Technical Specification Document

Careaway Treatment Planner



**Date**: 12/7/2017

**Team Name**: Team Tangent

Eugene Messina (014455816) Meeting Manager (Team Lead)

Felix Huang (013427386) Documentation Manager

Jimmy Chao (012677182) Accountability Manager

Taylor Tobin (011871260) Technology/Information Manager

Crystal Chun (012680952) Quality Manager

Tyler Dao (013343302) FDA Manager

**Table of Contents**

[**Introduction**](#_3tuxqrg79joq) **5**

[Scope](#_g7fl0lm9l2f8) 5

[Stack Description](#_iy4k9zao75n1) 6

[Client-Side Technology](#_da0oktnt0mb9) 6

[Languages](#_xb9hzrrctqlo) 7

[Team Operating System](#_9s83w6t85eyz) 7

[**Requirements**](#_eez2rm66neb) **8**

[System Requirements](#_fewxqd6toe29) 8

[User Requirements](#_7y9r7boigvlb) 8

[**Development Technologies**](#_g137mbd9aj5m) **9**

[Text Editors](#_tabgui979vw9) 9

[Visual Studio Code (Primary)](#_ptvv2abrxlet) 9

[Alternatives](#_nhhvtcmya74g) 9

[Atom](#_hplzzi649lyi) 9

[Graphic Illustration Tools](#_7izmnkeeaxch) 11

[Adobe Illustrator (Primary)](#_m2ak6fcpazfb) 11

[Alternatives](#_xi433lev9ip1) 11

[Adobe Photoshop](#_vz6167qgdkmv) 11

[**Collaboration Technology**](#_agucaxbeo88x) **13**

[Code Repositories](#_sd26ixbvp970) 13

[GitHub Desktop (Primary)](#_o04lnvh7qm9n) 13

[Alternatives](#_gc1ccknb4iuv) 13

[BitBucket](#_uozpnrtryq9p) 13

[Dropbox](#_59wwb3o3nd0i) 14

[Project Management Tools](#_ute0wlfu9l5l) 16

[Axosoft (Primary)](#_jrd4w122skkb) 16

[Alternatives](#_pashaqmlos5f) 16

[Trello](#_8vq6x3i60oc9) 16

[Communication](#_rvastwt92gfd) 18

[Discord (Primary)](#_vbanw1adcph7) 18

[Alternatives](#_xgfbax5iyo8v) 18

[Google Hangouts](#_nu4blkud5ki8) 18

[LINE](#_f0j6w3duwgq0) 19

[**Documentation Technology**](#_y9q07ey50gf5) **20**

[Document Creation](#_wzmfvm69927i) 20

[Google Drive (Primary)](#_twrajh8qoxl7) 20

[Alternatives](#_jt02zqy4t2k0) 21

[Microsoft Office 365](#_pnooe9hffv8e) 21

[Diagram Tools](#_o76wjftgklj6) 22

[Draw.io (Primary)](#_fjzlz8f4tenc) 22

[Alternatives](#_s0730pn09yza) 22

[Lucidchart](#_1hzaq52mhirc) 22

[**Deployment Technology**](#_61yehmauwzv6) **24**

[Hosting Service](#_a2wm9kll7d5b) 24

[Amazon Web Service (Primary)](#_bg6zgwpw7axt) 24

[Alternatives:](#_hxtdua6zwrui) 24

[Google Cloud Service (Google Application Engine)](#_8jou8jqh9vcg) 24

[Proxy](#_p7fgaikr4gqn) 25

[Nginx (Primary)](#_cbom3uxhvm7d) 25

[Alternative](#_jbiaaf3dmg3) 26

[mod\_proxy](#_u8c0mitm0wxc) 26

[Operating System](#_ftt64p7eirsr) 26

[Linux Ubuntu (Primary)](#_jldyouf5c61r) 26

[Alternative:](#_mfjg2g6rao70) 27

[Amazon Linux](#_krwg73iwfl4f) 27

[Domain Naming Service](#_69lp5a7y4cmz) 28

[NameCheap for Education (Primary)](#_glqvwtrvgjuh) 28

[Alternatives](#_5kwaarr77l7v) 28

[Domain.com](#_r5wekvuukqjt) 28

[**Protocols**](#_9ngw64tkbopu) **29**

[HTTP (Primary For Testing Purposes)](#_s1m169wue1yy) 29

[HTTPS (For Deployment Purposes)](#_d5e4lfjnw0g7) 29

[SSL](#_y65qi1j1vo6m) 30

[TLS](#_z73lunso00ty) 30

[**Frontend Technology**](#_5u1o9hz89a3u) **31**

[CSS/Styling Frameworks](#_l2266emk2cva) 31

[BULMA (Primary Choice)](#_mxuhcg4h952f) 31

[Syntactically Awesome Stylesheets (SASS/SCSS) (included in Bulma)](#_cbgi454e2hpy) 31

[Alternative(s)](#_171gk5bhlciw) 32

[Bootstrap](#_en78motqawf1) 32

[Cascading Style Sheets (CSS)](#_j044iwcncaig) 32

[Foundation](#_9yvesnvy1m5i) 32

[Semantic UI](#_i66k22d5kxp7) 33

[vue-foundation](#_t46emtilz62c) 33

[Semantic-UI-Vue](#_cuxa8gs77gni) 33

[Javascript Frameworks](#_p7cg602tzf8v) 34

[VueJS (Primary)](#_424c5vwar3cr) 34

[Alternative(s)](#_8s97wva6gruv) 34

[EmberJS](#_3vrf1rufp2yi) 34

[Angular](#_m1v1ddkgulte) 35

[Discussion for Primary Choice](#_8woj1mvqs8xu) 35

[Plugins (Calendar):](#_rp06cn9nm1pg) 36

[Vue-simple-calendar(Primary)](#_gp04x0qashqd) 36

[Vue-fullcalendar](#_81k4kmktcfzz) 36

[FullCalendar.IO](#_9a4ex35rb1qc) 36

[Discussion for Primary Choice](#_siv9eexj5gle) 37

[Plugins (Charts):](#_gpm05lptjyur) 37

[Vue-chartjs (Primary)](#_t3yzsl5ixpei) 37

[Alternative(s)](#_klouwvgi41zt) 38

[Chart.js](#_wy1kx2wpll31) 38

[Plugins (Drag & Drop):](#_69ty7si1hmiq) 38

[Vue.Draggable(Primary)](#_xgct72onjvil) 38

[Alternative(s)](#_yid5joofph9a) 39

[Vue-drag-drop](#_8sx61qvc418z) 39

[Discussion for Primary Choice](#_5aqtq1kpc0zh) 39

[**Backend Technology**](#_n1kvaxg2298n) **40**

[Runtime Engine](#_30m4vdpg5d0) 40

[Node.js (Primary)](#_irawe4fpxrpu) 40

[Alternative](#_qc604n16gotr) 41

[Nashorn](#_o03zonug0jhm) 41

[Package Manager](#_54nft7xfyt80) 41

[Npm (Primary Choice)](#_t1by9i7g6cg3) 41

[Alternative](#_xzdnx5qliog1) 42

[Bower](#_j7vezz3ic2cc) 42

[Plugins](#_nxj1rs6ka5m0) 43

[Express (Primary)](#_w8lhyxco6c3z) 43

[HTTP (Primary for Development)](#_uu44rniqbyv2) 43

[HTTPS (Primary for Deployment)](#_osi0yvnft3zs) 43

[MongoDB (Primary)](#_q3dite50fnt) 44

[Alternative](#_1q9tfg81f89q) 44

[MongooseJS](#_p1gd9f78dnwf) 44

[Path (Primary)](#_4mzo1oyj31rl) 45

[Express-session (Primary)](#_cl1ni4w9oz5s) 45

[Alternative](#_thdi8e75sa1s) 45

[Cookie-session](#_9952ranruvre) 45

[Body-parser (Primary)](#_ezn9v5ukv2dj) 46

[Alternative](#_a0rb6uvvauel) 46

[body](#_b0eo6b3slo2v) 46

[Passport (Primary)](#_qjsgywsisxp2) 47

[Crypto-js (Primary)](#_oedbtj9jzw28) 47

[Alternative](#_oqwu0da5vw26) 47

[Bcrypt](#_sludm5nzlt9q) 47

[CSURF (Primary)](#_4gakaqm3buc) 48

[Helmet (Primary)](#_ke7mhw9bxsob) 48

[Jasmine (Primary)](#_xujr0xekza2q) 49

[Randomstring](#_ufrwoab1dr2o) 49

[Forever](#_4jefxzwgiyuf) 49

[Database](#_ik5142i380rn) 50

[MongoDB (Primary)](#_wbuf0vegx4oj) 50

[Alternative](#_1svbo9ietp2e) 50

[Couchbase](#_uxed1ge43key) 50

[HTTPS Certificate](#_id0pw9ia4u3r) 51

[Certbot (Primary)](#_9cg6sfdqyl55) 51

[Encryption](#_iah5sequad5c) 51

[**Testing Technology**](#_5937bwm4wsm1) **51**

[Google Chrome and Chrome DevTool (Primary)](#_kiv75ex9gyoj) 51

[Mozilla FireFox and Page Inspector](#_8h07kn3918m) 52

[Apple Safari and Web Inspector](#_fzobxmgi5zrd) 52

[Node.JS and Node-Inspector(Primary)](#_bscgllrnzo49) 53

[Jasmine (Primary)](#_k5s4idvrgx2f) 53

[Alternative](#_xt6g5u6lf11t) 54

[protocol](#_k5s4idvrgx2f) 54

[**Glossary**](#_r1rby9ikcof) **54**

# 

# 

# **Introduction**

## **Scope**

The Technical Specification Document will describe the technologies that will be used in the implementation and production of this project. Any software that the team uses for the development of the project will be listed on here with technology name, version number, reason of use, and license if applicable. These technologies includes documentation, software development, or communication within the team. Alongside with those technologies, the software development stack in which the web app will be deployed, and developed with is also listed and described in this document. All these technologies are based off of the needs of the development team to finish the twelve business requirements included in the Business Requirements Documents.

The purpose of this document is to outline the technologies that will be used to implement the Careaway Treatment Planner, our reasons for using them, and the alternative technologies we considered. This document will focus on the following technologies:

**Development Technology**: Technologies used for writing and saving code written for the single page application.

**Collaboration Technology**: Technologies used for communicating, assigning tasks, organizing team meetings, logging of team activities, and file sharing.

**Documentation Technology**: Technologies used for collaboratively writing and sharing the core artifacts - Business Requirements Document, Project Plan, Technical Design Document, Design Document, and Test Plan - for the project.

**Deployment Technology**: Technologies used to deploy our technology including the operating system, server host, proxy, the network protocols, and the domain naming service used for our system.

