

# Project Report

---

## Project title

Gift Recommendation System

## Project Goal

This website will recommend perfect gifts to users, depending on their preferences, and will make their experience an easy one. It will help them compare the availability of products on ecommerce websites such as Amazon, Macy's etc., thereby getting to pick the website of their choice, based on what suits them best.

## Description

Many-a-times, we face situations where in we want to gift our family or friends on their birthdays or special occasions, and face difficulty in making a decision regarding what to buy. Our website will make this decision simple for such users. Users will enter basic information about the person such as their age, gender, and location. Users can also enter a budget range, expected delivery date, the occasion for which they want to gift, and their relationship with the recipient.

Our system will then analyze these criterions and display a list of items that the users can buy. Users can go through each option and finalize on a particular product. Once the users pick an item, our website will generate different available options on different e-commerce websites. The website will present these items in a way that users can choose from multiple e-commerce websites, scroll through these products and pick the one they like the best.

Once the users select a particular product, they will be directed to the respective e-commerce website to purchase the product. Hence, the goal of our website is to provide a recommendation of the perfect gift suiting all the user needs and choices, in a centralized fashion.

## Client/target audience

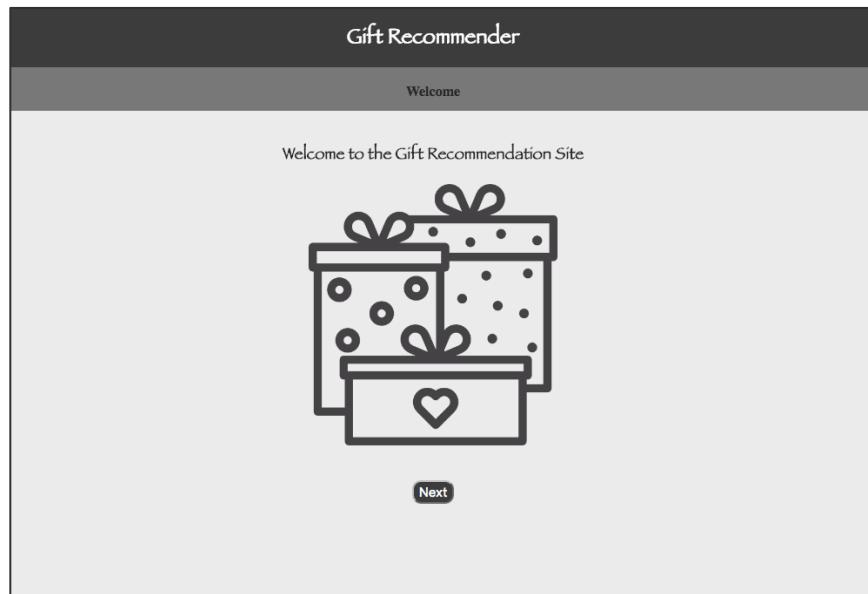
Target audience: Users who spend too long on e-commerce websites looking for gifts to buy for their family or friends can use our website to make the process faster and efficient.

## Team members

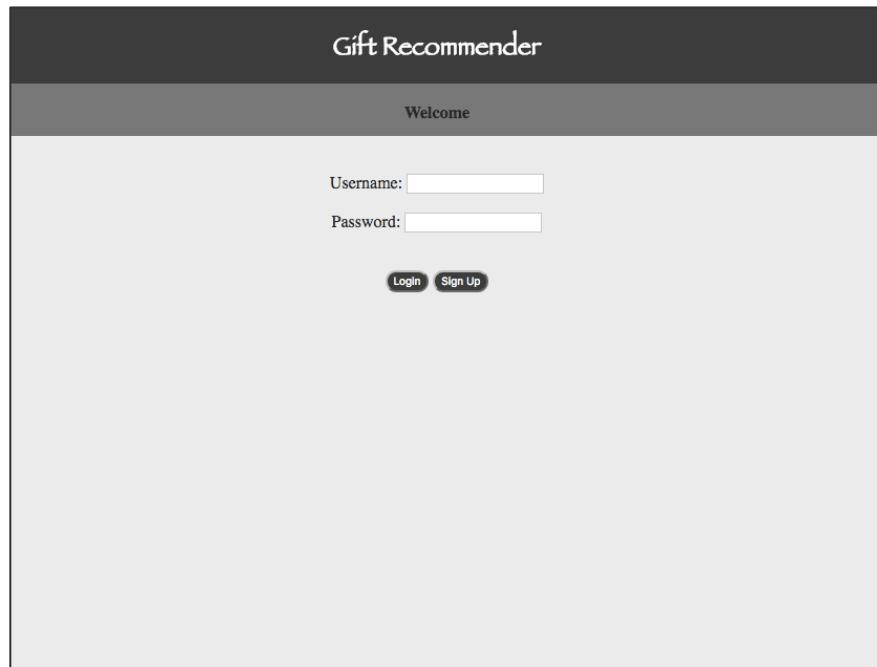
Name	Email
Karishma Ghiya	<a href="mailto:kghiya93@umd.edu">kghiya93@umd.edu</a>
Anushka Kher	<a href="mailto:akher@umd.edu">akher@umd.edu</a>
Pujita Tipnis	<a href="mailto:ptipnis@umd.edu">ptipnis@umd.edu</a>

