150 per hour. The cost per hour will be multiplied by the number of total hours spent by developers on the project to obtain the cost based on time. Twenty percent of the cost based on time will be added as a cost buffer due to variability in the actual amount of time that may be spent on the website. The total cost is thus calculated using the following formula:

Total Cost = $150 x H + 0.2 ($150 x H) = 1.20 ($150 x H), where H = total number of hours

The table below provides a cost breakdown according to each task (analysis, design, coding, and testing). The total cost for the website is $100,080, which will require 556 hours.

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Number of Hours** | **Percentage of Hours** | **Cost** |
| **Analysis** | 135 | 24% | $20,250 |
| **Design** | 175 | 32% | $26,250 |
| **Coding** | 146 | 26% | $21,900 |
| **Testing** | 100 | 18% | $15,000 |
| **Task Total** | 556 | 100% | $83,400 |
| **Additional Cost** |  |  | $16,680 |
| **Grand Total** |  |  | **$100,080** |

There should be no cost for the technology used in development because open-source technologies will be used for the website and free plans and options will be used for version control and website deployment. However, to pad for potential growth or any unforeseen expenses, 20% of the total cost of development will be added to the budget as *additional costs*.

# System Development Timeline

The website will be completed within 15 weeks as decided by the customer. The timeline below provides detail regarding which components of the website will be completed at five, ten, and fifteen weeks.

## 5 weeks

The analysis for the website should be complete and result in this analysis document. End users should be interviewed to see what sort of features and functions the site must contain. This first stage should also be used as a time to research technologies that may be used in the development of the website.

An initial layout and design for the website will also be developed. End users would also need to see this early design layout so if there are any requirements that could not be thought of initially, they may be addressed here. The initial layout will include the four pages with a background and navigation bar and any other functionality that can be implemented at this early stage.

## 10 weeks

A basic iteration of the website should be implemented that contains the features mentioned on all of the pages. All functional features of the website should be implemented at this stage. At this stage however, the features are still to be developed and worked on.

The main page, about page, and hours and locations page will contain all of the features, but the online takeout ordering page will still be in progress as it is the most complex portion of the website. Final design and format choices will still be in development. Any new requirements discovered by the customers or end users will be addressed by the development team.

## 15 weeks

The website should be complete and include the functionality presented in the “User Discussion” and “Developer Discussion” sections for the four pages of the website. Final design and format choices should be locked in at this point. Errors that are discovered in the second phase should be addressed and fixed. Finally, any additional changes to user requirements should be addressed but may potentially be held off for the maintenance phase.

## Additional Features beyond Timeline

If there were two more months added to the timeline, the developers would implement additional features that would improve the user experience and potentially increase the number of potential customers visiting the site. The additional features under consideration for inclusion in the website are described in detail below.

### Admin Page

An admin page on the website could be developed to allow restaurant staff members with admin access to the website to update the menu, pictures on the website, and store hours without relying on the development team to update content that may change often on the website. The admin section would be role-restricted and require authentication using a username and password known to the restaurant staff members and could be accessed via a link that would not be displayed in a conspicuous location.

Once logged in, users would be able to view the different sections where content could be edited. A user-friendly interface could be presented that would enable a menu item and store hours to be edited directly by replacing the existing text and associated details with updated information. New pictures on the website could be uploaded and staff members would be able to select from available pictures to replace existing pictures on the menu or on the chefs/owners slideshow. A preview page could also be generated to allow for the changes to be viewed prior to actual publishing of the changes for public viewing.

There would also need to be a feature that would allow the owner of the site or another managerial employee to remove or block certain menu items in case the restaurant runs out or cannot fulfill the order. This functionality would be handled mostly on the backend of development but there would be a need to have it be shown on the front end. This feature would have to show that the items in question are normally available but have an additional message that would show that the items are temporarily unavailable. Development of the admin page will result in additional development time and cost, but will reduce future development time and cost related to content maintenance.

### Account Creation

The implementation of a page where returning customers could create an account would allow customers to view the history of past orders placed and receive a customized web experience based on preferences (e.g., sign-up for email notifications when restaurant specials or new menu items are available). Users would be required to enter a username that has not already been used by another customer and create a password matching a certain length and character requirement. Users would be able to enter an email address where they would be able to receive notifications related to account changes, orders, and restaurant news and specials. Functionality could also exist for customers to be able to reset a password. Cookies could be used to track the users and remember their previous orders.

### Google Analytics

Google Analytics could be added to the website in order to track visits to the website and allow the restaurant to be able to better understand the demographics, location, and browser preferences of potential customers. The website content and design can be updated to target specific demographics based on the analytics data collected.

The data could also be used to identify the areas of the website where visitors spend the most time, which could indicate that users either find the content highly relevant or find the section of the website difficult to use. Browser preferences could help developers focus their testing efforts on the browsers that are used most frequently.

### Reservation Page

A reservation page could be added that would allow end users to check reservation availability via a calendar and database. The user would be able to choose the number of guests in a booking and browse the time and date calendar to see if there is any availability. The page would need to take in user credentials and update a database to ensure there are no double booking conflicts. It would also need to have functionality to send end users a confirmation after placing a booking. Potentially, this page could be handled by a third party API such as *Resy or OpenTable.*

1. The main, about, and online takeout ordering pages are three of the four pages to be included in the website. The menu page was later decided to be incorporated as part of the main page during initial requirements gathering and analysis. A fourth page, the hours and location page was later included as part of the scope for the website. [↑](#footnote-ref-0)