**Link Tailor**

Team Contract

University of Cincinnati

College of Education, Criminal Justice and Human Services

School of Information Technology

Senior Design Project 2021

Daniel Hickman

Kemal Ozturk

Lauren Tillery

## Table of Contents

[Table of Contents 2](#_Toc50723540)

[Contract Intent 3](#_Toc50723541)

[Project Introduction 4](#_Toc50723542)

[Project Name 4](#_Toc50723543)

[Problem Statement 4](#_Toc50723544)

[Solution 4](#_Toc50723545)

[The Team 5](#_Toc50723546)

[Responsibilities and Contact Information 5](#_Toc50723547)

[Team Values and Rules 7](#_Toc50723548)

[Communication Plan 8](#_Toc50723549)

[Problems and Escalations 8](#_Toc50723550)

[Project Breakdown 9](#_Toc50723551)

[Project Source 9](#_Toc50723552)

[Project Examination 9](#_Toc50723553)

[Project Goals 11](#_Toc50723554)

[Project Scope 12](#_Toc50723555)

[Project Timeline 13](#_Toc50723556)

[Technologies Used 15](#_Toc50723557)

[Team Signatures 16](#_Toc50723558)

## Contract Intent

The following contract was written and agreed upon by Daniel Hickman, Kemal Ozturk, and Lauren Tillery. The contract provides expectations, objectives, and methods for the development of the Link Tailor Application.

This contract is effective for all team members participating in IT 5001/5002 during the 2020-2021 school year.

# Project Introduction

**Senior Design Team Contract**

**2020-2021**

Project Name: Link Tailor

Problem Statement**:** Keeping track of browser bookmarks, descktop icons, file system shortcuts and other “quick links” can be a frustrating task for computer users who use many links a day[[1]](#endnote-1) and frequently add new ones.[[2]](#endnote-2) Desktops get cluttered with files and documents, bookmark bars fill to the brim, and internet browsers cram with tabs.[[3]](#endnote-3) This is an increasingly relevant concern as a growing number of computer users[[4]](#endnote-4) are accessing a growing number of websites and applications on a daily basis[[5]](#endnote-5). Current solutions such as Raindrop.io for bookmarks, Pocket for articles, and the Windows Start menu for applications are helpful, yet many users desire additional capabilities.[[6]](#endnote-6)

Solution: The Link Tailor application is a desktop application that allows users to quickly and easily save links – web links, application links, and file system links – adding them to layouts, tagging them, and organizing them spatially. Users can use this application as a supplement to their current link managing options (such as the Windows Start Menu and Chrome Bookmarks). The Link Tailor is designed to be convenient for a user who wishes to simply use the app to save and return to links without losing them, but also provides deep functionality beyond what current link managing solutions provide. One unique feature is the use of spatially organized widgets, which can placed anywhere on the screen and accessed quickly by mouse movements or hotkeys. While some link managing applications today are techy-oriented, not easily navigated by regular users, the Link Tailor app is designed to be easily beneficial with minimal effort and to guide users into good link-organizing habits.

# The Team

“The team” refers to the three parties of this contract who are working together to create the Link Tailor application.

## Responsibilities and Contact Information

**Daniel Hickman**

Project Manager

leinadhickman@gmail.com

hickmadc@mail.uc.edu

513-438-1597

Responsibilities

* Schedule and facilitate team meetings based on the Agile methodology. This will include Sprint Planning Sessions, Stand-ups, Sprint Reviews, and Retrospectives.
* Maintain the Kanban board, track velocity, and update the project timeline.
* Consider future features and groom the Product Backlog with the team.
* Take on the Developer roles in a smaller but meaningful way, including pushing code on GitHub.
* Present completed work at the Sprint Reviews and to class advisors.