## Required capabilities

1. Screen 1: Login page for users to log in to the website. Provides a signup button for first-time users to register with the website's services

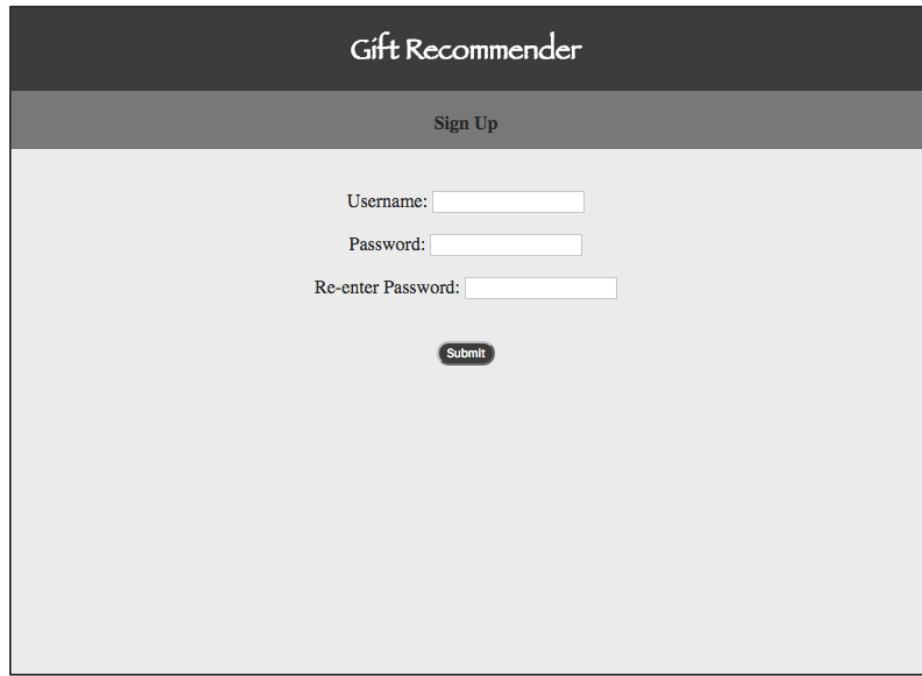


Img: Welcome Page (index.html)



Img: Login Page (login.html)

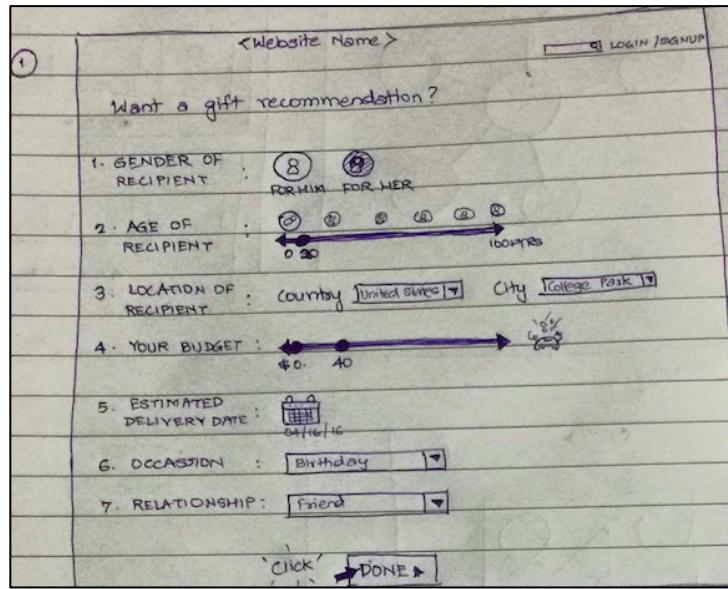
2. Screen 2: Sign up page for first-time users to sign up for the website's services



The image shows a 'Sign Up' page for a service called 'Gift Recommender'. The page has a dark header with the service name. Below it, a grey bar contains the word 'Sign Up'. The main area has three input fields: 'Username:' with a text input box, 'Password:' with a text input box, and 'Re-enter Password:' with a text input box. A 'Submit' button is located below the password fields.