**Front End Technology**: Technologies used to create the client and the user interface on the browsers that the CareAway Treatment Planner will support.

**Back End Technology**: Technologies used to save data from the client, and provide core functionality to our system.

**Testing Technology**: Technologies used to test and debug each component in our system.

This also includes testing deployment and user experience.

In each category we will specify our primary technology that we will utilize for the Careaway Treatment Planner and any alternatives to those technology. Adobe Illustrator was previously acquired. However, all remaining chosen technologies will be *freeware*, software that is available for use at no monetary cost, MIT licensed software, and services that are free for a limited time to adhere to the budget as specified in the Project Plan. Furthermore, any updates for any of the technologies our team has agreed upon using will first be reviewed to see if it will have any major impact to our application. If not then the team will proceed with the upgrade and will reflect it on this document.

### **Stack Description**

Our team’s development stack is a modified MEAN stack which allows us to use one language for both server-side and client-side environments. Our customized stack consists of the following:

**MongoDB**: a NoSQL database

**Express**: the middle-ware between the front-end and the back-end

**Node.js**: our back-end server

**Vue.js**: the front-end framework

Our team has decided on MongoDB as it satisfies our application’s need to store and retrieve data while complying with HIPAA regulations. In addition our team will be using the Express.js as a backend server framework due to its compatibility with Node.js. Our team has decided on Node.js as it allows our team to code in one language which creates consistency and readability for our team. For our front end framework our team has decided on Vue.js due to its ability to create modular components for our system to allow our team to create separate non-dependent components. Lastly, our team has decided on Linux as our host operating system for our application due to the flexibility it provides and our team familiarity with the operating system. In addition, our team has decided to utilize this stack due to our groups familiarity with many of the components in the stack. In addition, this stack allows our single page application to be cross-platform over operating systems such as macOS, Windows, and Linux.

### **Client-Side Technology**

For the single page web application we are supporting these browsers for the minimum viable product:

* Google Chrome - Version: 62.0.3202.94 and newer
* Mozilla Firefox - Version: 57.0 and newer
* Apple Safari - Version 11.0.1 and newer

In addition, we are limiting the CareAway Treatment Planner to these web browsers due to their compatibility with our customized stack.

### **Languages**

Our team will be utilizing, in part or in entirety, the following web-development languages:

* HTML5 (HyperText MarkUp Language): The standard markup language to create web pages. HTML5 is the latest standard to markup language thus far and our team will utilize this markup language to create a web application
* CSS (Cascading Style Sheets): Defines the presentation of the markup language. Our team will be using many CSS frameworks that will be defined in the front-end section of this document
* TypeScript (Version 2.6.1): An advanced Javascript language that is translated into Javascript.
* Javascript 1.8.5 (ECMAscript 5): An object based programming language to create functionality and business logic within our system.

### **Team Operating System**

Our team will be using the following Operating Systems to develop and test the CareAway Treatment Planner web application:

* MacOS Sierra Version 10.12.6 or later: The majority of our team consists of Macbook users utilizing the most current version of Sierra. Our team agreed to use MacOS Sierra as it is the standard Operating System for Macbook at this given time.
* Windows 10 Build 16299.64 or later: Other members of our team will be using Windows 10 as their working operating system. The members of our team using Windows will be using the latest version of the Windows 10 operating system.
* Linux Ubuntu Version 16.04 LTS OS: Our team is utilizing Linux Ubuntu in order to deploy and test the server in its home environment. Further detail regarding Linux Ubuntu is under the Deployment Technology section. In addition, our team will be testing the CareAway Web Application under this Operating System environment. In addition, we will be using this specific version of Linux due to our familiarity with the system.

# **Requirements**

Our team has decided on these CORE and Content requirements for the minimum viable product and have provided a high level introduction of how the system requirements integrate with the user requirements.

The following are the **CORE Requirements**:Registration, Security, Authentication / Session Termination, User Access Control, Data Access, Error Handling, User Management, Network Communication

The following are the **Content Requirements**:Treatment Plan Creation/Revision, Interacting with Treatment Plan, Appointment Scheduling, Data Analysis Reports

## **System Requirements**

|  |  |  |
| --- | --- | --- |
|  | **System Requirements** | **Feature Reference** |
| #1 | Database to hold data about patients, medical professional, treatment data, and appointments | UR#1, #2, #3, #4, #5, #7 |
| #2 | 3rd party service to generate data analysis reports | UR#5 |
| #3 | Service to host interactive calendar | UR#3, #4 |
| #4 | User Session Creation | UR#2 |
| #5 | Error Handling | UR#1, #2, #6 |

## **User Requirements**

|  |  |  |
| --- | --- | --- |
|  | **User Requirements** | **Feature Reference** |
| #1 | Registration | SR#1, #5 |
| #2 | Login/Logout | SR#1, #4, #5 |
| #3 | Creating/Revising Treatment Plans | SR#1, #3 |
| #4 | Appointment Scheduling | SR#1, #3 |
| #5 | Generating Data Analysis Reports | SR#1, #2 |
| #6 | Sending Breach Notification Reports | SR#5 |
| #7 | Viewing Patients | SR#1 |

# **Development Technologies**

## Text Editors

### **Visual Studio Code (Primary)**

**Version: 1.18.1**

**Description:**

Visual Studio Code is a text editor optimized for building and debugging web and cloud applications. Visual Studio Code is available on the major three operating system (Windows 10, macOS, and Linux). It’s lightweight and portable, making it very versatile as a text editor.

**Reason:**

The development team will be using Visual Studio Code as our primary choice for coding and text editing. The fact that Visual Studio Code is available for both Windows 10 and macOS is very important as it accommodates the hardware for the entire team. Furthermore, the team likes that the specified version of Visual Studio Code contains the IntelliSense feature which provides intelligent code completion based on language semantics and analysis of the code. Visual Studio Code also allows the installation of extensions which mean that we can install add-on software to support tools for development. This includes linters and coding standard checkers to keep consistent code. Lastly, Visual Studio Code supports integration with Git for code collaboration. This makes pushing code and cloning repositories easy to do.

**Source:** [**https://code.visualstudio.com/**](https://code.visualstudio.com/)

**Cost:** Free Software (No Cost)

**License:** MIT License/Freeware

### Alternatives

#### **Atom**

**Version: 1.22.1**

**Description:**

An open source, customizable text editor to assist in team development of HTML, CSS, JavaScript, Node.JS and many other languages.

**Reason:**

Atom is an alternative to Visual Studio Code for coding and text editing. Atom supports many of the features that Visual Studio Code does such as Git integration and smart suggestions and autocompletion. Additional features can be added to Atom as well through packages. The Teletype package (described above) allows multiple users to collaborate and work on the same code simultaneously. This feature would be ideal for any collaborative project. However, we are detracting from Atom because of the many issues that users have faced when using the program. As described above, many users reported unexpected crashes and the IDE being slow to respond at times. Our team cannot afford to have these issues surface during the implementation phase. Furthermore, although the Teletype feature would be useful for our project, the feature is still currently in beta and our team does not feel comfortable depending on software that is not fully developed.

**Source:** [**https://atom.io/**](https://atom.io/)

**Cost:** Free Software (No Cost)

**License:** MIT License

**Discussion For Primary Choice**

Above, we have listed Visual Studio Code as our primary choice for text editing and coding. The fact that Visual Studio Code is compatible on macOS and Windows is a huge selling point for us. This makes it so that all members of our team will be able to contribute during the implementation phase, regardless of their operating system. Secondly, since we are using GitHub Desktop (specified below in the next section) for code sharing and collaboration, the fact that Visual Studio Code supports integration with Git is a very helpful feature that our team can utilize. We can push and commit code directly from Visual Studio Code which makes our workflow more efficient.

We have listed Atom as an alternative to our primary choice. Like Visual Studio Code, Atom allows the user to add additional functionality to the editor in the form of packages. Furthermore, since Atom was developed by GitHub, it supports integration with Git just like Visual Studio Code. One feature that we really like about Atom is the new Teletype feature that was just recently released. The Teletype package allows a user to share their tab with other developers, and they will be able to see and edit code on the tab in real-time. This feature could be useful for a collaborative project such as this one.

Despite the abundance of similar features to Visual Studio Code that Atom can provide, we are still sticking with Visual Studio Code as our primary choice. One reason for this is because from our research, many Atom users have reported bugs and crashes while coding and installing packages, and in turn many of those users have turned to Visual Studio Code for their work. Many Atom users have also reported the IDE being slow and unresponsive at times, which hindered them from doing their work. Because Atom has this history of issues, our team is going with Visual Studio Code.

## **Graphic Illustration Tools**

### **Adobe Illustrator (Primary)**

**Version:** CC 2017 (21.0.1)

**Description:**

A vector graphic editor that provides tools to create logos, icons, illustrations for web, videos, and mobile.

**Reason:**

Our team will be also using Adobe Illustrator CC to create assets and designs for the CareAway Treatment Planner. Adobe Illustrator is compatible with macOS and Windows 10. Designs created in Adobe Illustrator can be easily exported and shared with other team members. Our team plans to use Adobe Illustrator to create the wireframe and mockup UI designs for the Design Document.

**Source:** [www.adobe.com/Illustrator‎](http://www.adobe.com/Illustrator%E2%80%8E)

**Cost:** Normally $19.99 a month (*Free* because it was installed on one team member’s machine before the start of the project)

**License:** Adobe Proprietary Software/Adobe General Legal Term of Use

### **Alternatives**

#### **Adobe Photoshop**

**Version:** CS6 13.0

**Description:** Adobe Photoshop is a graphics editor for multiple image formats. Photoshop is used to enhance photographs, and create UI designs, banner ads, and video graphics. Photoshop creates Raster or bitmap images which is ideal for printed images that will be displayed as photographs or billboards.

**Reason:** Our team has listed Adobe Photoshop as an alternative for Adobe Illustrator because it can accomplish many of the things that Illustrator can do such as editing images and creating UI designs. However, it remains an alternative because we feel that Adobe Illustrator is more in line with what the team needs.

**Source:** <https://helpx.adobe.com/x-productkb/policy-pricing/cs6-product-downloads.html>

**Cost:** Normally$39(*Free* because it was installed on one team member’s machine before the start of the project)

**License:** Adobe Proprietary Software / Adobe General Legal Term of Use

**Discussion For Primary Choice**

Above we have listed Adobe Illustrator as our primary choice to graphic illustration. One reason that we have commited to Adobe Illustrator is the software’s level of familiarity with our team. 2 members of our team have worked extensively with Adobe Illustrator for graphic design.