**Kemal Ozturk**

Developer

97kemalozturk@gmail.com

ozturkkl@mail.uc.edu

832-520-1467

Responsibilities

* Develop code for the Link Tailor Application.
* Self-assign and create Kanban tasks to reflect work in-progress and completed.
* Present completed work at the Sprint Reviews and to class advisors.
* Estimate future work in the Sprint Planning sessions.
* Research features and technology that could improve the Link Tailor Application.
* Take on Project Manager roles in a smaller but meaningful way, including writing for project documents and facilitating meetings.

**Lauren Tillery**

Developer

tillerylauren@gmail.com

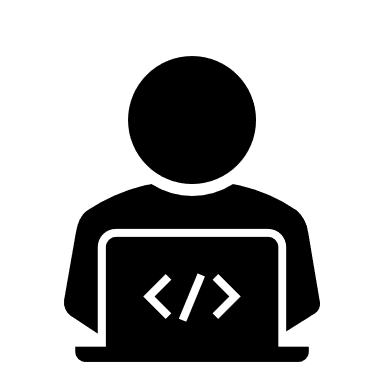
tillerlo@mail.uc.edu

513-240-6907

Responsibilities

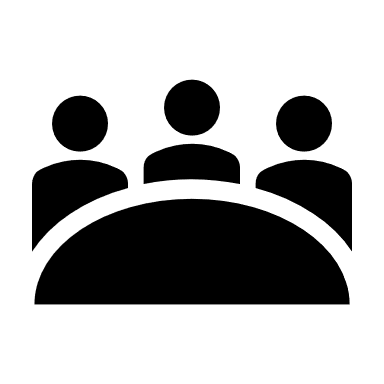
* Develop code for the Link Tailor Application.
* Self-assign and create Kanban tasks to reflect work in-progress and completed.
* Present completed work at the Sprint Reviews and to class advisors.
* Estimate future work in the Sprint Planning sessions.
* Research features and technology that could improve the Link Tailor Application.
* Take on Project Manager roles in a smaller but meaningful way, including writing for project documents and facilitating meetings.

## Team Values and Rules



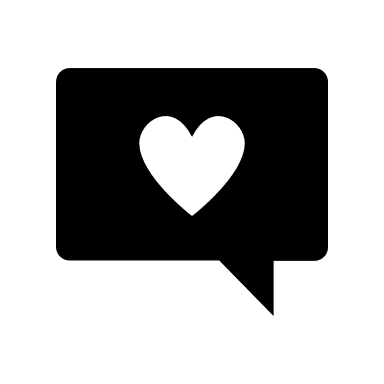
**Working Software**

Working hard to create a satisfying product.



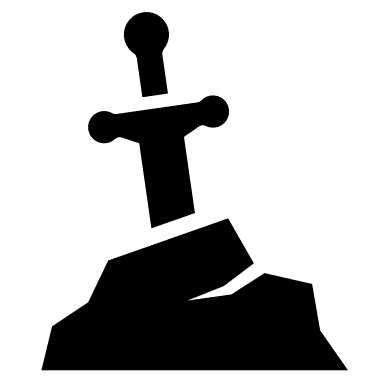
**Communication**

Frequent, effective, and honest communication.



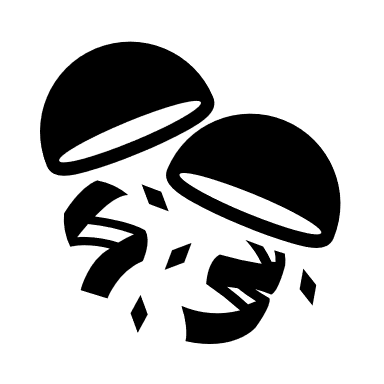
**Empathy**

Assuming the best of one another, treating each other with respect and care.



**Integrity**

Doing what is right, avoiding deceit, and owning up to mistakes.



**Celebrating the Try**

Our efforts are worthy of

recognition and celebration!