Img: Sign Up Page (signup.html)

3. Screen 3: Website's screen providing filter options



The image is a hand-drawn storyboard of a filter options screen. At the top, it says '<Website Name>' and 'LOGIN / SIGNUP'. Below that, there's a question 'Want a gift recommendation?'. The screen is divided into seven numbered sections: 1. GENDER OF RECIPIENT (radio buttons for male and female, with 'FEMALE FOR HER' handwritten next to female), 2. AGE OF RECIPIENT (a horizontal scale from 0 to 100 with marks at 20 and 100, with '20' handwritten near 0 and '100+ yrs' near 100), 3. LOCATION OF RECIPIENT (Country: United States, City: College Park MD), 4. YOUR BUDGET (a horizontal scale from \$0 to \$100 with marks at \$0 and \$100, with '\$0' handwritten near 0 and '\$100' near 100), 5. ESTIMATED DELIVERY DATE (calendar icon showing 04/16/16), 6. OCCASION (dropdown menu set to 'Birthday'), and 7. RELATIONSHIP (dropdown menu set to 'Friend'). At the bottom, there are buttons labeled 'Click' and 'DONE'.

Img: Storyboard image of the filter options screen

This is our fourth screen. It provides the user with options to pick from, which are relevant to gift recommendation. Here, the user has selected gender as female, age as 20 using a scale, country as USA, city as College Park. The budget scale allows the user to pick min and max values so user can provide a range.

**Gift Recommender**

Want a gift recommendation?

\* Gender of recipient :  ♂  ♀

Age of recipient :

\* Age range: 5 - 15

Select Country :

Select State :

Your Budget :

Price range: \$75 - \$300

Estimated delivery date: mm/dd/yyyy

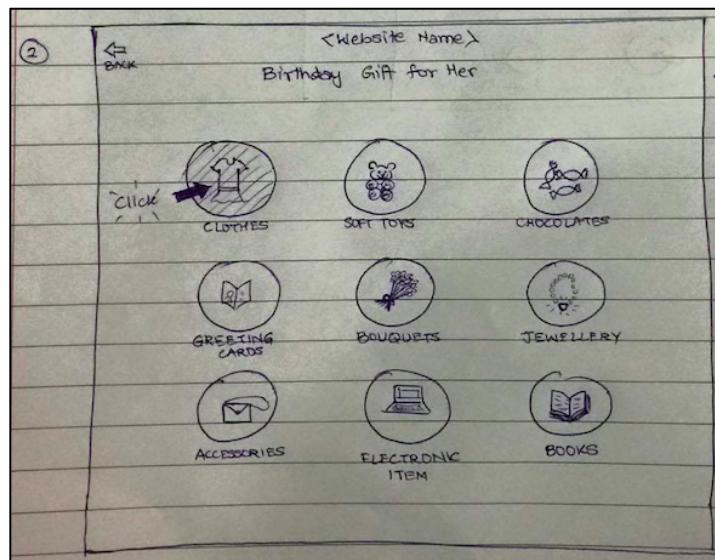
\* Occasion :

Relationship :