Below we have listed Adobe Photoshop as an alternative for our primary choice. Photoshop and Illustrator are similar in that both allow the user to edit graphics and images. However, after doing research of our own, our team felt that Illustrator fit our needs better. Illustrator was better for creating mock UI from blank designs which is what we needed, while we felt that Photoshop was better for editing existing images. Furthermore, after using Illustrator to design the logo and create initial designs for the UI, we felt that Illustrator was much easier to use.

# 

# 

# **Collaboration Technology**

## **Code Repositories**

### GitHub Desktop (Primary)

**Version:** 1.0.9

**Description:**

GitHub Desktop is a desktop program that can be used for code collaboration, change tracking, and merging changes. GitHub Desktop provides the user with a desktop interface to use GitHub services.

**Reason:**

GitHub Desktop allows our team to pull source code to our computers, make modifications to the system, and test the changes on our own system’s without fully committing the changes to our main code repository. Allowing the creation of *code branches* is very helpful because it allows each member to individually develop or collaborate on different parts of the code without overwriting changes that another member makes. Another reason that we are utilizing GitHub Desktop is because whenever new code is pushed and committed, we can see exactly what changed. This makes it so that we can easily revert back to an old version of our code if there are issues with the new version. Also, by knowing what parts of the system have changed, the QA Manager can determine which part of the system needs to be tested. GitHub Desktop is also cross platform on macOS and Windows 10. Our team will be utilizing GitHub Desktop version 1.0.9 as it is the most stable version.

**Source:** [**https://desktop.github.com/**](https://desktop.github.com/)

**Cost:** No Cost

**License:** Freeware/GitHub’s Terms of Service

### **Alternatives**

#### **BitBucket**

**Version**: 5.6

**Description:**

BitBucket is a web based Version Control System used to host source code.

**Reason:**

BitBucket is an alternative to GitHub for hosting and managing source code and related content. BitBucket allows for webhooks, which are used to update users on new functionalities whenever someone pushes to a repository. BitBucket is a good alternative due to its compatibility with a multitude of our deployment technologies, such as Amazon Web Service, and it allows for private repositories. However due to the size of our team using this technology will cost our team money, in addition our team has more experience with GitHub than BitBucket so we decided upon GitHub as our main repository and may alternatively use BitBucket if needed.

**Source:** [**https://bitbucket.org/**](https://bitbucket.org/)

**Cost:** No Cost

**License:** Atlassian Customer Agreement

#### **Dropbox**

**Version**: 39.4.49

**Description:**

A file hosting service that offers cloud storage and file synchronization on the web or on their client software.

**Reason:**

Our team chose DropBox as an alternative to host the source code of the CareAway web application due to its simplicity and easily accessible services throughout each team member’s computer. Dropbox can be easily accessed through a desktop application, just like GitHub Desktop. Dropbox iscross platform and is compatible with macOS High Sierra, Windows 10, and Linux Ubuntu. However, Dropbox does not offer version control and only saves a backup of older file updates for up to 30 days before it gets deleted. Due to this we may alternatively turn to Dropbox to save backups of our software files or use it as a temporary storage service.

**Source:** [**https://www.dropbox.com/**](https://www.dropbox.com/)

**Cost:** No Cost

**License:** Dropbox Terms of Service and Dropbox Business Agreement

**Discussion For Primary Choice**

Above, we have listed GitHub Desktop as our primary choice for code collaboration. Since GitHub Desktop is a desktop application for GitHub, we can easily utilize the Git services that GitHub provides from our own machines. This is one reason that we are in favor of these technologies. As we specified in the Project Plan, each team member will be providing their own machines for development, so GitHub Desktop can be easily installed on each person’s machine. Each member can also quickly upload and pull code to view. Another reason that we have chosen GitHub Desktop as our primary choices is because of its accessibility. All that is needed is a working Internet connection and a valid GitHub account. GitHub also provides students with a Student Developer Pack which allows students to create private repositories.

We have listed some alternatives to our primary choices. BitBucket is similar to GitHub Desktop in that it is a web-based hosting service for source code and other related content. Many of the features that GitHub Desktop offers such as pull requests, commit history, and private repositories can also be found on BitBucket. However, using BitBucket would incur a monetary cost on the team to create a team repository. By utilizing GitHub through GitHub Desktop, we can create a team repository for free and we can avoid any charges entirely. Another alternative that we have identified is Dropbox. Dropbox is a popular file hosting and backup service. However, with Dropbox, older versions of files are automatically deleted after 30 days. With GitHub Desktop, commit changes can be reverted back to the very first file upload.

## 

## 

## **Project Management Tools**

### **Axosoft (Primary)**

**Version:** 17.1.3

**Description:**

Axosoft is a project management web application that provides tools to manage software development in a SCRUM Agile team. Using Axosoft, the team can plan work for sprints, and organize work into epics, stories, and tasks. For each piece of work, the team can specify estimations for how long they think that piece of work will take, and log their work hours for their assigned tasks.

**Reason:**

Our team will utilize the Axosoft because it aids in following a SCRUM Agile style of development. As specified earlier, Axosoft provides our team with a simple, easy way to organize our epics, stories, and tasks in the software development process. Axosoft also provides ways to assign tasks to each individual in our team. Axosoft can also keep track of our team’s estimation for each task and track the amount of work each member has done for each task. This allows our team to calculate a more accurate team velocity in our software development. As we continue to use Axosoft, the software will calculate the work velocity for each individual along with the work velocity for the entire team. Furthermore, Axosoft provides our team with interactive charts and graphs of the amount of activity each member has contributed to the system.

**Source:** [**https://www.axosoft.com/**](https://www.axosoft.com/)

**Cost:** Free Trial - 12 month subscription

**License:** Axosoft Software as a Service Agreement.

### **Alternatives**

#### **Trello**

**Version:** Version 1.0 (Stable)

**Description:**

Trello is a Project Management web application that makes collaboration on a project easy by organizing work that needs to be done into interactive cards.Trello allows the team to add Power-Ups to add functionality to the Trello page. Power-Ups allow integration with popular development tools such as GitHub, Google Drive, and JIRA.

**Reason:**

Trello is our team’s alternative project management tool for task tracking. Trello helps organize these tasks into columns and can label them according to status and the individual performing the task. Our team chose Axosoft over Trello as our main collaboration technology, because Axosoft shows the assignment of task a lot better, and it keeps a log of work completed and estimations. The fact that we can also organize work into sprints on Axosoft and easily see the status of a work item is also a key reason we chose to go with Axosoft.

**Source:** [**https://trello.com/**](https://trello.com/)

**Cost:** No Cost

**License:** Trello Terms of Service

**Discussion for Primary Choice**

Above, we have listed Axosoft as our primary choice for project management. We chose Axosoft because it is made specifically for Agile development which is in line with the SCRUM methodology that our team will be using. Axosoft allows work to be assigned to each member, so this holds people accountable. Another reason that we like Axosoft is that it calculates the work velocity for the entire team. This is helpful so that we can ensure that the entire team is meeting the required work velocity as defined in the Project Plan. Axosoft also has a feature to track the progress of a sprint visually. Axosoft can create fully interactive kanban boards and the team can see what tasks are in-progress, blocked, and completed. Our team thinks that this feature would be useful during Stand-Up meetings with the client during the implementation phase so that we can easily show the progress of our work. This feature would also be helpful in making sure that our project stays on schedule.

We have listed Trello as an alternative to our primary choice. Trello allows the team to create cards for work that needs to be done and organize that work into categories. Trello is a tool that has a lot of visual appeal. At a glance, you can see what card is under what category and move each card as work gets done. Each card also allows each member to leave comments, which would be helpful if the team needs to communicate an idea about a specific topic.

However, we are still sticking with Axosoft as our primary choice because our team feels it is better for SCRUM software development. One key feature that we really like about Axosoft is the ability to track work logs for each task. We want to be able to know how accurate our estimations are. Unfortunately, Trello does not offer this feature. In terms of use for project management, our team felt that Trello is much harder to use. In Axosoft, we can add requirements and tasks to be done for each sprint. If work needs to be added to a sprint or needs to be moved to a new sprint, Axosoft can easily accommodate this change. In Trello, we would have to create new categories and drag individual cards, which would be very time consuming.

## **Communication**

### **Discord (Primary)**

**Version:** 0.0.248

**Description:**

Discord is a video, audio calling, and instant messaging communication service that is free for anyone with no sign up required. Discord can be accessed as a desktop client or through a web application.

**Reason:**

Our team decided on Discord for our main method of communication because of how simple the service is to use and its accessibility. Discord can be easily installed on macOS and Windows for free. In the event that the user does not want to download the desktop application, they can easily join in to any group calls through the web application. In addition, Discord can be accessed on mobile (iOS and Android). Discord also allows the team to pin important notifications and upload code snippets. Lastly, Discord allows the team to share screens which may be helpful when the team needs to debug a problem during the implementation phase, and not all members can physically be at the same location.

**Source:** [**https://discordapp.com/**](https://discordapp.com/)

**Cost:** No Cost

**License:** Proprietary Freeware

### **Alternatives**

#### **Google Hangouts**

**Version:** Current Release

**Description:**

A communication service that provides video, voice, or text communication for group or one to one discussions. Utilization of Google Hangouts will require an existing Google account.

**Reason:**

Google Hangouts is readily accessible to every member in our team as it is a free feature for anyone with a Google account. Google Hangouts supports many of the features of Discord such as file transfers, voice and video calling, and text communication. However, users must have a google account in order to access Google Hangouts, while Discord users don’t have to provide any credentials to join or make chat rooms.

**Source:** [**https://hangouts.google.com/**](https://hangouts.google.com/)

**Cost:** No Cost

**License:** Freeware, Google Terms of Service

#### **LINE**

**Version:** Current Release

**Description:**

A communication service that provides one to one or group text messages and instant photo messaging to group members. LINE can be accessed as a desktop application for macOS and Windows 10 and on mobile platforms on iOS, Android, and Windows Phone.

**Reason:**

LINE could be an effective mode of communication for our team due to it mobile platform and easy to use system. Like Hangouts, LINE provides quick instant messaging services for free for our entire team. However the member must still have a connection to the Internet to use LINE. The main drawback with LINE is that it does not provide any file transfer system to share project files we want to share with the group unlike Discord, however you can send pictures of group work, though that is still very limiting.

**Source:** [**https://line.me/en/**](https://line.me/en/)

**Cost:** No Cost

**License:** Freeware**,** LINE Terms and Conditions.

**Discussion for Primary Choice**

Our team has chosen Discord for communication between members of the team. We like Discord because it serves as a multi-purpose communication hub. Discord is also very accessible because users don’t have to create an account to utilize the service. We also like how the Discord can be accessed through the desktop client and the web application. Using Discord, we can send messages to the entire team, host video and audio calls, and share files and images.

