



COMP 3020 Project Milestone 3

Presented to: Dr. Patrick Dubois

COMP 3020 A02, Human Computer Interaction 1, Fall 2023

By: Team 27

Nguyen, Khuc (nguye64@myumanitoba.ca)

Khangura, Jaskaranjot (khangur2@myumanitoba.ca)

Bian, Fengfan (bianf@myumanitoba.ca)

Sanyasi, Nishan (sanyasin@myumanitoba.ca)

A. Technology Review:

We decided to use pure HTML, CSS, and JavaScript for this project. The main reason behind this is because firstly, we don't want to be refrained by any framework and secondly, we don't want to add another learning layer for our team members as everyone doesn't have that much experience with JavaScript.

Our project employed a tree like book view structure. We implemented that using d3.js version 5 library. D3.js is a free open-source JavaScript for data visualization. D3.js has a built-in tree structure and we leveraged that to make the tree view in our webpage. To facilitate the tree structure for the library, the back-end data was also implemented in a tree like structure. Every tree starts with the department as a root node, then branches down to year, to courses and to the actual book. When the search page receives a query for a book, it finds the root node of that book and reconstructs the entire tree by iteratively travelling downward until all branches are included, encoded as JSON in local storage. The tree page then fetches this from the local storage and renders a hierarchical tree. D3.js is also used to make drag and drop interaction for the tree page. Users can drag a book to the cart to add it to cart. We check if a node is a leaf node (meaning it is a book) and add this functionality through .append() in D3.js.

As we wanted our website to support multiple users (meaning each user has their own data), we leveraged local storage. On sign up, every user will be stored as a key – value pair in local storage. This makes it possible to encapsulate user data and make sure one user cannot modify other user data. We also use local storage to communicate across pages. The previous page will save its information in local storage and the destination page will fetch that data.

B. Design Deviation and evolutions:

These are the changes from the paper prototype:

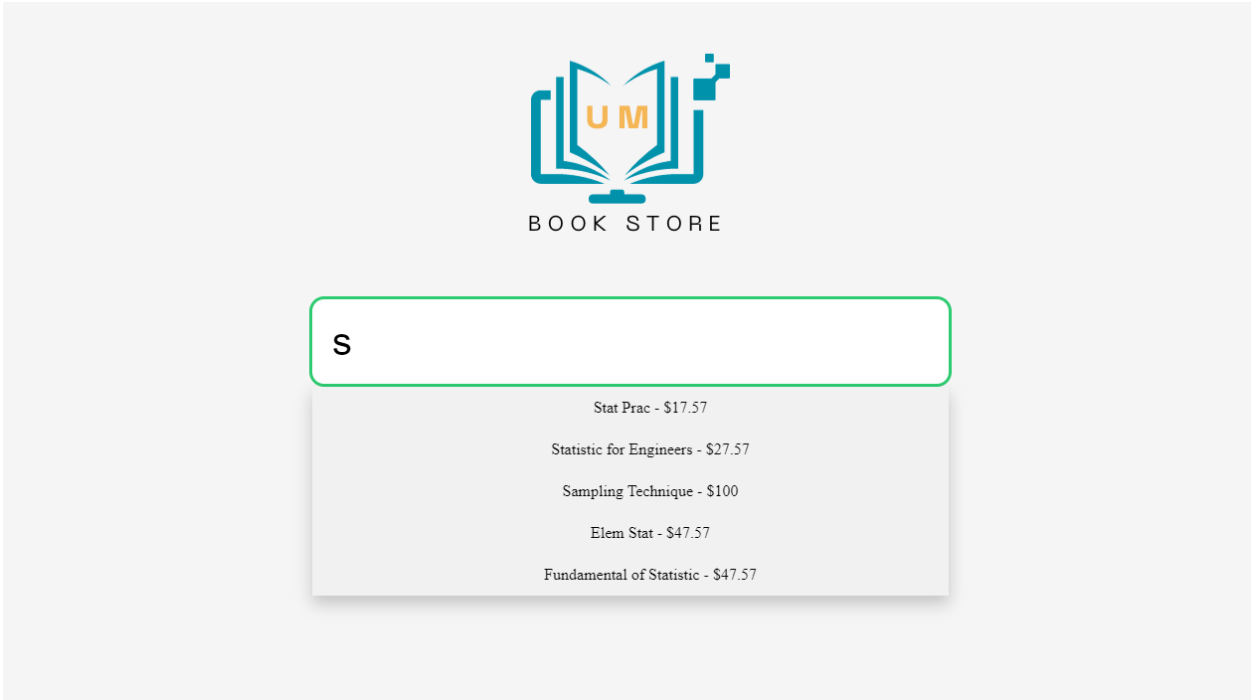
- + We don't allow multiple trees. Many interviewers' comment that having multiple trees is confusing as too much data is being shown on the screen. Because of that, only one tree is shown at any given time.
- + Only one branch can be expanded at a time. We found that having the tree entirely expanded is not helpful. Having too many nodes makes the tree cluttered and unreadable. It also makes clicking and dragging a specific node difficult.
- + There is no right click menu for node anymore. Initially, the right click menu is so that users can choose which children node to show. We received feedback that this is not intuitive and hard to use. The users now can only expand the tree, one node at a time by double clicking the node. This restriction makes it easy to navigate the tree and helps the tree display more clearly and spaced out.