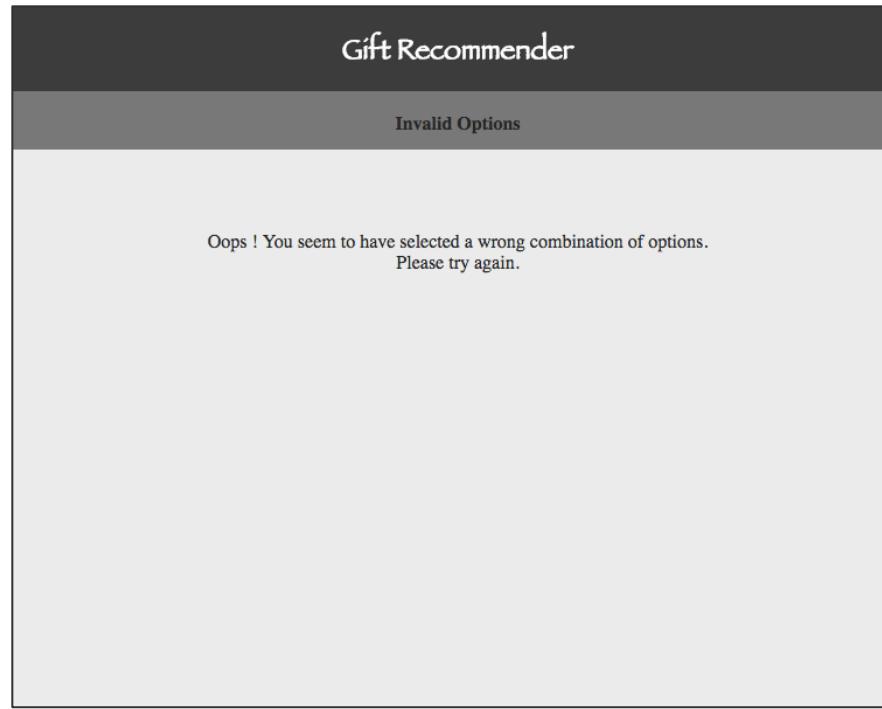
**SUBMIT**

***Img:*** Filter options for gift recommendation (filter.html)

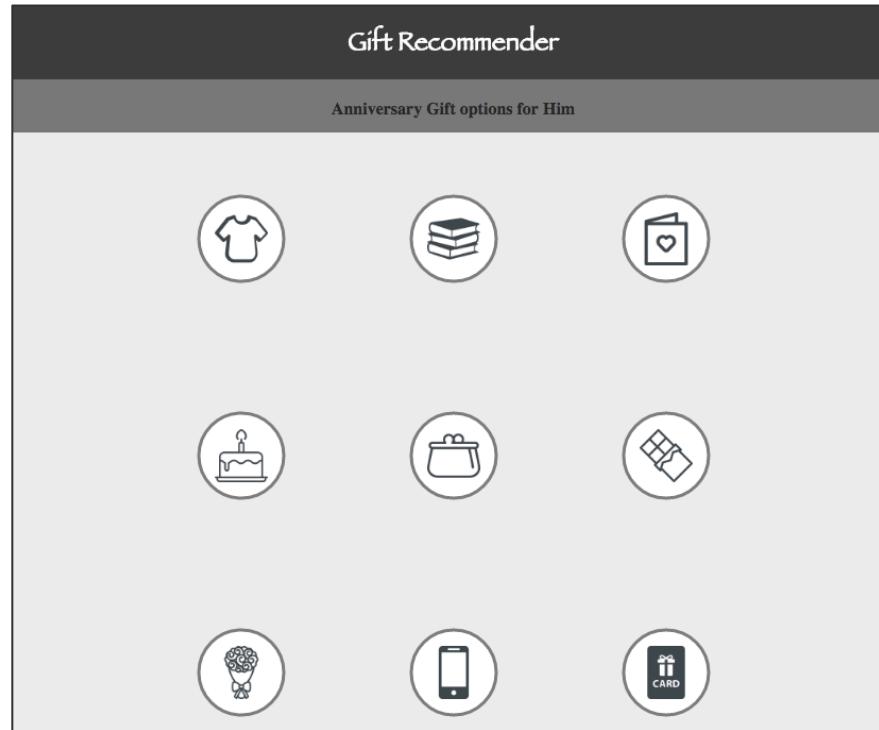
4. Screen 4: Screen providing users with product categories to pick from, based on filter selections made on previous page



***Img:*** Storyboard image showing gift category options



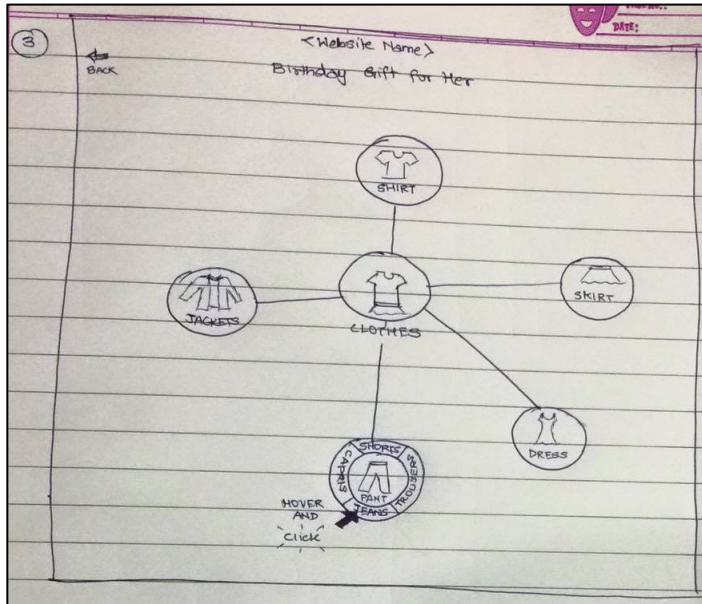
**Img:** Screen displaying error when user selects invalid combination of options  
(showRecommendation.html)