Below we have listed alternatives for Discord. The first alternative that we have listed is Google Hangouts. Google Hangouts has many of the features that Discord does such as file transfers, video and audio calling, and messaging. The main drawback for Google Hangouts that deterred our team away from using it was its lack of file support; Hangouts only allows users to transfer PNG, PDF, JPG, and DOCX files. With Discord, we can also share CSS, HTML, and JavaScript files.

Another alternative that we have identified is LINE. LINE is very effective as a messaging and video calling service. However, our team feels that that is the limitation of LINE. However, LINE does not allow file transfer of any sort, which we feel would hinder our progress during the implementation phase.

# **Documentation Technology**

## **Document Creation**

### **Google Drive (Primary)**

**Version**: 3.36.6721.3394

**Description**:

A file storage service that allows users to store and share files with other users.

**Reason**:

Google Drive is a free application we plan to utilize for collaborating on our written deliverables. Google Drive is an attractive option because each member of the team has an existing Google account. We decided to save all the core artifacts of our system onto Google Drive. Storing our written deliverables on the Google Drive ensures that all of our documents are stored in a single place and that all the documents are kept together. This allows every member on our team to have easy access to all existing documentation for the CareAway Treatment Planner. Google Drive also allows any member to upload their own documents. In addition, Google Drive allows us to manage viewer privileges and edit permissions. This keeps the files in our Google Drive secure and prevents any persons without permission from viewing or modifying our work. Also, Google Drive can be used on multiple browsers and has a mobile version for both Android and iOS allowing us to view files even if we are away from our work stations.

Accessing Google Drive also allows the team access to Google Docs and Google Sheets. Google Docs is an online word processor where we can outline and write the written deliverables. Multiple members of the team can access a Google Doc document at the same which, which is effective for collaborating and sharing ideas. Google Sheets is an online spreadsheet creation tool. Google Sheets is used to generate graphs and organize any tables that the team needs to show on our written deliverables. Both Google Docs and Google Sheets allow support for comments where users can highlight specific sections that need to be addressed.

**Source:** <https://drive.google.com/>

**Cost**: No Cost

**License**: Google Terms of Service

### **Alternatives**

#### **Microsoft Office 365**

**Version**: Current Release

**Description**: Microsoft Office365 is a free online application that allows the user to utilize the full Office suite (Access, Word, Excel, Powerpoint, OneNote, Sway, Mail, OneDrive, Calendar, etc.). Office365 can be accessed anywhere there is network connection and through a valid login. Office Online can be used to create documents, spreadsheets, and presentations.

**Reason**:

Microsoft Office365 is a fair alternative to Google Drive. In fact, Office365 supports many of the same features that Google Drive does like simultaneous collaboration and a single storage location. Office365 would also be a viable alternative to Google Drive because as CSULB students, our team has Microsoft Office accounts already setup.

**Source:** https://products.office.com/en-us/office-online/documents-spreadsheets-presentations-office-online#office-online-products

**Cost**: Free

**License**: Microsoft Online Services Acceptable Use Policy, Customer Portal Terms of Use

**Discussion for Primary Choice**

The team has chosen Google Drive as our primary choice for document creation. We like how Google Drive is a multi-purpose hub for document storage, creation, and revision. Another reason that we are choosing Google Drive is because of its familiarity with our team. All members have utilized Google Drive previously for other classwork so we can easily understand the system.

We have listed Microsoft Office365 as an alternative to our primary choice. Office365 is unfamiliar to our team as no members have used this service before. Another big drawback of Office365 is that the many users have often reported the interface becoming sluggish after an extended period of use. We have decided to stick with Google Drive as our primary choice because we want to stick with something that we are familiar with and know works reliably.

## 

## 

## **Diagram Tools**

### **Draw.io (Primary)**

**Version**: Current release

**Description**:

A web-based drawing tool that allows users to create and edit diagrams while collaborating with other users in real-time. Draw.io can be utilized as a standalone service online or through Google Drive, Dropbox, and OneDrive.

**Reason**:

Draw.io allows us to draw various types of diagrams and collaborate online on the same files, similar to Google Docs and Google Sheets. Our team will be utilizing Draw.io for modeling business requirements and the system. Draw.io also integrates with Google Drive, allowing anyone to easily access and edit the diagrams. In addition, Draw.io allows multiple team members to work on each diagram simultaneously while updating and saving the file with every new modification it has received.

**Source:** <https://draw.io/>

**Cost**: No Cost

**License**: draw.io EULA / Terms of Service

### **Alternatives**

#### **Lucidchart**

**Version**: Current release

**Description**:

A web-based visualization tool that allows users to create and edit diagrams.

**Reason**:

Lucidchart has been listed as an alternative to Draw.io because it offers better visualizations and more features to create diagrams for our documentation. Lucidchart also has a better aesthetic appeal than Draw.io where diagrams simply appear flat. However, since Lucidchart has such limited access at the budget for a free account, the team is planning to use Draw.io to satisfy our modeling needs.

**Source:** <https://www.lucidchart.com/>

**Cost**: No Cost

**License**: Lucidchart Terms of Service

**Discussion for Primary Choice**

Above, we have listed Draw.io as our primary choice for creating and editing diagrams. Our team likes Draw.io because of its simplicity of use. To create a diagram, we simply drag a shape from the toolbar on the left and drag it onto the content pane. Connecting each shape is done by simply dragging the arrow from one shape to the next. Another reason we like Draw.io is due to its integration with Google Drive. Our team can create Draw.io diagrams from inside Google Drive, and they will be saved there. This is beneficial because all of our documentation will be stored in a single location.

We have listed Lucidchart as an alternative to our primary choice. Like Draw.io, Lucidchart also integrates with Google Drive. Lucidchart works just as well as Draw.io for diagramming, but our team feels that what we can access on Lucidchart is very limited. Since this project will have a monetary cost of $0 as specified in the Project Plan, a free account on Lucidchart only allows the team to create 3 documents with a limit of 60 objects per document. Our team feels that this limit is too restrictive and that we are unable to model all aspects of our system in only 3 documents. If we wanted to create more than 3 documents, the team would incur a monetary cost.

# **Deployment Technology**

## **Hosting Service**

### **Amazon Web Service (Primary)**

**Product:** EC2 Instance T2.Micro (General Purpose)

**Description:**

AWS is a cloud service that allows a user to deploy Internet-based services.

**Reason:**

Our application will be deployed on the Amazon’s Web Service EC2 because it provides our system with a secure and resizable computing capacity. It also provides control within our system so that any future patches can be quickly deployed. AWS T2.Micro allows communication with Node.js and MongoDB which our team will be using. In addition, the hardware provided is sufficient enough to run our web application using High Frequency Intel Xeon Processors. Furthermore we decided to utilize the standard Amazon Web Service rather than the Amazon Web Service Government as the standard AWS gives our team a 12 month free trial and is secure enough for our web application’s minimum viable product. AWS Government is not necessary in our current scope and is out of our budget. However, for future implementation we may use AWS Government to host our system. The reliability, security, and scalability of Amazon’s web services satisfies the requirements to deploy our system.

**Source:** <https://aws.amazon.com/free/>

**Cost:** 12 month Free Trial

**License:** Amazon’s Software License

### **Alternatives:**

#### **Google Cloud Service (Google Application Engine)**

**Version:** 1.9.51

**Description:**

A cloud computing platform service and a web framework to host internet based services and provides services such as computing, data storage, and data analytics.

**Reason:**

An alternative to AWS could be to use the Google Cloud Services, most notably the Google Application Engine to host the Careaway Treatment Planner. Google Cloud Service will be a good alternative if Amazon Web Service cannot host our system due to any incompatibility or unforeseen changes in our system. We chose Google Cloud Services alternatively because it has the ability to run the deployment operating system, Linux Ubuntu. It also has the ability to integrate and use Node.js. We chose AWS as our primary hosting service because of our team’s familiarity of AWS.

**Source:** <https://cloud.google.com/>

**Cost:** No Cost

**License:** Google Cloud Platform Terms of Service

**Discussion for Primary Choice**

The team has chosen Amazon Web Service EC2 as our primary hosting service. We have chosen Amazon Web Service as it specifically meets our application needs to host and run our modified MEAN stack as it supports communication with MongoDB, Node.js, and other components for our system. Furthermore, Amazon Web Services also provides documentation about their servers and how to integrate specific technology components into the server allowing the team to understand how to integrate our stack into the web service.

In addition the team has listed Google Cloud Services as an alternative to our primary choice. Unlike Amazon Web Service, Google Cloud Service has a specific version that can comply with our MEAN stack which deterred our team to use as a primary web service. However, Google Cloud Services still provides the team with a means to integrate our MEAN stack and provides the team with a multitude of services for the CareAway Application.

## **Proxy**

### **Nginx (Primary)**

**Version:** 1.13.7 (Ubuntu)

**Description:**   
Nginx is a reverse proxy server that can be used to reroute traffic from one port to another.

**Reason:**

Our application will use Nginx as a proxy server for our system to route user ports to our system’s designated port. Nginx supports HTTPS porting, which we will use when we deploy our application. We chose Nginx as it was a free and open source service that offers high performance proxy servers with easy configuration. In addition, we decided on Nginx as it was compatible with Node.js and our Ubuntu operating system. As of now we are using the latest version of Nginx to deploy into our system.

**Source:** <https://nginx.org>

**Cost:** No Cost

**License:** 2-Clause BSD (FreeBSD License)

### **Alternative**

#### **mod\_proxy**

**Version:** 2.4

**Description:**

The module mod\_proxy is provided by Apache for reverse proxies.

**Reason:**

We will use mod\_proxy as an alternative since it is similar to Nginx where it creates a proxy. The reason we choose not to use mod\_proxy is because no one on our team is familiar with it.

**Source:** [https://httpd.apache.org/docs/current/mod/mod\_proxy.htm l](https://httpd.apache.org/docs/current/mod/mod_proxy.html)

**Cost:** No Cost

**License:** Apache License 2.0

**Discussion For Primary Choice**

Nginx is the primary choice for our proxy since it is easier to create a proxy server compared to mod\_proxy. Setting up the proxy and modifying the ports is easier to accomplish in Nginx.

## **Operating System**

### **Linux Ubuntu (Primary)**

**Version:** 16.04 LTS OS deployment Tool

**Description:**

An open source operating system that can run from a desktop and to our cloud service.

**Reason:**

Our application will use Ubuntu version 16.04 as the operating system to deploy and run our system. The reason we chose Ubuntu as our operating system is due to its ability to run our system across many platforms to allow many user to access our system. In addition, we are using the latest version of this operating system which is offered by Amazon’s Web Service to deploy our system.