- + The leaf node (book node) doesn't display the book cover anymore. We couldn't figure out a correct way to display the book cover (and other buttons as shown in the prototype) without breaking something else. Having the book cover in the node also makes the tree larger than necessary and the tree will get cut off. We moved this to an overview panel. When a user double clicks on a book node, book cover, title and other relevant information will be displayed in the overview panel.
- + We combined the overview and book preview panel into one. The overview starts with a book cover, book information, a summary of the book and some of the first content line as text. We decided not to use image for book preview as the image is too small to be readable in the panel.
- + Node cannot be relocated anymore. If user relocated a node and then forget where it is, it would be difficult to find that node. The entire canvas can still be moved but only in the case when the full tree cannot be displayed on the screen.
- + We couldn't implement drag and drop from the department search panel to the canvas for tree populating as it is too difficult to do this using D3.js.

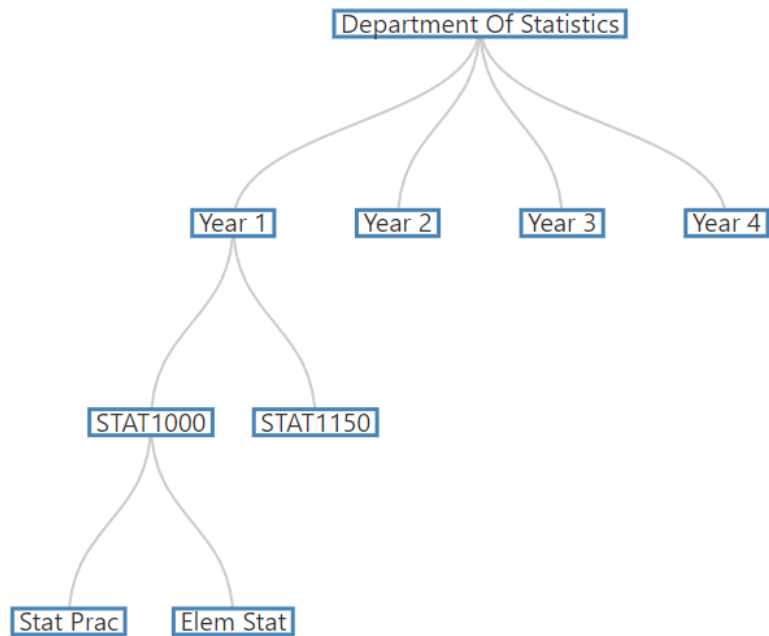
C. Sale pitch

+ Simplicity

We designed our interface to be simple and easy to use. We make it so that one page only serves a certain function so overall, no single page is overly complicated or too cluttered. Searching for the book is efficient as you only need 2 steps: typing the search query in the input field of the search page and click search. The book will be revealed in the tree structure as a leaf node. Adding a book into cart is also intuitive and fast. Users can either drag and drop a book to the cart area or use the add button in the overview section. Because of that, adding a book is a one step only operation. Users can also change the current department tree by using the search panel by clicking on a department name and then click search, which is a 2-step procedure. The tree can be easily explored by double clicking on any parent node. Overall, this design makes searching and exploring books more interesting and efficient thanks to how simple it is. Books can also be easily removed from cart either on the check-out page or in the tree page. In the check-out page, the user can either adjust the quantity of the book using an input field and remove the book by clicking the red "trash bin" icon. Alternatively, they can remove the book from cart by clicking on the book title in the cart panel of the tree page.