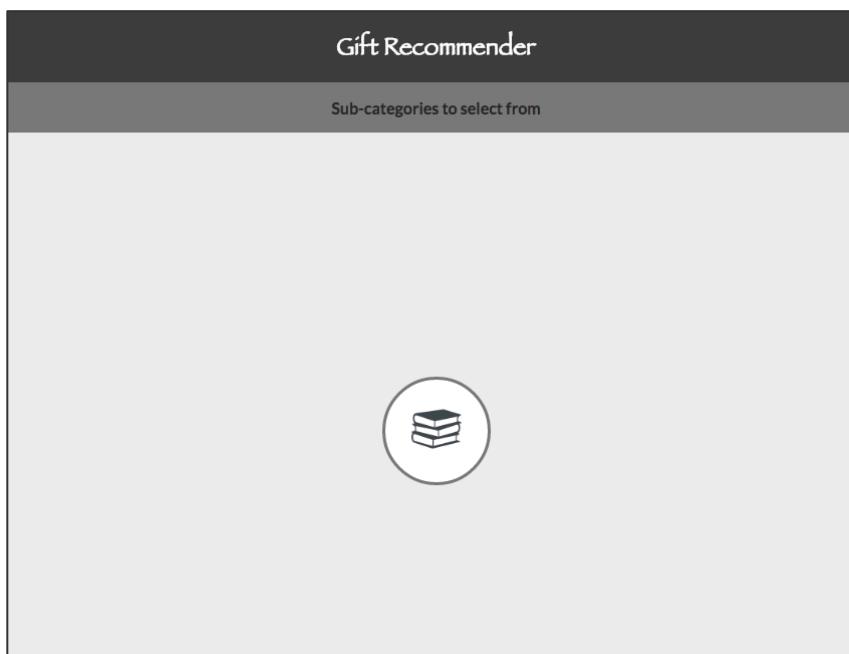
**Img:** Gift category options (showRecommendation.html)

This screen gives the users gift options (high level categories) to pick from. Once the user makes a selection, all other options disappear and selected category moves to the center and displays further sub-categories radiating from it.

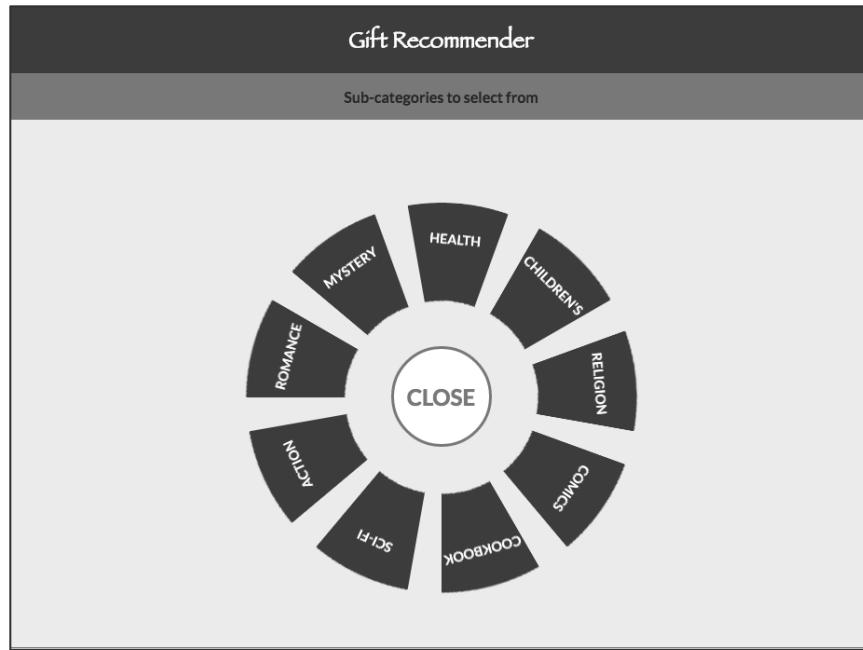
##### 5. Screen 5: Screen providing sub-categories for selected categories



Img: Storyboard image depicting gift sub-category options



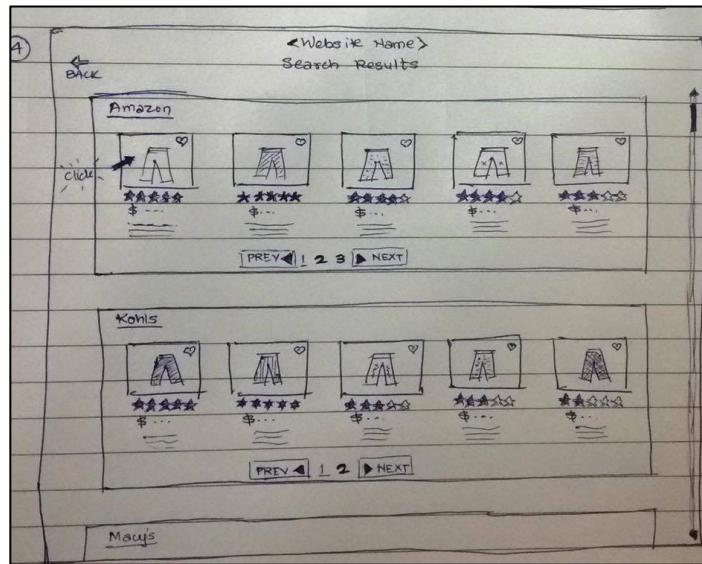
Img: Screen displaying selected gift category (index2.html)