**Additional Rules**

1. Plagiarism will not be tolerated. Any team member that plagiarizes will be subject to university policies and a team meeting will be called.
2. Each team member will stay current on their tasks to ensure the project milestones are being met. If an event conflicts that will affect the completion of a deliverable, the team member will notify the other team members at least 24 hours in advance of the scheduled due date.
3. If a group member will be absent on class days or for an extended period of time, they will notify the other team members and the Instructors.
4. All team members are required to attend all scheduled meetings and provide updates to the acting project manager during the meeting. If a team member cannot make a scheduled meeting, they must notify all team members at least 4 hours in advance and provide an update via messenger or email.
5. All team members will review the oral presentation and final white paper.
6. All team members will respect the opinions and ideas of each team member, other students, and faculty.

## Communication Plan

**Meetings**

We plan to have 2-week long Sprints with the following recurring meetings:

10 minute **Stand-Up** meetings every Monday, Wednesday, and Friday.

* Concerns/impediments?
* Updates to share?

1 hour **Sprint Planning** meetings every Monday at the beginning of a new Sprint.

* Assign Story Points to tasks
* Determine acceptance criteria for each assigned task

Half-hour **Sprint Reviews** every Friday at the end of a Sprint.

* Review each completed task based on the acceptance criteria

Half-hour **Sprint Retrospectives** every Friday at the end of a Sprint.

**Kanban Board**

We plan to use a Kanban Board through GitHub Projects. It is expected that all team members use the board to update their work. The Kanban tasks will likely become the primary place for communicating feature and project requirements and acceptance criteria of work.

**Code**

In this project, it is always preferable to over-comment rather than under-comment. As a team of relatively inexperienced developers, we need all the help we can get from frequent, verbose comments. If after the code is fully tested, we decide some comments are unnecessary clutter and want to trim them up we can.

## Problems and Escalations

If a team member has a concern with another member, they must express their concern to that member. If the problem persists, all team members must discuss the issue with each other, in face-to-face or video discussion. If a resolution still cannot be reached, members of the team may then reach out to their Senior Design advisors/professors for advice and mediation.

# Project Breakdown

The project is the creation of a desktop application that functions as a hub for link management – one place from which to organize and open all regularly visited web links, device apps, and system files.

## Project Source

This project was conceived by Daniel Hickman, who found himself suffering from an over-abundance of bookmarks: more than 300, sorted into various folders such as “Games,” “Technology,” “Reading,” “Podcasts,” “Art/Music,” “Work,” “Recipes,” “D&D,” and “Return To.” Finding and organizing saved bookmarks was becoming an increasingly difficult and frustrating task. In investigating possible solutions such as Tagpacker, Pocket, Dewey bookmarks, Dragdis, Dropmark, and Station, all were found significantly lacking. None of these possible solutions allowed for easy adding of links to a spatially organized Daniel did end up using RocketDock and the Windows Start menu for better organization of file-system links and applications, but neither provided any improvement to the original concern of bookmark management.

Thus, the concept for an easy way to organize not only bookmarks, but also file system files/folders and application links was born! Daniel found two other Software Development majors who were passionate about the project idea, Kemal and Lauren, and together they brainstormed ways to create an incredible app, using the exciting new technologies of Vue.js and Electron.

## Project Examination

After realizing the need for a better link managing solution in our own computer use, the team investigated the current options that address this issue. They looked at relevant UX case studies to identify what other computer users found to be pain points in their link management experience.

**User Pain Points:**

* Scrolling to find newly saved links
* “losing” saved bookmarks (or not finding them quickly)
* Bookmark list feels not part of the application (like it is an overlay) (chrome)
* When space is out on the bookmark bar, other links get time-consuming to find & access
* No intuitive bookmark searching feature
* No intuitive way to see your last-saved bookmarks
* Waiting for load (raindrop.io)
* Too many links that they do not return to (Pinboard)
* Unable to use bookmarks for other browsers (No cross-browser syncing)
* Wanting to save their open tabs to return to later, but not crammed at the bottom of some list

The Link Tailor application is meant to be used alongside current solutions such as the Windows Start menu, Chrome Bookmarks, and the File Explorer Quick Access Menu. These all are well integrated into their space and allow for a quick, intuitive experience. Link Tailor will enhance the user experience with additional spaces to store links, which can be custom-located, custom-themed, and custom-accessed.