Search with suggestion



Pre-expanded Tree

Overview

★ ★ ★ ★ ★

FOR THE AP[®] EXAM

The PRACTICE
of STATISTICS

FIFTH EDITION

STARNES • TABOR • YATES • MOORE

Title: Stat Prac

Author: Moore

Version: 5

ADD

Summary: Lorem ipsum dolor sit amet. Aut
galisum voluptatem eos quis eveniet eum dolores
internos est illo quae quo facere voluptate. Aut
error maxime sed internos placeat qui dolorem
rerum non impedit distinctio. Est harum deleniti
aut maxime exercitationem et impedit nostrum et
ipsum quisquam. Ad natus autem eum autem


Overview Section with add button

X

Shopping Cart

Stat Prac Qty: 1

Remove from cart panel in tree page. A red line will appear.


Search

Department/Root node search

Shopping Cart



Stat Prac

Version: 5

Price: \$17.57

Quantity:



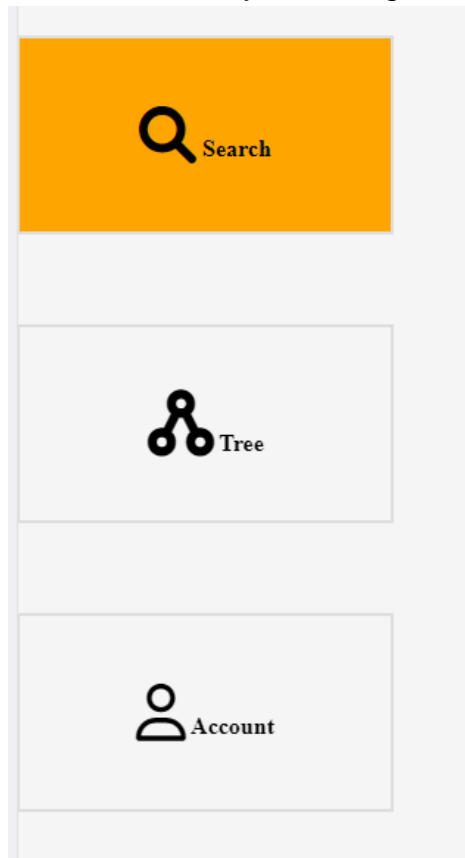
Price Summary

GST:	0.88
PST:	1.23

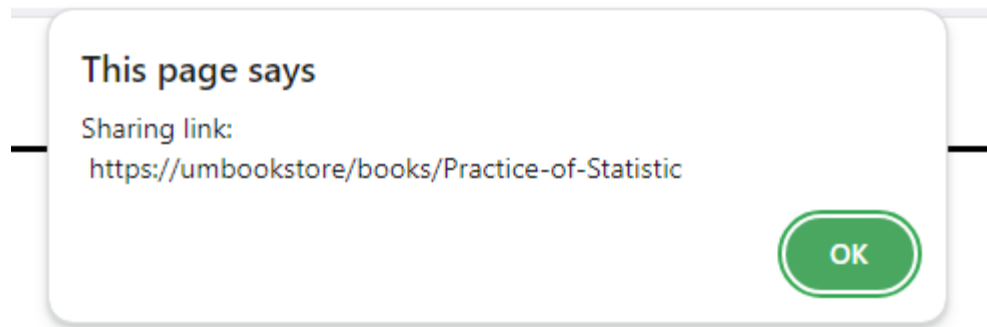
Remove item from the shopping cart using the trash bin icon.

+ Feedback

We provide feedback in many places throughout the interaction. We implement a responsive navigation bar where button color differs based on where the user is. Thanks to this, the user knows where they are in the interface just by looking at which button is colored differently in the navigation bar. The user can click on the navigation bar to move to a new page and the buttons' color will reflect the change. Moreover, we provide instant responses for every interaction. For example, if a user shares a book, a pop up with the book link will immediately show up on top of the screen. When a user changes their profile, a similar pop up will inform the user that their information has been successfully changed. Similarly, a pop up for order verification will also appear when user places their order successfully. Even when the user searches for a non-existent book, we will show an empty tree to indicate that their search was not successful. As such, we ensure that our interface lets users know where they are and what they are doing.