**Source:** <https://www.ubuntu.com/download/desktop>

**Cost:** No Cost

**License:** GPL/ Free Software

### **Alternative:**

#### **Amazon Linux**

**Version:** 2017.03

**Description:**

A Linux distribution created by Amazon based on the Red Hat Enterprise Linux.

**Reason:**

Amazon Linux can deploy our system and easily integrate with Amazon Web Servers. We are using version 2017.03 as it has updated security features and uses Linux kernel 4.9, a more recent stable version.

**Source:** https://aws.amazon.com/amazon-linux-ami/

**Cost:** No cost

**License:** Amazon’s Software License

**Discussion for Primary Choice**

The team has chosen Linux Ubuntu as the primary operating system to host the CareAway Application. Linux Ubuntu is language agnostic which allows for a universal deployment of the system. In addition, the majority of the team has a copy of the Linux Ubuntu operating system and experience with using the operating system. Lastly, Linux pairs well with our MEAN stack given it’s language agnostic nature allowing it to understand many of the components of our system.

In addition the team has chosen Amazon Linux as our alternative choice to host the CareAway Application. As it comes from the Linux family, this operating system is also language agnostic which makes it compatible with many technologies in our MEAN stack. Amazon Linux is also only compatible with Amazon Web Services, which ties us down in terms of our choices for changing web services in the future.

## 

## 

## 

## 

## **Domain Naming Service**

### **NameCheap for Education (Primary)**

**Description:**

An Internet domain registrar that can register and sell domain names. It can also do web hosting.

**Reason:**

For our Domain Naming Service we chose NameCheap for Education due to it’s free trial for students for a whole year. NameCheap provides us the domain name for our single page application, which will allow users to freely access our system once it is deployed.

**Source:** https://nc.me/

**Cost:** Free Trial**.** 12 month subscription

**License:** Namecheap Universal Terms of Service Agreement

### **Alternatives**

#### **Domain.com**

**Description:**

Domain.com is a domain naming service that allows us to choose our domain name for our system if it is available over a wide ranges of high level domains

**Reason:**

The reason we chose Domain.com as an alternative is because of the high number of top level domains it offers. Such top level domains it provides are .com, .org, .net, .online, and many more. However, we decided to use Domain.com as an alternative due to its cost of $2.99 for our domain name our team wanted which was (careaway.net).

**Source:** <http://www.domain.com/>

**Cost:** $2.99 a month

**License:** Domain.com's Terms of Service

**Discussion for Primary Choice**

The team has chosen NameCheap for Education as our primary domain naming service. The main reason is the 12 month free trial it provides students, which allows our team a long period of time to host our website on. This is our primary choice as it complies with our budget of using free services in order to create our web application.

In addition is the alternative domain naming service which is Domain.com. The reason why Domain.com is our alternative is that it does not comply with our budget as it costs money to use their service to create a domain name for our web application.

# **Protocols**

## **HTTP (Primary For Testing Purposes)**

**Version:** 2.0

**Description:**

HTTP is a data communication protocol for the World Wide Web for distributed, collaborative, and hypermedia data.

**Reason:**

Since our solution is a web application, we will need a way to transfer data between the client and the server, which is what HTTP does. We are using HTTP as our data communication protocol during the testing phase of our system. Our team has decided to use HTTP for testing in order to have a prototype system to run different components on the system. We will later use HTTPS on our deployed system when we have all product ready components and system encryption ready. Our team is going to implement this protocol through a Node.js plugin which is specified in the back end technology.

**Source:** <https://www.w3.org/Protocols/>

**Cost:** No Cost

## **HTTPS (For Deployment Purposes)**

**Version:** 2.0

**Description:**

A secure data communication protocol for the exchange of data over the World Wide web. It’s HTTP with a security layer that consist of either the Transport Layer Security (TLS 1.2) or the Secure Sockets Layer (SSL), which encrypts the data sent to and from a server. HTTPS protects against man-in-the-middle attacks and eavesdropping. All traffic is routed through port 443.

**Reason:**

Our system will be transferring sensitive data to and from the server, and to prevent unwanted data leaks along the way, we will need a secure way of transfering that data. Therefore, our team has decided to use HTTPS as our data communication protocol for the deployment of our minimum viable product due to the security layer it provides for our system. HTTPS also ensures that the client-server connection is kept private and that the client isn’t communicating with an imposter.

**Source:** <https://https.cio.gov/>

**Cost:** No Cost

## **SSL**

**Version:** 3.0

**Description:**

A cryptographic protocol that provides a layer of security over the sockets of data transfer on the Internet.

**Reason:**

Since we are hosting a web application on the Internet, we will be sending a lot of data back and forth between clients and our server. This data will be critical data, which we will need to be encrypted to prevent someone from the outside from seeing it as it is being transferred. SSL will be used to prevent someone from seeing the data as it is being transferred since it will provide a secure layer of protection over the data.

**Cost:** No cost

## **TLS**

**Version:** 1.2

**Description:**

An upgraded version of SSL. Transport Layer Security (TLS) is a protocol for transferring data securely between two computer applications. The protocol uses a handshake to create the secure connection between a client and a server. The protocol ensures the data’s integrity and a connection’s authenticity.

**Reason:**

Due to the flow of data through our system, our application needs a secure way of moving data from the client to the server and vice-versa. The type of data we handle is very sensitive, and this data will be moving through the Internet. The data will be vulnerable without a layer of security on top of it. TLS provides the layer of security needed when transferring data to prevent it from being seen by unwanted eyes. It also protects the integrity of the data, ensuring that the data we send to the client and the data the client receives is the same. Furthermore, TLS ensures the identity of the client and the server.

**Cost:** No cost

# **Frontend Technology**

## **CSS/Styling Frameworks**

### **BULMA (Primary Choice)**

**Version**: 0.6.1

**Description**:

Bulma is an open free CSS framework that simplifies the CSS stylings of web development and gives a variety of choices in creating front facing UI. It is based off Flexbox and supports most current browser version.

**Reason:**

Bulma supports the SASS preprocessor and is a package on NPM. It is also very barebones compared to the other frameworks making it easier to learn. That approach makes it easy for the development needs. Bulma is also compatible with all the browsers in the scope and with the technologies that will be used in this project. Not only that, it has its own plugin support for Vue and Ember which is great for a seamless development experience in this project. The Css designs works with the components we build and doesn’t require a huge learning curve to incorporate the technology.

**Source:** https://bulma.io/

**Cost**: No Cost

**License:** MIT License

### **Syntactically Awesome Stylesheets (SASS/SCSS) (included in Bulma)**

**Version:** 3.5.1

**Description:**

SASS is a css preprocessor.

**Reason:**

SASS allows the team to use nesting, variables, and mixins.There is support in both Bulma and Vue for SASS. It will allow the team to create a style guide and make the over css development easier and more efficient. Due to the team’s familiarity with SASS, this will be the primary CSS framework for the web application.

**Source:** http://sass-lang.com/

**Cost:** No Cost

**License:** MIT License

### **Alternative(s)**

#### **Bootstrap**

**Version:** 4.0.0-beta.2.

**Description:**

One of these most common toolkits when developing a web page. Builds a multitude of items for web pages that include, but limited to Sass Variables, grid systems, components, and plugins.

**Reason:**

Bootstrap has a lot of support behind it, making it very convenient for the team to find support or mentorship online or offline. The support also ensures that bootstrap is always going to be able to support the latest technologies and needs of the community.

**Source:** http://getbootstrap.com/

**Cost:** No Cost

**License:** MIT License

#### **Cascading Style Sheets (CSS)**

**Version:** 4.0.0

**Description:**

Standard styling language for HTML

**Reason:**

CSS allows for styling of HTML and is the base form of any preprocessor. When used correctly it can be faster than with a preprocessor. Anything done with a preprocessor must be converted to CSS to be compiled.

**Source:** https://developer.mozilla.org/en-US/docs/Web/CSS

**Cost:** No Cost

**License:** MIT License

#### **Foundation**

**Version:** 6.4.3

**Description:**

Foundation is a framework that utilizes semantic and customizable styling for CSS and HTML. It includes templates and guides for developer support.

**Reason:**

Foundation provides our web application professional and customizable style sheets, javascript functionality, and a HTML building blocks. This framework is easily integrated into Vue.js as a Vue module and provides great documentation to assist our team in design and implementation of frontend page components.

**Source:** <https://foundation.zurb.com/>

**Cost:** No Cost

**License:** MIT License

## **Semantic UI**

**Version:** 2.2.12

**Description:**

Development framework that creates semantic layouts using easily readable HTML, utilizing JQuery for the behavioral functionalities.

**Reason:**

Semantic promises easy, concise, and intuitive development and debugging. This could be helpful for the development team since html can be difficult to organize to always look a certain way and having templates to guide the developers would help limit the time cost of styling.

**Source:**

**Cost:** No Cost

**License:** MIT License

#### **vue-foundation**

**Version:** 3.0

**Description**:

A Vue plugin that incorporates the foundation for sites and VueJS.

**Reason:**

This plugin will provide the semantic UI framework into the Vue project. It also can integrate the full Semantic UI frameworks to the Vue project if need be.

**Cost:** No cost

**License**: MIT License

**Source:** https://github.com/vue-foundation/vue-foundation

#### **Semantic-UI-Vue**

**Version: 0.1.1**

**Description**:

A Vue plugin that incorporates Semantic UI and provides its components to the project.

**Reason:**

This plugin will provide the foundation framework into the Vue project. Also it has native support with Vue CLI and creating single page applications.

**Cost:** Nocost

**License**: MIT License

**Source:** https://github.com/Semantic-UI-Vue/Semantic-UI-Vue

**Discussion for Primary Choice**

The team has chosen Bulma as the primary styling framework for the project because Bulma works hand in hand with SASS preprocessor and Node. Bulma also works well with VUE and has many VUE modules that can already be imported. The Framework itself is pretty easy to learn and has a large amount of customizable features. The company developing also keeps the software regularly updated and post guides on how to start or even make custom features such as layouts and forms.

In addition, the team has chosen Bootstrap as the primary alternative. Bootstrap is a powerful all in one framework, and provides the team with a lot of built in and custom features with a heavy supporting from the Mean Stack community. This makes it a very strong alternative as it has the largest foundation to work on, but the team’s inexperience in using bootstrap hinders it as a choice for the team. Bootstrap requires a lot more effort to learn and get acquainted with and might provide more, but is not really helpful in the perspective of the scope. Bulma is far easier to learn and has straight forward support with the Vue and Ember Framework.

## **Javascript Frameworks**

### **VueJS (Primary)**

**Version:** 2.5.8

**Description:**

Vue is a progressive front end framework designated to be easily adoptable and straightforward. Vue is view layered focused making it easy to integrate or creates projects. It is also geared towards and capable of handling single page applications.