Ways Link Tailor can fill a niche these solutions do not:

**Windows Start Menu**

*Provides an easily accessible space (bottom-left click or windows key) for spatial grid-organization of links. Links can be added from the right-click menu. Web-links (bookmarks) have an unintuitive, 7-click process to be added and must first be saved as shortcuts. Files are also difficult to add. There is are no tags for links.*

Link Tailor will…

* Allow for multiple layouts and widgets (which can be in various locations on the screen)
* Allow for more intuitive, quicker addition of web links and files (both of which require a tedious process to add to the start menu).
* Allow for the tagging of links.
* Allow for greater customization of the look and feel of layouts.

**Raindrop.io**

*The current best overall free bookmarking option. It sorts web links (and pictures and videos) into “collections” and by “tag.” Collections can be displayed as lists or grids. It has a Chrome extension that makes adding new bookmarks and accessing them very easy. There is a paid upgrade with additional features.*

Link Tailor will…

* Let the user save of applications, files, and folders (not just websites & images).
* Allow for layouts to be arranged spatially (Raindrop.io lets a list be displayed in grid-format, but not to visually group the links or set the layout to be consistent across screen sizes).
* Visually theme based on tags (which is a requested feature of some users).
* Allow for creation of widgets on the desktop (Raindrop has a desktop app but no widgets).

**Rocket Dock**

*Provides a widget which can placed in a custom location on the screen and holds files (and folder paths). It can be accessed by mouse-movements over a certain location of the screen. Web-links can only be added by first saving the link as a file.*

Link Tailor will…

* Allow for a spatial layout (not just linear).
* Let users create multiple widgets (not just one).
* Make the addition of new links much easier (quicker).
* Be accessible by hotkey.

We envision Link Tailor would work very well for users who wish to have layouts specific to particular projects (work or school). Users could quickly tag new links with a project name so the addition of new links to the project is easy and works cross-layout. They could also create a widget for the project with all the frequently-accessed links (web, application, and files) for the project. This could be an effective way to organize links for dozens of projects.

We envision some users will choose to use Link Tailor just for the ability to create multiple widgets in custom locations on the screen. They may opt to organize in ways like: social media links on the right, coursework links on the bottom, notes/reminders on the left, and fun/casual links on the top.

We envision some users will use Link Tailor as a simple solution for linking that does not require much work to organize. They will not create widgets or do much management of their layouts, but they will make use of the ability to make and tag new links, appreciating that the links are visually distinctive based on the tags. Features such as saving all open tabs in one category or opening all links in a category will allow users to quickly “dump” large numbers of links to return to later without having to micro-manage them.

## Project Goals

**Functionality**: Links can be added and organized.

* Web-links, files, folders, and application links can all be added and arranged spatially.
* Next-level functionality is bulk saving and opening links, setting links to run in a particular program, as admin, or with other specifications.

**Convenience**: Quick to use.

* Measured by how many clicks/key-presses it takes to perform an action (such as adding a link). We want to minimize the steps it takes!
* This may mean creating browser extensions or Windows-integrations.

**Aesthetics**: The app looks good.

* Measured by user testing. Users will look at their layouts a lot so we want them to look awesome!

**Guidance**: Helps build good workflow habits.

* Measured by how little users have to change or think about for the app to be useful.
* Users who will not invest much time in the app must be given the best link-managing options by default and helped with suggestions (or default templates) of how to tag or what kinds of layouts or widgets to create.