Navigation bar (currently on search page)



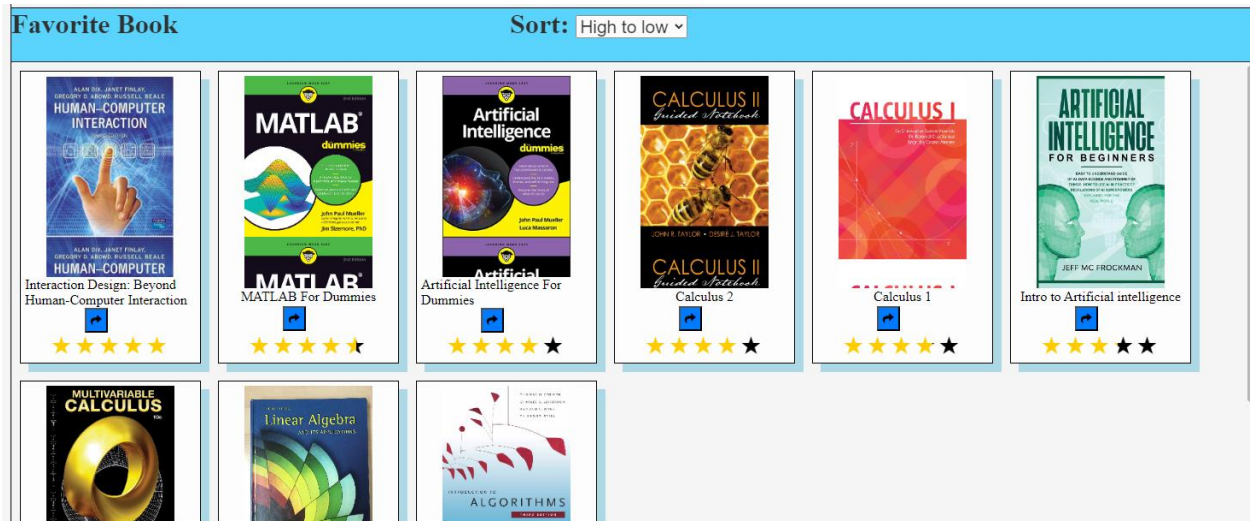
Share pop-up

+Utility

Aside from the book searching and purchasing feature, our interface provides several other functions. We allow each user to have their own account. Users can have their own profile and list of favorite books. The profile can be changed freely. Our gender section is not binary so that the interface is more inclusive. Users can also change their password if they know their previous password, with some error checking and verification. Users can rate a book from the tree pages or from the account page and the change will be reflected immediately. Users rate a book by clicking (or dragging) the 5 stars. There is also a sorting option under the account page. Users can sort their list of books based on rating (high to low or low to high) or by alphabetic order. As mentioned in the previous section, user can easily share a book and a link will pop up on top of the screen.

A screenshot of a web form titled "Account Information" in a blue header bar. The form is divided into two columns. The left column is titled "Profile" and contains fields for "Username:" (with the value "Rob"), "Gender:" (with a dropdown menu showing "Male"), and "Birthdate:" (with the value "12/29/2011" and a calendar icon). Below these fields is a "Change Profile" button. The right column is titled "Password" and contains three input fields: "Current Password", "New Password", and "Confirm New Password". Below these fields is a "Change Password" button.