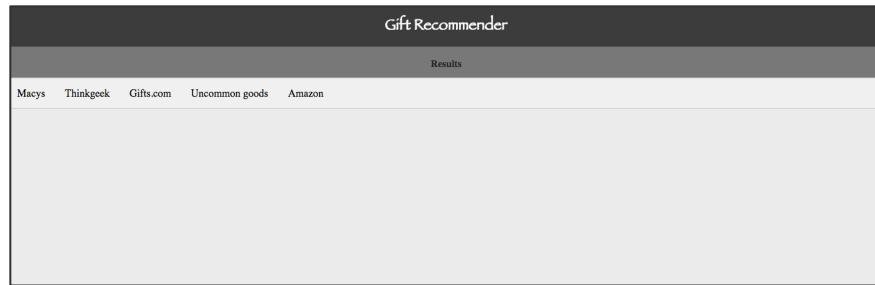
Img: Screen displaying gift sub-categories (index2.html)

When the user would click the sub-category, further categorization will be displayed, which the user can click on. Once clicked, our website would generate results from other popular e-commerce websites and display their search results.

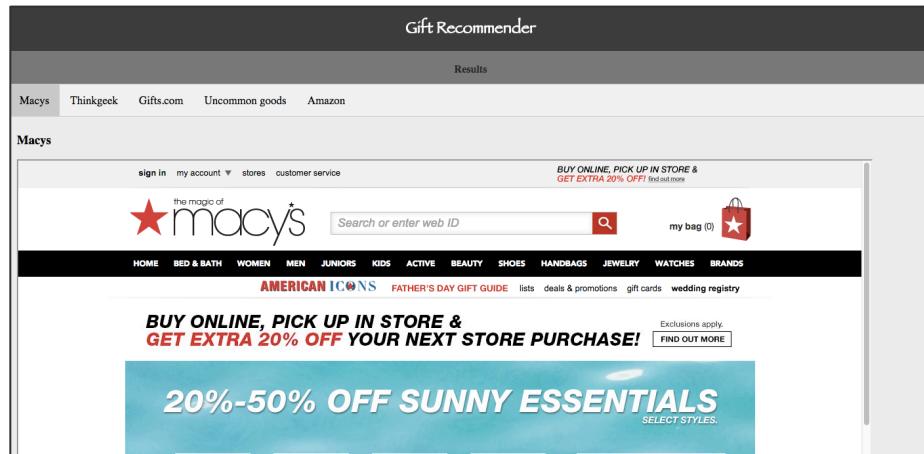
#### 6. Screen 6: Screen displaying gift recommendations



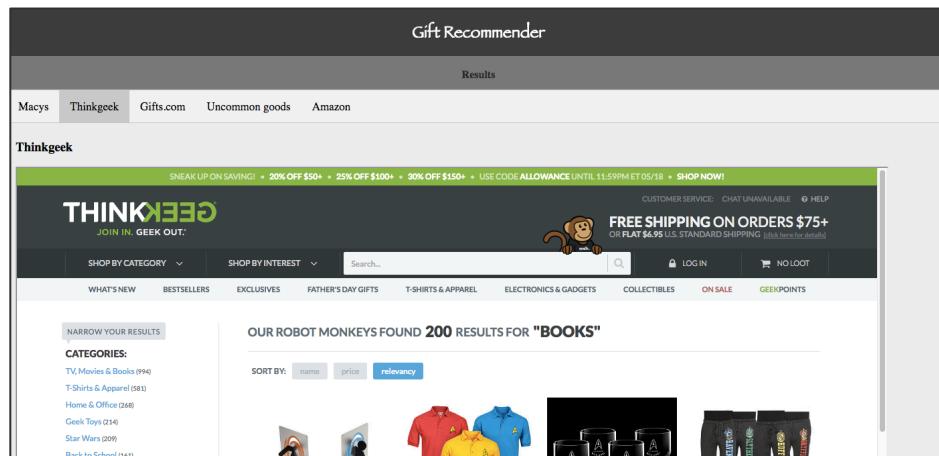
Img: Storyboard depicting the search results across various e-commerce websites