**Freedom**: Users are able to customize their experience.

* Measured by users being able to change the look and feel of their layouts easily. This might be done through allowing many variables to be changed by users very easily. This may also be done through providing more advanced users with the ability to craft and publish various designs.

**Sharing**: The ability to share design themes and link layouts with others.

* Measured by how easy and intuitive it is to export/import layouts.

**Website**: A space for users to share their layouts and download what others have created.

* Will promote community innovation around the app.
* If successful, new users will be able to browse and download beautiful themes and layouts.

**Data Recovery**: If a user’s PC is irrecoverable, they can load their link layouts on their new device.

* The first level is allowing for the importing and exporting of layout files within the app.
* The second level is creating an auto-back-up feature.

**Device Syncing**: A user’s change to a layout on one device will update a synced layout on another device.

## Project Scope

We, in using an Agile approach, will create user stories, assign story points to them, track our team’s velocity, and adjust our scope accordingly. Below are a list of features that are currently within scope, roughly organized in high-to-low priority.

**App Features**

* Layouts with spatial organization and drag-and-drop functionality
* Tagging functionality with stylistic differences based on tags
* Link Searching
* Widgets with custom locations and ability to summon based on mouse movements
* Extra options for opening links (admin, open with…)
* Themes & detailed customizations (color/shape/size)
* Custom hotkeys
* Importing/Exporting link layouts
* Chrome extension (for easier web-link adding and accessing)
* Windows-integration (adding “New link…” to the right-click menu)
* Detailed searching filters
* Syncing between devices
* Mobile compatibility (beginning with Android)
* An easy way for others to contribute themes, customizations, etc (& share them with others)
* Custom notifications, rotating messages, reminders, and triggers

## Project Timeline

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Name** | **Duration (Days)** | **Start Date** | **End Date** |
| **1.0 Project Management and Deliverables** | 233 | 8/24/2020 | 4/13/2021 |
| 1.1 Team Building | 7 | 8/24/2020 | 8/31/2020 |
| 1.2 Project Idea Brainstorming | 7 | 8/24/2020 | 8/31/2020 |
| 1.3 Fall Semester Assignment 0: Team Members and Project Name | 1 | 8/24/2020 | 8/24/2020 |
| 1.3.1 Project Name | 1 | 8/24/2020 | 8/24/2020 |
| 1.3.1 Project Logo and Branding | 1 | 8/24/2020 | 8/24/2020 |
| 1.4 Fall Semester Assignment 1: Team Contract | 7 | 8/24/2020 | 8/31/2020 |
| 1.4.1 Project Approval | 7 | 8/24/2020 | 8/31/2020 |
| 1.4.2 Quick Project Timeline | 7 | 8/24/2020 | 8/31/2020 |
| 1.5 Fall Semester Assignment 2: Project Abstract for Tech Expo | 29 | 9/14/2020 | 10/12/2020 |
| 1.6 Fall Semester Assignment 3: Team Contract Resubmission | 7 | 10/05/2020 | 10/12/2020 |
| 1.7 Fall Semester Assignment 4: User Profile | 7 | 10/12/2020 | 10/19/2020 |
| 1.8 Fall Semester Assignment 5: Use Case Diagram | 7 | 10/12/2020 | 10/19/2020 |
| 1.9 Fall Semester Assignment 6: Draft Report | 22 | 10/19/2020 | 11/9/2020 |
| 1.10 Fall Semester Assignment 7: Final Report | 22 | 11/9/2020 | 11/30/2020 |
| 1.11 Fall Semester Oral Presentations | 29 | 11/2/2020 | 11/30/2020 |
| 1.11.1 Practice Presentation | 29 | 11/2/2020 | 11/30/2020 |
| 1.12 Spring Semester | 93 | 1/11/2021 | 4/13/2021 |
| 1.13 IT Expo | 1 | 4/13/2021 | 4/13/2021 |