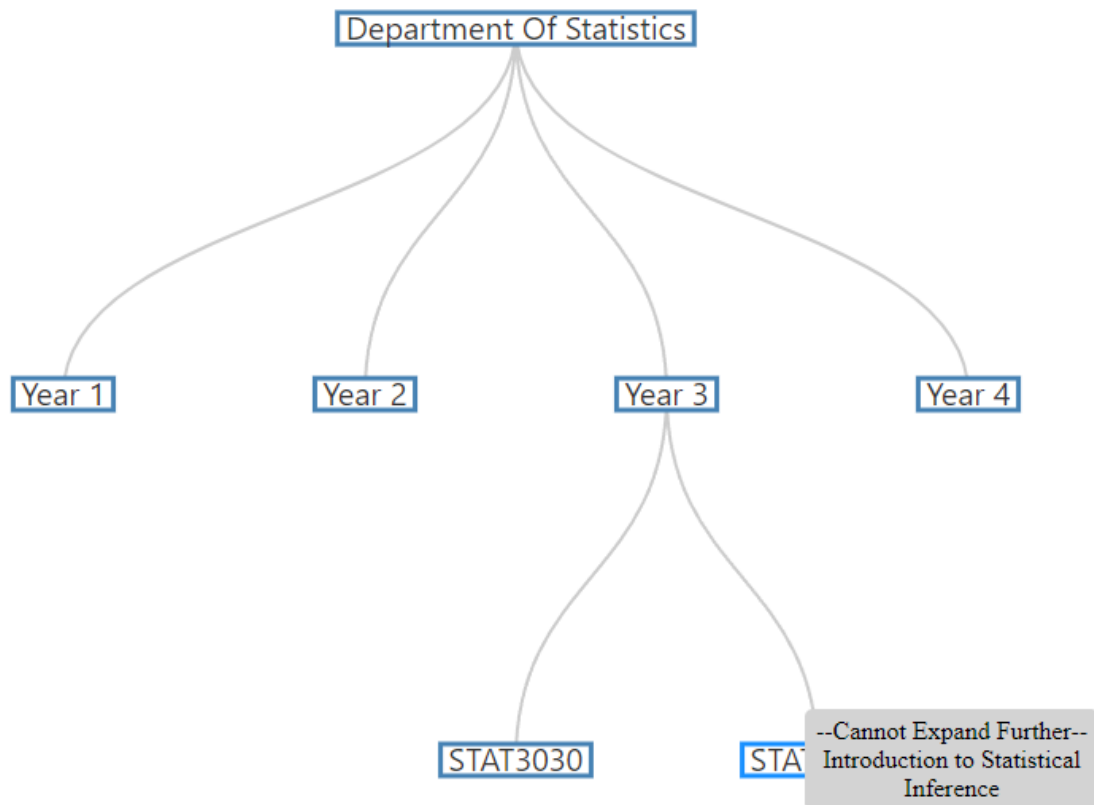
Profile Section



Favorite book section. Each book can be rated by clicking on the 5 stars icon.

+Constraint

We provide some constraint to user input to avoid error. For instance, when searching for a book, user will be provided with a list of suggestions, that is every book that match the query in the system, and we encourage the user to uses that information (admittedly, this is not much of a constraint). However, if a user tries to search for a non-existed book, we still produce an empty tree (for feedback). When a user tries to explore the tree, there are some parent nodes that is not expandable. We attach a “—Not Expandable Anymore —” note to those nodes when the user hover mouse on it. When trying to modify Date of birth, we restrict the input using a mask MM/DD/YYYY. Alternatively, users can use the calendar to select their birthday.



Cannot expand further tooltip.

Account Information

Profile

Username:

Gender: ▼

Birthdate:

Date of birth input is restraint to the MM/DD/YYYY format.

+ Consistency

The pages design is consistent. Every page has a header containing “Hello <User>” text and a sign out button. On the left-hand side is a navigation bar with different color buttons depending on where the user is in the interface. This makes it easy to remember and navigate the interface.

D. Appendix

+ Work distribution:

Nguyen, Khuc: Data preparation, Full responsibility of Account Page, localStorage implementation, Tree page implementation, Coordinator.

Khangura, Jaskaranjot: Tree’s interaction designer (node expansion, node drag and drop), Full responsibility of Tree page design.

Bian, Fengfan: Full responsibility of SignUp and Login page, localStorage implementation, responsive nagivation bar,

Sanyasi, Nishan: Data preparation, Full responsibility of Search and Check-out page design, Tree implementation, overall interface layout design.

+ What doesn’t work and what changes(cont):

+The tree graph is smaller than we expected and D3.js is making it so hard to make this display correctly. While the tree still renders fine, if you try to drag a node out of the graph, the node will disappear (but it can still be added to the cart this way though).

+ We decided to simplify the item in the shopping cart for the Tree page. We remove the book cover and the 2 button to change the quantity. The only option for an item in the cart at this point is to remove it. We did this because we want user to focus more on exploring instead of spending too much time adjusting their cart.

+ Some part of the interface is not responsive.

+ About the tooltip for parent node. Ideally, we wanted to grey out node that is not expandable. However, for some unknown reason, it makes the book nodes unclickable. Because of that, we settle with a tooltip.