**Reason:**

Vue is helpful for minimizing the work in the view layer of this project. It’s adoptability and simplicity should be easily integrated to the team’s experience in vanilla javascript and node. This makes the separation of concepts consistent with the mvc architecture. In addition, Vue also has great support for the Single Page Application. The project is also oriented in the same structure allowing for fluid and constant support in development. Vue also takes more approachable approach to components and straight forward to as to which page view corresponds to which functionality.

**Source:** https://vuejs.org/

**Cost:** No Cost

**License:** MIT License

### **Alternative(s)**

### **EmberJS**

**Version:** 2.16.0

**Description:**

Ember is a “framework for ambitious web applications.” It is a frontend javascript

framework that has a good code organization and is open source. It especially good because it has sharing conventions that allow add ons and plugins to be added and modified to fit any app.

**Reason:**

Our application will use ember.js for the applications frontend framework. We chose ember.js because it allows for easy integration of frontend plugins and will integrate well with the other parts of our web app and our project manager has experience with building ember apps. We chose ember 2.15.0, because it is the latest stable release. Perhaps the most important reason for using ember.js is for it’s component based architecture. Ember components are simple to integrate into a project and allow for efficient code reuse. Our app will rely on reusable widgets, which can easily be made into components that can be reused throughout the app. Ember also allows us to create API endpoints that will make integrating the API and database much easier.

**Source:** https://www.emberjs.com

**Cost:** No Cost

**License:** MIT License

### **Angular**

**Version:** 5.0.5

**Description:**

Angular aims to be the only framework needed for both mobile and desktop applications. It is the chosen front end javascript framework for the MEAN stack and built for progressive web apps.

**Reason:**

Angular being usually the top 2 of the javascript frameworks has the most support out of all of them and have a great amount of resources with ides and text editors. This includes development environments and linters. The component router also optimizes the applications making the user experience a lot smoother compared to some other frameworks with the added benefit of performance as well.

**Source:** https://www.angular.io

**Cost:** No Cost

**License:** MIT License

#### **Discussion for Primary Choice**

The team has chose Vue as the primary pick for the front-end javascript framework because of its barebones style. The framework is easy to learn and very straight forward when it comes to component based javascript. In addition, The framework has been known to have taken the best features from React and Angular. The team can take advantage of those features into the project while also using the minimum amount of work in doing so. Vue also has major supporting for the plugins in that will be implemented into CareAway.

On the other hand, the team has chosen Ember as the primary alternative, this is because a team member has a massive supporting for it at work and mentors to learn from. This resource can be invaluable. Ember also provides a massive amount of modules, plugins, and component support in its framework. However, this massive amount of support is not needed in the scope of this project, Ember can be overwhelming and has a higher learning curve than Vue. In another perspective, Vue claims to have full support in building Single Page Applications in an MVC style architecture. Vue being the View in the Model View Controller.

### **Plugins (Calendar):**

#### **Vue-simple-calendar(Primary)**

**Version:** 1.0.9  
**Description**:  
A Vue plugin that creates a calendar that requires no JQuery and works with Vue 2.0+.  
**Reason**:   
This simple calendar has the same functionality as fullcalendar.io, but requires no JQuery. This directly works with the vue framework and is built on top of the Flexbox layout.   
**Cost**: No cost  
**License**: MIT License  
Source: https://www.npmjs.com/package/vue-simple-calendar  
**Alternative(s)**

#### **Vue-fullcalendar**

**Version:** 1.0.9

**Description**:

A Vue plugin that creates a calendar that requires no JQuery and works with Vue 2.0+.

**Reason:**

This fullcalendar will be the plugin to implement the calendar that will be used in the main view for the user. The calendar plugin directly works with vue and gives the calendar view for the projects requirements.

**Cost:** No cost

**License**: MIT License

**Source:** https://github.com/Wanderxx/vue-fullcalendar

#### **FullCalendar.IO**

**Version:** 2 & 3

**Description**:

A javascript event calendar, easily customizables and has support for different views and themes. Utilizes JQuery for events and calendar behaviors.

**Reason:**

This fullcalendar will give the dev team the calendar that will be used in the main view for the user. This calendar simplifies the creation of treatments and appointments and has drag and drop feature built in. This means the creation of the widgets and custom widgets should be simple and easy to use.

**Cost:** No cost

**License**: MIT License

**Source:** <https://fullcalendar.io>

#### **Discussion for Primary Choice**

The team has chosen to use Vue-simple-calendar because of its versatility and support. This calendar does not need JQuery or other external dependencies. It come in a vue package via npm and works directly with the framework. The features included are the drag and drop, calendar, scheduling, and onclick events for the date events. This will be handy in creating treatments and appointments. The developers for this plugin have also included tutorials and guides to creating and customizing the calendar and its drag and drop features. The learning curve be short and implementation should be fluid.

The main alternative to this would be FullCalendar.io, it is simple and quiet easy to implement. There are many views included, such as the day view which gives insight on events on a given day. However, this has a heavy dependence on jquery. The team has little experience in creating applications with jquery and have no idea how to make custom components or features with this. The Vue-simple-calendar has most of the core functionalities and has a straightforward build. Adding custom widgets and components should be seamless and easier to implement as opposed to doing it on Fullcalendar.io & jquery.

### **Plugins (Charts):**

#### **Vue-chartjs (Primary)**

**Version: 3.0.2**

**Description**:

A Vue plugin that incorporates charts.js enables creation of charts and data aggregation visualizations

**Reason:**

This plugin will provide the charts for the project that require the patient data analysis. It includes all of chart.js and is easy to learn and customize. Many developers also contribute and support the projects along with tutorials and easy module implementation.

**Cost:** Nocost

**License**: MIT License

**Source:** https://github.com/apertureless/vue-chartjs

### **Alternative(s)**

#### **Chart.js**

**Version:** 2.7.1

**Description**:

Chart.js is a javascript charting tool that has simple chart components and designed to have a simple material design look.

**Reason:**

Chart.js will be very helpful in creating the charts and data aggregation for the patient. The plug in serves as a alternative if the Vue component were to not work, but is still fairly straightforward. Implementing a chart.js will be a lot quicker and easier than completely aggregating and creating graphs in the back end.

**Cost:** No cost

**License**: MIT License

**Source:** http://www.chartjs.org/

**Discussion for Primary Choice**

The team has chosen the Vue-chartjs plugin as the main chart plugin because of its support with vue and having include the bulk of ChartJS. The plug in is relatively straight forward to implement as it uses the vue cli to install and run. Aggregating tutorials are included in the github and most of the learning for this plugin can be found there as well. The Vue plugin also includes all the charts needed to complete the project’s user requirements for Data Analysis Graphs.

The alternative for the charts functionality would be to use chart.js itself. This a viable option since it includes all the latest features and bug fixes. However, those are not needed in this scope. Having the simplest learning curve with highest fluidity is most important. The advantage of the plugin is that it is known to work with great success and already has more than enough to display the project’s aggregate data

### **Plugins (Drag & Drop):**

#### **Vue.Draggable(Primary)**

**Version:** 2.15.0

**Description**:

An Vue.JS plug in that utilizes the drag and drop feature to Vue projects.

**Reason:**

The team will need the drag and drop functionality in multiple areas in the project and having the plugin to do it for them would remove time consumed in making that functionality work. Drag and dropping can be used for list, widgets, and appointments in this project.

**Cost:** No cost

**License**: MIT License

**Source:** https://github.com/SortableJS/Vue.Draggable

### **Alternative(s)**

#### **Vue-drag-drop**

**Version:** 0.2.2

**Description**:

A barebones Vue.JS plug in that utilizes the drag and drop feature to Vue projects.

**Reason:**

This plugin also gives the ability of giving the team the functionality of dragging and dropping. However this one is more barebones and can be easily customizable for the team’s needs.

**Cost:** No cost

**License**: MIT License

**Source:** https://www.npmjs.com/package/vue-drag-drop

#### **Discussion for Primary Choice**

The team has chosen Vue.Draggable, the plugin has continous support that features drag handles and selectable text, smart auto scrolling, dragging between list, and no JQuery dependency. It also supports event reporting, Vue transition-group, and synchronization with viewmodel list. While the team will not be utilizing all these features they are a great benefit in making the listable views cleaner and easier to manipulate. This comes important with moving around tasks or list items.

The team has chose, vue-drag-drop as as the alternative for the drag & drop feature. This plug is also integrated with Vue and requires no other dependency. It’s very bare bones and can be easily manipulated and customizable. However, this plugin is relatively new and can change at any given update and might no longer support on the team’s goals with the feature. That is why the more mature Vue.Draggable has been chosen.

# 

# **Backend Technology**

## **Runtime Engine**

### **Node.js (Primary)**

**Version:** 8.9.1

**Description:**

Node.js is a JavaScript runtime engine written in C++ that is used to execute JavaScript code server-side.

**Reason:**

The server will be written in JavaScript, and NodeJS provides a JavaScript engine for running the server-side code. Our team chose to use Node.js since it can be used cross-platform, meaning it’s compatible with each of our development operating systems and our deployment operating system, Ubuntu 16.04. Its availability over multiple platforms allows us to focus on the development of the actual product rather than worry about its compatibility. Some of our team members have had experience using Node.js. Node.js is easy to understand and learn since it is written using JavaScript. Node.js also provides plenty of documentation and plenty of aids to help our project progress. Node.js also comes with npm.

**Cost:** No cost

**License:** MIT License

**Source:** <https://nodejs.org/en/>

### 

### 

### **Alternative**

#### **Nashorn**

**Version:**

**Description:**

Nashorn is a JavaScript engine written in Java used to run JavaScript code.

**Reason:**

Since the development of our system’s back-end will be written in JavaScript, we would need a JavaScript engine to run it and Nashorn is developed to accomplish that task.

**Cost:** No cost

**License:** GPLv2

**Source:** <http://openjdk.java.net/projects/nashorn/>

**Discussion For Primary Choice**

Node.js was chosen over Nashorn as our primary JavaScript engine. The advantage of using Node.js is the security benefits from a well-maintained product as compared to Nashorn, which hasn’t been updated in over a year to this date (November 29, 2017). The ecosystem that Node.js provides is far more extensive and rich than Nashorn’s. Node.js’s V8 engine is faster than Nashorn’s engine. Also, Node.js has more documentation and is more organized than Nashorn (Nashorn doesn’t even have a version number). Node.js comes with a package manager, npm, without any configuration needed. Nashorn does not come with a package manager.

## **Package Manager**

### **Npm (Primary Choice)**

**Version:** 5.5.1

**Description:**

Npm is a package manager that comes with Node.js that provides a plethora of libraries for applications using JavaScript. The package manager manages the installation and versions of all of the dependencies needed for a JavaScript project.