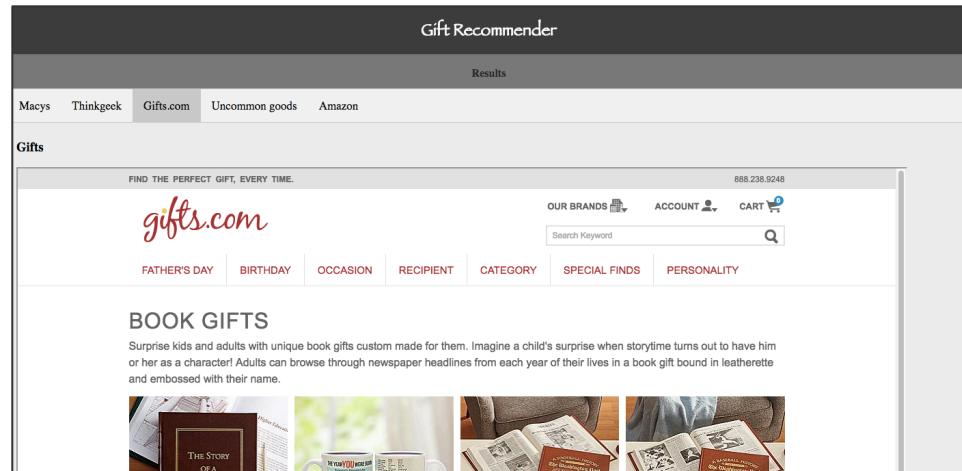
*Img:* Screen displaying the e-commerce website tabs (resultsTry.html)



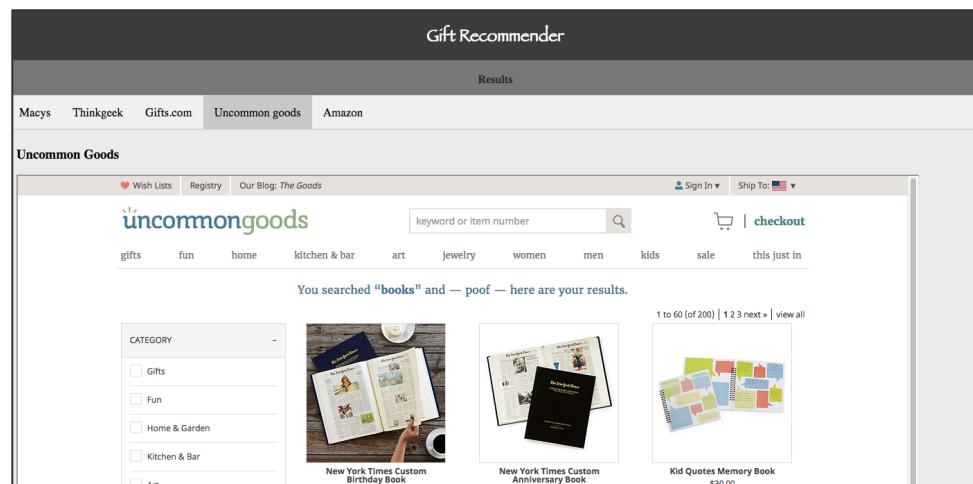
*Img:* Screen displaying the search results on Macy's (resultsTry.html)



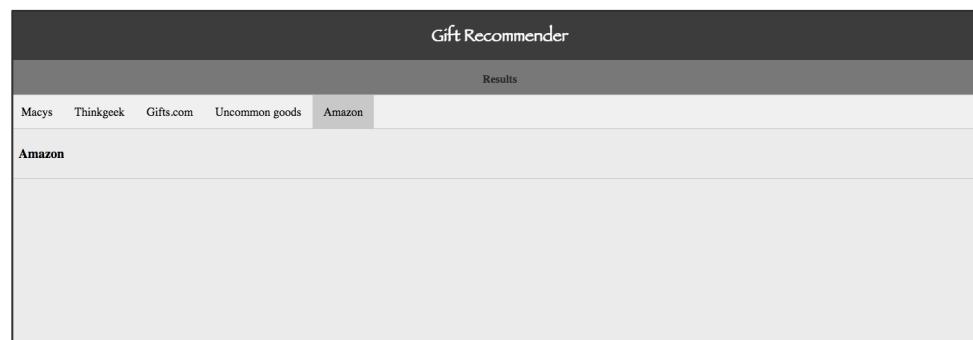
*Img:* Screen displaying the search results on ThinkGeek (resultsTry.html)