|  |  |  |  |
| --- | --- | --- | --- |
| **2.0 Research** | 43 | 9/7/2020 | 10/12/2020 |
| 2.1 Software Requirements | 7 | 9/7/2020 | 9/14/2020 |
| 2.1.1 Determine Front End Development Language | 7 | 9/7/2020 | 9/14/2020 |
| 2.1.2 Determine Back End Development Language | 7 | 9/7/2020 | 9/14/2020 |
| 2.2 Miscellaneous Research | 7 | 9/14/2020 | 10/5/2020 |
| 2.2.1 Average Number of Websites Booked Marked | 7 | 9/14/2020 | 10/5/2020 |
| 2.2.2 Conduct Interviews with Mock Stakeholders | 7 | 9/14/2020 | 10/5/2020 |
| 2.2.3 Budget Analysis | 7 | 10/5/2020 | 10/12/2020 |

|  |  |  |  |
| --- | --- | --- | --- |
| **3.0 Environment Set-Up** | 14 | 9/14/2020 | 9/28/2020 |
| 3.1 Install Libraries for Development | 7 | 9/14/2020 | 9/21/2020 |
| 3.2 Install Electron | 7 | 9/14/2020 | 9/21/2020 |
| 3.2.1 Setup Electron | 7 | 9/14/2020 | 9/21/2020 |
| 3.3 Setup Vue.js | 7 | 9/14/2020 | 9/21/2020 |
| 3.3.1 Install and Configure | 7 | 9/14/2020 | 9/21/2020 |
| 3.4 Setup Github | 7 | 9/21/2020 | 9/28/2020 |
| 3.5 Create Kaban Board | 7 | 9/21/2020 | 9/28/2020 |
| 3.5.1 Create Project Tasks | 7 | 9/21/2020 | 9/28/2020 |
| 3.5.2 Assign Tasks | 7 | 9/21/2020 | 9/28/2020 |

|  |  |  |  |
| --- | --- | --- | --- |
| **4.0 Software Design** | 21 | 9/28/2020 | 10/19/2020 |
| 4.1 Create System Diagrams | 7 | 9/28/2020 | 10/12/2020 |
| 4.1.1 Create Network Diagrams | 7 | 9/28/2020 | 10/12/2020 |
| 4.1.2 Create Database Diagrams | 7 | 9/28/2020 | 10/12/2020 |
| 4.1.3 Create Wireframe Diagrams | 7 | 9/28/2020 | 10/12/2020 |
| 4.2 Create Legal Documentation | 7 | 10/12/2020 | 10/19/2020 |
| 4.2.1 Draft Legal Disclaimers and Privacy Policy | 7 | 10/12/2020 | 10/19/2020 |

|  |  |  |  |
| --- | --- | --- | --- |
| **5.0 Development (Back End and Front End)** | 120 | 10/5/2020 | 2/1/2020 |
| 5.1 Create Main Page Grid View | 14 | 10/5/2020 | 10/19/2020 |
| 5.2 Create Navigation Side Bar | 14 | 10/5/2020 | 10/19/2020 |
| 5.3 Create Buttons | 14 | 10/5/2020 | 10/19/2020 |
| 5.3.1 Add Link | 14 | 10/5/2020 | 10/19/2020 |
| 5.3.2 Create View | 14 | 10/5/2020 | 10/19/2020 |
| 5.3.3 Edit Link | 14 | 10/5/2020 | 10/19/2020 |
| 5.3.4 Add Folder | 14 | 10/5/2020 | 10/19/2020 |
| 5.3.5 Settings | 14 | 10/5/2020 | 10/19/2020 |
| 5.5 Create Search Bar | 14 | 10/5/2020 | 10/19/2020 |
| 5.5.1 Configure Filtering | 14 | 10/5/2020 | 10/19/2020 |
| 5.6 Design and Develop UI Color Scheme | 30 | 10/19/2020 | 11/19/2020 |
| 5.6.1 Design Multiple Themes | 30 | 10/19/2020 | 11/19/2020 |
| 5.7 Develop App Features | 92 | 11/2/2020 | 2/1/2021 |
| 5.7.1 Develop Notification Functionality | 21 | 11/2/2020 | 11/23/2020 |
| 5.7.2 Develop Syncing Devices Functionality | 71 | 11/23/2020 | 1/11/2021 |
| 5.7.3 Develop Toggling Functionality | 22 | 1/11/2020 | 2/1/2021 |