**Reason:**

The system will need certain libraries and dependencies to accomplish the tasks it is meant to do. With npm, we will be able to focus on building an optimal product. Npm gives us access to many libraries, providing us the resources we need so we can focus on the important parts of our product. The package managing provided by npm takes a lot of the busy-work out of managing and figuring out how to install libraries/versions/dependencies, which allows us to focus on building an optimal product. Npm comes automatically installed when NodeJS is installed, which is convenient. Npm is also cross-platform, so it’s easy for every member to install it and use it, we won’t have to worry about compatibility issues. We can use npm on both the front-end and back-end systems, allowing us to simplify our workflow.

**Cost:** No cost

**License:** npm Open Source

**Source:** <https://www.npmjs.com>

### **Alternative**

#### **Bower**

**Version:** 5.5.1

**Description:**

A front-end package manager.

**Reason:**

Bower is a viable option for managing dependencies and libraries on our system, however, bower is mainly used for front-end packages.

**Cost:** No cost

**License:** MIT License, Copyright 2016 Twitter and other contributors: <https://github.com/bower/bower/graphs/contributors>

**Source:** <https://bower.io>

**Discussion For Primary Choice**

Npm is the primary choice as our package manager since it comes installed with Node.js and contains packages for both the front-end and back-end. Bower contains primarily packages for the front-end. Some packages that are available through npm for back-end servers will not be available through bower. We would have to manually create the solution ourselves, which takes time away from the development of the main functionalities our system. Furthermore, bower would need additional configuration to install on Windows and possibly on Ubuntu, causing a disruption to the development environment of some of our team members. Overall, the reason we chose npm over bower as our package manager is because npm is more convenient and provides more server-side packages when compared to Bower. However, bower is a viable alternative in the event we would need to switch.

## 

## 

## **Plugins**

### **Express (Primary)**

**Version:** 4.16.2

**Description:**

Express is a framework built on top of the HTTP server provided by Node.js and it provides additional functionality for creating and managing a server. In addition to server creation, Express provides a plethora of middleware to handle communication between a client and the server.

**Reason:**

Our solution is a web application, which needs a server to host it. Express is used to create the server and allow data transfer between the server and the client. The functionalities provided by Express lets us focus more on the important parts of our system rather than worry about how data will be sent to and from the server. Express is a package that is available through npm, so it’s simple to install and include as a dependency for our project. A few of our teammates have used it before so it’s familiar to us.

**Cost:** No cost

**License**: [Creative Commons Attribution-ShareAlike 3.0 United States License](http://creativecommons.org/licenses/by-sa/3.0/us/)

**Source:** <https://expressjs.com>

### **HTTP (Primary for Development)**

**Version:** Current Release

**Description:**

The HTTP framework is provided by Node and interfaces with Node.js to provide HTTP methods for the application.

**Reason:**

HTTP is a library provided by Node that is easy to use and provides a protocol for communicating between the server and the client. For development purposes, HTTP will be used to send data to and from the server.

**Cost:** No cost

**License:** MIT License

**Source:** <https://nodejs.org/api/http.html>

### **HTTPS (Primary for Deployment)**

**Version:** Current Release

**Description:**

This is the secure version of the HTTP module for the HTTPS protocol. It is a way to transfer data securely between the client and server.

**Reason:**

When we deploy our application, we will need a secure method of transporting the sensitive data we handle. HTTPS is a library provided by Node that is easy to use and provides a protocol for sending data securely between the server and the client.

**Cost:** No cost

**License:** MIT License

**Source:** <https://nodejs.org/api/https.html>

### **MongoDB (Primary)**

**Version:** 2.2.33

**Description:**

MongoDB is a high-level API used to access and manipulate a Mongo Database.

**Reason:**

Since we are using a Mongo Database as our database, it is appropriate to use a MongoDB plugin to access that database via the server. MongoDB simplifies our transactions with the database system through an easy-to-use API. We will be able to easily access our database without having to worry about a multitude of configurations and low-level details. This library is also provided by npm, and it is easy to install and use.

**Cost:** No cost

**License:** Apache License 2.0

**Source:** <https://www.npmjs.com/package/mongodb>

### **Alternative**

#### **MongooseJS**

**Version:** 4.13.5

**Description:**

Mongoose is an API that allows manipulation and access of a Mongo database.

**Reason:**

MongooseJS provides us an alternative way of accessing the Mongo database. It’s compatible with NodeJS and is easy to use. The API is for the database system we want to use (MongoDB). There is also a lot of documentation that can guide us when we get stuck.

**Cost:** No cost

**License:** MIT License

**Source:** <http://mongoosejs.com>

**Discussion For Primary Choice**

Although both MongoDB and MongooseJS allow access to a Mongo database via a high-level API, MongoDB does not require the use of schemas to use the database. MongooseJS’s schema model is useful, but for our minimum viable product, it is unnecessary. This is why we’re using MongoDB instead, which is simple and allows us to accomplish our task, which is accessing a Mongo database.

### **Path (Primary)**

**Version:** 2

**Description:**

Path is an API with the tools necessary to access files in a directory.

**Reason:**

Since we will be using an operating system to host our server, we will store some of the files we need on that operating system. Path provides us a way to access those files and directories. In the future, if we need to migrate to a different operating system, Path will take care of the actual design of those paths, which are specific to the operating system. It’s better to use this module overall since it takes care of the lower-level specifics and allows us to focus more on development of the actual functionalities specific to our product.

**Cost:** No cost

**License:** MIT License

**Source:** <https://nodejs.org/docs/latest/api/path.html>

### **Express-session (Primary)**

**Version:** 1.15.6

**Description**:

A middleware layer that creates sessions on an express server. The session data is stored on the server, and the session identifier is sent to the client.

**Reason:**

Since certain parts of our web application are only accessible to those who are logged into our system, our server will need to keep track of who is allowed (authenticated) into our system. The Express server alone does not provide this functionality, so we will need this plugin to create and manage sessions.

**Cost:** No cost

**License:** MIT License

**Source:** <https://www.npmjs.com/package/express-session>

### **Alternative**

#### **Cookie-session**

**Version:** 2.0.0-beta.3

**Description**:

Middleware used to store a user session by storing session data on the client in a cookie.

**Reason:**

The web application we’re developing restricts access to certain users. Our system requires a way to verify that a client is actually allowed access to a portion of our site, and sessions are required to make sure the client connecting is who they say they are. Therefore, cookie-session comes into play by creating sessions and verifying incoming sessions.

**Cost:** No cost

**License:** MIT License

**Source:** <https://expressjs.com/en/resources/middleware/cookie-session.html>

**Discussion For Primary Choice**

Express-session is our primary choice since we will be handling sensitive data so we want the server to store the session data rather than the client. In the server, the data will be more tightly locked up, and this is preferable than sending it all to the client where the endpoint might not be secure. Cookie-session stores session data on a client, whereas Express-session stores session data on a server.

### **Body-parser (Primary)**

**Version:** 1.18.2

**Description**:

Body-parser is a middleware layer provided by npm that parses the bodies of incoming client requests.

**Reason:**

Body-parser is available via npm and it’s easy to install and use. JSON objects are what we’re going to use to send objects to and from the server, which is why we need body-parser to parse these objects. Body-parser also supports urlencoded objects, and in the event we will need to parse urlencoded objects, body-parser will take care of that. Body-parser allows us to create our own headers and choose how we parse those requests, giving us more capability and flexibility if we need it.

**Cost:** No cost

**License:** MIT License

**Source:** <https://www.npmjs.com/package/body-parser>

### **Alternative**

#### **body**

**Version:** 5.1.0

**Description:**

Body is a library used to parse incoming client requests.

**Reason:**

Information is sent between the client and server in various formats. This API gives us a way to easily decode JSON objects, form objects, etc. and it is provided by npm so we could easily install and use it.

**Cost:** No cost

**License:** MIT License

**Source:** <https://www.npmjs.com/package/body#readme>

**Discussion For Primary Choice**

Body-parser was chosen as our primary parsing middleware due to its compatibility with Express. With body-parser, it is simple to include the middleware layer in our server once, and the framework will do the rest for us. Body is a viable alternative for parsing the body of a client request. However, body requires you to constantly reuse their parsing objects whenever you need to parse data. You would have to know exactly what type of data is being sent to your server at a specific route, whereas body-parser knows how to parse the data sent to the server via a request’s content-type. Thus, body-parser is easier to use and simplifies our server.

### **Passport (Primary)**

**Version:** 0.4.0

**Description:**

Passport is used for authentication on an Express server.

**Reason:**

Passport is provided by npm so it’s easy to install and use. We will use passport to authenticate our users on the server side. It’s compatible with Express and its session handlers so there is no extra work that needs to be done when using this plugin for authentication.

**Cost:** No cost

**License:** MIT License

**Source:** <https://www.npmjs.com/package/passport> & <http://www.passportjs.org>

### **Crypto-js (Primary)**

**Version:** 3.1.9-1

**Description:**

Crypto-js provides tools for encrypting and decrypting data.

**Reason:**

The system handles sensitive data that will need to be protected. Encrypting the data provides an extra layer of protection, and crypto-js has the tools for encrypting and decrypting it. The library is readily available via npm so we could easily install and use it.

**Cost:** No cost

**License**: MIT License

**Source:** <https://www.npmjs.com/package/crypto-js>

### **Alternative**

#### **Bcrypt**

**Version:** 1.0.2

**Description:**

Bcrypt is a tool used for hashing.

**Reason:**

Our system will be using usernames and passwords as the primary form of authentication, and these passwords are sensitive. For security purposes, the passwords of each user will need to be encrypted and bcrypt’s hashing functionality allows us to do this.

**Cost:** No cost

**License**: MIT License

**Source:** <https://www.npmjs.com/package/bcrypt>

**Discussion For Primary Choice**

Crypto-js comes with an abundance of cryptography tools to encrypt and decrypt data, whereas bcrypt only allows data hashing. The tools that crypto-js supplies will be useful for our application due to the type of information we will be handling. In the event that we choose to encrypt more data than what our minimum viable product states, crypto-js will already have the tools ready for us to use.

### **CSURF (Primary)**

**Version:**

**Description:**

A middleware for validating incoming client requests and preventing cross-site request forgery (CSRF) attacks by creating a token on the server and sending it to the client. If the client’s token doesn’t match the server’s token, then the middleware will prevent the client’s request from progressing any further.

**Reason:**

Cross-site request forgery (CSRF) attacks are a huge security vulnerability to web applications, and the CSURF plugin will be used on our server to prevent this type of attack from harming our system.

**Cost:** No cost

**License**: MIT License

**Source:** <https://github.com/expressjs/csurf>

### **Helmet (Primary)**

**Version:** 3.9.0

**Description:**

Helmet is a collection of middleware for setting HTTP headers correctly to secure applications.