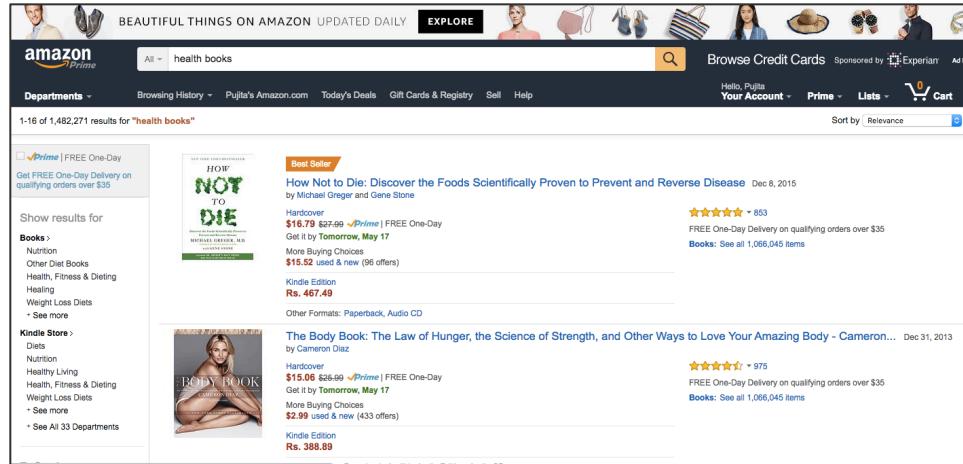
**Img:** Screen displaying the search results on Gifts.com (resultsTry.html)



**Img:** Screen displaying the search results on UncommonGoods (resultsTry.html)



**Img:** Screen displaying 'Amazon'. On clicking the Amazon tab, a window displays Amazon's search results (resultsTry.html)



**Img:** Screen displaying the search results on Amazon (new window)

Our search results show results from multiple e-commerce websites and their match to the user's query. The user can either purchase the product through our website (in the case of Macy's, ThinkGeek, Gifts.com and UncommonGoods) or purchase directly from the re-directed e-commerce website's page (in the case of Amazon).

### Additional capabilities

Once the user selects the product and clicks on it, our website directs the user to the respective e-commerce site to make a purchase. A feature which can be implemented in the future can include making the purchase from our website itself. Additionally, a wish list feature can be added using which the users can add items on their wish list to review later.

For this project, we have used the local storage of the browser as our database. For a fully functional system, a robust database like MySQL or Oracle can be used. Also after completing all functionalities, one can host the website onto the server.

### Needed materials and sources

Technical tools: We required JQuery library for JSON parsing and managing the DOM animation and events (when we pull up the search results from various websites onto our website). The Bootstrap library was used to customize the CSS. We anticipated using the item search or product advertising API's from popular e-commerce websites to integrate their search results into our website. But the requirement was satisfied by modifying the URLs of the popular e-commerce websites, according to the search criteria selected by the user.

Additionally, for this project, we have implemented a code that clears local storage through the index.html webpage. This is because the user can have data already present on their local storage, which could cause issues with smooth functioning of our project.

Reference material: We browsed through websites like w3schools, e-commerce websites and other similar platforms, that helped cover our knowledge gaps while working on this project.

## **Implementation summary**

### P2:

- First code prototype. Created initial draft of signup, login and filter pages to enable a user to sign up and login and enter filter criteria for gift recommendation.
- Created first prototype of the sub-category display feature.

### P3:

- Made necessary changes in sign/login page.
- Made changes in the filter page. Added a slider using JQuery, which lets you select a range of values. Added all functionalities necessary to proceed to next page.
- Added a new html page for gift recommendation based on selected values in previous page. Hence, depending on gender, age range, occasion etc., a recommendation of suitable products is made, for the user to select from.
- Added another page for once the user selects a recommended item, its different possible categories and sub-categories are generated for the user to choose from.

### P4:

- Modified the category display page to include one layer of sub-categories on clicking each category. Certain categories, like chocolates and gift cards, don't have sub-categories and directly display the recommendations when clicked.
- Final page to display recommendations (displayed in iFrames) in tabs – one for each vendor (if they sell that category of goods). List of vendors (e-commerce websites) that are currently integrated into the website include – Amazon, Macy's, ThingGeek, Gifts and UncommonGoods.

**Note:** Once the search results are displayed, to view the Amazon website result, the user would have to click on the image displayed on the 'Amazon' tab. This will redirect the user to a new window displaying the Amazon search results. We did this because Amazon has a secure interface, which does not open in an iFrame, unlike the other websites included in our search.

### P5:

- Slight modifications to fix the glitches on the results page.
- When the search criteria is such that no sub-categories satisfied them, the category display page showed a blank page. For such cases, we have displayed a message for the user so that s/he understands that it is because of improper combinations of the search criteria.

## **Lessons learned**

We learnt a lot more about dealing with CSS from this project. We understood the role of Bootstrap. Also we explored some more types of HTML elements that we had never used before. In JavaScript, we came across a few challenges while using various DOM element functions and event handlers, and it was fun solving them. We also learned to use the JQuery library for event handling and animation.

## **Suggestions for future projects**

If you ever want to integrate the search results from a variety of e-commerce websites just like ours, it is better to use iFrame and modify the URLs for search to each website and giving that URL path to these iFrame(s), instead of using APIs. You can also explore some more HTML elements like these or use your own creative ideas.