|  |  |  |  |
| --- | --- | --- | --- |
| **6.0 Testing and Refining** | 30 | 2/1/2021 | 3/1/2021 |
| 6.1 Functionality Test | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.1 Add Link | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.2 Add Folder | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.3 Create View | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.4 Edit/Remove Links and Folders | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.5 Change Between Views | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.6 Choose Theme | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.7 Search Bar with Filtering | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.8 Open Links with Hot Keys | 30 | 2/1/2021 | 3/1/2021 |
| 6.1.9 Syncing Between Devices | 30 | 2/1/2021 | 3/1/2021 |
| 6.2 Perform User Acceptance Test | 30 | 3/2/2021 | 4/2/2021 |
| 6.2.1 Create Multiple Files, Folders and Views | 30 | 3/2/2021 | 4/2/2021 |
| 6.2.2 Conduct Usability Test | 30 | 3/2/2021 | 4/2/2021 |

## Technologies Used

**NPM**: Node environment has countess packages to support fully built applications ranging from file operations to user interface libraries. We will be able to utilize node package manager in the application.

**Electron**: Electron is an open-source framework that enables creating cross-platform applications using web technologies. It uses Chromium engine for the UI along with Node.js runtime for the back-end.

**JavaScript**: JavaScript programming language will be used to develop the application through node and electron.

**HTML**: Since the app is using Chromium engine for its front end. HTML will be used to map out the UI elements.

**CSS**: The styling of the UI elements will be set using Cascading Style Sheets.

**Vue.js**: Vue will be used to create the front-end as components to improve modularity and performance.

**SQLite**: For the storage and back-end database needs SQLite will be used. This is a great database solution that works amazing with Electron.

**Visual Studio Code**: A popular code editor made by Microsoft.

# Team Signatures

Signature: Daniel Hickman Date: 9/1/2020

**Daniel Hickman**  Project Manager

Signature: Kemal Ozturk Date: 9/1/2020

**Kemal Ozturk**  Developer

Signature: Lauren Tillery Date: 9/1/2020

**Lauren Tillery**  Developer

1. L’Angelier, Eliot. (2019) *Google Chrome bookmarks – a UX case study*. Medium. <https://uxdesign.cc/google-chrome-case-study-bookmarks-83a32b7c754> [↑](#endnote-ref-1)
2. Descy, Michael. (2018) *Re-committing to Pinboard, after many months away*. Snippets. <https://mjdescy.me/2018/07/10/re-committing-to-pinboard-after-many-months-away/> [↑](#endnote-ref-2)
3. Dubroy, Patrick. (2009) *How many tabs do people use? (Now with real data!)*. dubroy.com. <https://dubroy.com/blog/how-many-tabs-do-people-use-now-with-real-data> [↑](#endnote-ref-3)
4. Alsop, Thomas. (2020). Computer penetration rate among households worldwide 2005-2019. Source/Publisher: ITU. Release Date: November 2019 <https://www.statista.com/statistics/748551/worldwide-households-with-computer/> [↑](#endnote-ref-4)
5. Statista Research Department. (2015) Number of websites visited by new and established internet users in the UK 2014. <https://www.statista.com/statistics/322946/number-of-websites-visited-by-new-and-established-internet-users/> [↑](#endnote-ref-5)
6. Fyodor. (2019) *How do you manage your browser bookmarks?* Dev. <https://dev.to/fyodorio/how-do-you-manage-your-browser-bookmarks-1hcb> [↑](#endnote-ref-6)