**Reason:**

Our web application handles sensitive information that will need to be secured, and Helmet provides an extra layer of security by ensuring HTTP headers are set correctly.

**Cost:** No cost

**License**: MIT License

**Source:** <https://helmetjs.github.io> & <https://github.com/helmetjs/helmet>

### **Jasmine (Primary)**

**Version:** 2.8.0

**Description:**

A behavior-driven framework provided by npm used for testing applications written in JavaScript. Jasmine does not require a browser to run tests.

**Reason:**

Our server is written in JavaScript and uses Node.js. Tests will need to be run on our code to ensure it functions properly. Jasmine is a framework used to create these tests and it is compatible with Node.js and can easily be installed via npm. There are no additional modules needed, all of the tools are provided by Jasmine and are ready to use.

**Cost:** Free/No cost

**License:** MIT License

**Source:** <https://github.com/jasmine/jasmine> & <https://jasmine.github.io>

### **Randomstring**

**Version:** 1.1.5

**Description:**

Randomstring creates random strings.

**Reason:**

Our team might make use of this library for generating random strings for the Medical Professional Code and user sessions. This library is available via npm so it’s easy to install and ready to use.

**Cost:** No cost

**License**: MIT License

**Source:** <https://www.npmjs.com/package/randomstring>

### **Forever**

**Version:** 0.15.3

**Description**:

Forever allows a script to run continuously (forever).

**Reason:**

Our team will make use of forever to keep our server running forever once it’s finished and ready to be deployed. Forever is a library provided by npm and is easy to install and use.

**Cost:** No cost

**License**: MIT License

**Source:** <https://www.npmjs.com/package/forever>

## 

## **Database**

### **MongoDB (Primary)**

**Version:** 3.4.10

**Description:**

A non-relational database system.

**Reason:**

Our application requires a persistent data store and MongoDB provides a NoSQL (non-relational) database to store all our data. MongoDB is scalable, which will be useful when our application gains a lot of users. MongoDB is very well-documented so our team can pick it up quickly. We chose MongoDB for its flexibility and its ability to integrate well with NodeJS.

**Cost:** No cost

**License**: [GNU Affero General Public License](https://en.wikipedia.org/wiki/GNU_Affero_General_Public_License) version 3, Apache License 2.0, MongoDB Terms of Use

**Source:** <https://www.mongodb.com>

### **Alternative**

#### **Couchbase**

**Version:** 5.0

**Description**:

A non-relational database system.

**Reason:**

Couchbase is an alternative database system that also employs NoSQL. It is highly scalable, which will fit our needs and demands. It also integrates well with NodeJS.

**Cost**: No cost

**License:** Apache License, Couchbase Proprietary

**Source:** <https://www.couchbase.com>

**Discussion For Primary Choice**

MongoDB was chosen as the primary database over couchbase since MongoDB has npm packages that allow us to easily access it. The ease of access is huge since writing a driver for a database would be excessive and redundant.

## **HTTPS Certificate**

### **Certbot** (Primary)

**Version:** 0.19.0

**Description:**

Certbot is an automation tool that generates free HTTPS certificates from letsencrypt and applies that certificate to a server. It uses an ACME protocol to fetch the certificate. This client handles all configurations for serving HTTPS on a webserver and it provides tools for automatic certificate renewal.

**Reason:**

Our product contains highly sensitive data and it needs security to protect it. Data transfer also takes place in our product, and we need that to be secure to prevent a loss of data to unwanted observers. HTTPS is a protocol for transferring data securely. To secure our web server with HTTPS, we would need a certificate to be configured on our web server. Certbot provides tools that handles all of the configurations needed for serving HTTPS on our server. It provides a tool for automatically renewing certificates too. All of these tools will make transitioning into HTTPS simple for us and allows us to focus on developing our project without having to worry about configuring the security of our system.

**Cost:** No cost/Free

**Source:** https://certbot.eff.org

**License:** Apache 2.0 license

## Encryption

AES and SHA-256 <https://en.wikipedia.org/wiki/Advanced_Encryption_Standard>

# **Testing Technology**

## **Google Chrome and Chrome DevTool (Primary)**

**Version:** v.62.0.3202

**Description:**

Google Chrome is a popular web browser that serves as our platform to run the CareAway Treatment Planner. It also has an integrated DevTool to inspect different components on our system.

**Reason:**

We chose Google Chrome to debug and test the implementation of the CareAway Treatment Planner. We chose this technology to test as it is one of the browser the web application supports, and it provides us with development tools to inspect each individual component on the web application and can edit code during the application’s run time.

**Cost:** No cost

**License:** Freeware and Google’s Term of Service

**Source:** <https://www.google.com/chrome/browser/desktop/index.html>

## **Mozilla FireFox and Page Inspector**

**Version: 57.0**

**Description:**

An open source web browser that our CareAway Treatment Planner supports. The browser contains a Page Inspector to inspect different components on a web page.

**Reason:**

We chose Mozilla FireFox to test the implementation of the CareAway Treatment Planner. We chose this technology to test as it is one of the browser the web application supports. In addition, the browser provides us with development tools to inspect each individual component on the web application and provides us with visuals on the database, network, performance of our web application and much more.

**Cost:** No cost

**License:** MPL 2.0

**Source:** <https://www.mozilla.org/en-US/firefox/new/>

## **Apple Safari and Web Inspector**

**Version:** 11.0

**Description:**

A web browser for the macOS Operating System. It contains a web inspector to inspect components on a web page.

**Reason:**

We chose Apple Safari to test the implementation of our CareAway Application as it is one of the browser the CareAway Application Supports. It has a built in Web Inspector that our team can utilize to locate and fix any issue we may find in the system.

**Cost:** No cost

**License:** Freeware with some components are GNU LGPL

**Source:** <https://www.apple.com/safari/>

**Discussion for Primary Choice**

The team has chosen Google Chrome DevTool as the primary tool to test the CareAway Application. The reason is that it’s many of the team’s default browser and it the team decided to do testing mainly on the Google Chrome Browser.

The team will use Apple Safari Web Inspector and Mozilla Firefox Page Inspector as an alternative. Since testing of the the CareAway Web Application will mainly be done on Google Chrome, it was decided that the team should also use the DevTool built into the browser. Although, the team may utilize the Mozilla FireFox Page Inspector and the Apple Safari Web Inspector in order to ensure that the application runs accordingly to the requirements.

## **Node.JS and Node-Inspector(Primary)**

**Version:** 11.0

**Description:**

Node.js is a JavaScript framework that is used to create web servers. Node-Inspector is a node.js plugin that is used to debug node.js commands.

**Reason:**

We will be using Node.js to host a local server on each member’s computer to locally debug and test the implementation of each functionality. In addition, we will include Node-Inspector to provide debugging tools to help locate any errors throughout our CareAway Treatment Planner.

**Cost:** No cost

**License:** Freeware with some components are GNU LGPL

**Source:** <https://nodejs.org/en/> **/** <https://github.com/nodejs/node-inspect>

## **Jasmine (Primary)**

**Version:** 2.8.0

**Description:**

A behavior-driven framework provided by npm used for testing applications written in JavaScript. Jasmine does not require a browser to run tests.

**Reason:**

Our server is written in JavaScript and uses Node.js. Tests will need to be run on our code to ensure it functions properly. Jasmine is a framework used to create these tests.

**Cost:** Free/No cost

**License:** MIT License

**Source:** <https://github.com/jasmine/jasmine> & <https://jasmine.github.io>

## **Alternative**

### **protocol**

**Version:** 11.0

**Description:**

Node.js is a JavaScript framework that is used to create web servers. Node-Inspector is a node.js plugin that is used to debug node.js commands.

**Reason:**

We will be using Node.js to host a local server on each member’s computer to locally debug and test the implementation of each functionality. In addition, we will include Node-Inspector to provide debugging tools to help locate any errors throughout our CareAway Treatment Planner.

**Cost:** No cost

**License:** Freeware with some components are GNU LGPL

**Discussion for Primary Choice**

The team will be utilizing Jasmine as the primary testing framework for the CareAway Treatment Planner. The reason is that Jasmine provides good documentation and example code to how to create tests for javascript code. In addition, Jasmin is fast in testing due to lower overhead and no need to use the DOM to test the code.

Protocol was chosen as an alternative to Jasmine as a testing framework. The main reason is that Protocol is still subjected to experimental changes that may change throughout the development of our system. The team wanted a more stable framework to use for testing thus Jasmine provided such need for our team.

# **Glossary**

**2-Clause BSD (FreeBSD License)-** Users are allowed to redistribute and use source/binary code only if two conditions are met:

1. The redistribution of source code under this license must retain the same copyright notice
2. The redistribution of the source’s binary form must also reproduce the same copyright notice

**Apache License 2.0-** This license allows the user of the software the freedom to use,distribute, or modify the software. It also states that the software must preserve the copyright license if distributed.

**Charityware-** Software that is licensed that benefits a charity. Charityware software are distributed for free, and may include author’s suggestion to make payment to some charity.

**Creative Commons Attribution-ShareAlike 3.0 United States License-** Allows users to share material or content under the license only if the user provides appropriate credit to the author of the material.

**EULA (End User License Agreement)-** legal contract between users and software application author/publisher. It states that the user agrees to pay for the privileges of using a software while complying with the restrictions that come with it.

**Freeware**- Software that is available for free use/ no cost to use

**GNU/GPL (General Public License)** - A free software license, allows users to run, study, share, and modify the software freely

**GNU GPL v2 (General Public License Version 2)-** Encompasses GNU, granting end users the right to use, study, share, or modify software. In addition the user can distribute the material only if it satisfies the same copyright when distributed

**GNU LGPL v2.1 (General lesser Public License Version 2.1)-** A free software license that allows developers and companies to use the software under this license into user’s software even if it is proprietary software.

**Legal Term of Use/ Terms of Service -** rules that users must agree to and abide by in order to use a service

**MIT (Massachusetts Institute of Technology)** - **Most Permissive Free software License,** Permits the reuse of proprietary software provided that copies of the licensed software includes that MIT License terms

**MPL 2.0 (Mozilla Public License Version 2) -** A free and open source software license made by the Mozilla Foundation. It gives the user liberal copyright and patent licenses which allows them to use, modify, or distribute the software but does not give the right to the contributor’s trademark of the material.

**Open Source-** Software where the original source code is made freely available to use, redistribute or modify

**PHP License v3.01-** Allows the redistribution and use of source/binary code with or without modification only if the condition in the License have been met.

Reference: <https://secure.php.net/license/3_01.txt>

**Proprietary-** Legal Ownership of software

**VIM License-** Permits the distribution and use of copies of VIM software but must include license text to use