|  |  |
| --- | --- |
| COS10011 Assignment 1 Report | Version: 1.0 |
| Team Name: Radiance | Date: 2019-04-07 |
| Title: Snow Sports | |

# Introduction

## Scope of the Website

The website is called Glacier. This website is designed to rent out equipment for various snow sports, which are snowboarding, skiing and camping. The equipment for these sports are split into sub-categories, so that the user may have choices over what specific kind of product they would like.

One of the objectives of the website is to present opportunities for people to get engaged in snow sports. Due to the sheer number and price of the equipment required to get started, it can seem very daunting to some people, thus making them shun away from what otherwise would have been a wonderful experience for them. This is why the website rents out snow sports equipment, where it is rented out at much lower prices compared to the original. This would help encourage people to try out snow sports and see if they like it, and if they do, they may invest more, or if not, at least it did not cost them much.

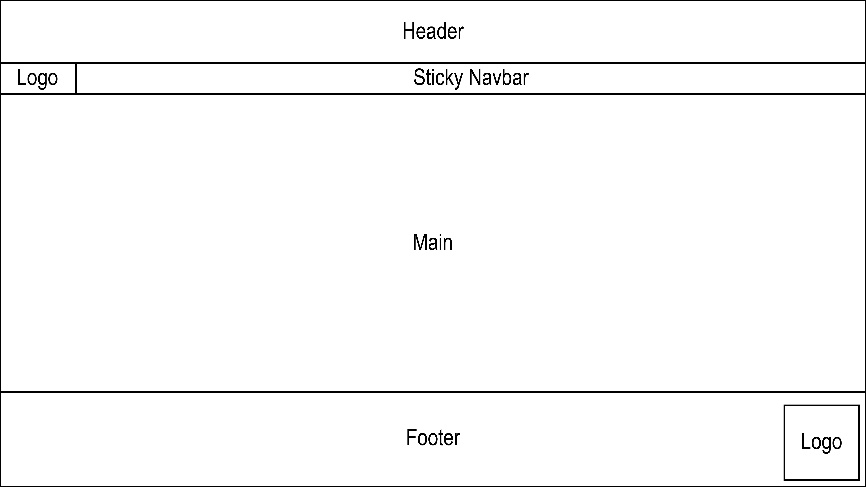
The website also aims to promote snow sports to the general public. Snow sports can be a wonderful experience, but many people do not want to get involved or possibly do not even know about snow sports. Thus, the website tries to provide information that is relevant and understandable to the general public, whom have little to no knowledge on snow sports.

## Intended Audience

This website is targeted at the general public within the age range of 21 – 40 years old, when they are still youthful and energetic, but old enough to support themselves financially. People that are looking to be get involved in snow sports, or sports in general are typically the “sporty” kind and like cool things. Thus, the webpage is designed to look modern and cool, so that it may appeal to them and interest them.

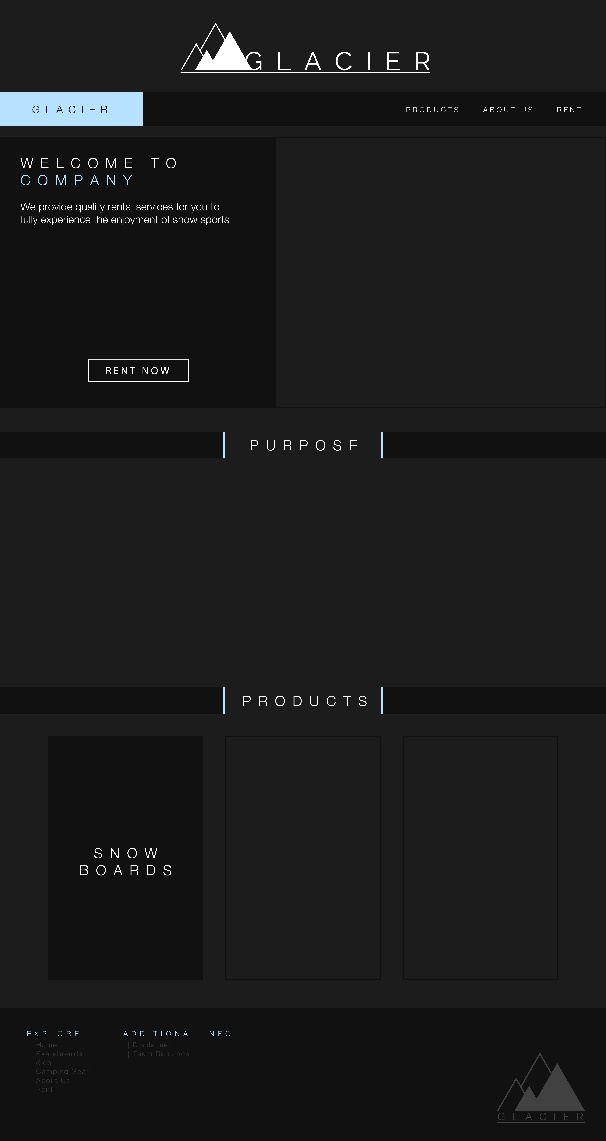
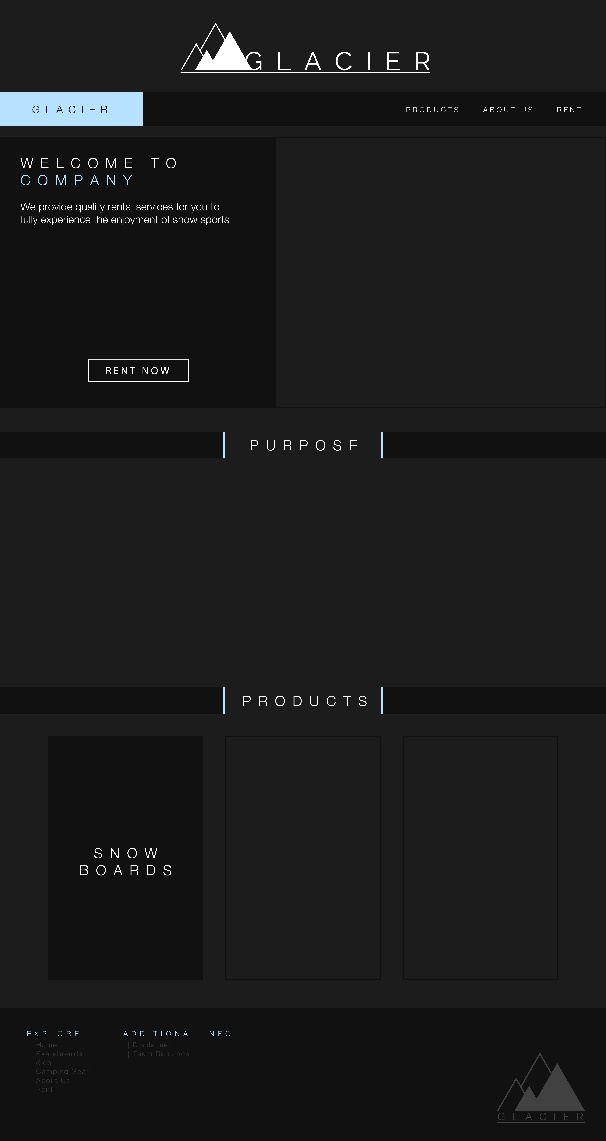
# Webpage Design and Structure

## Website Layout Mock Ups

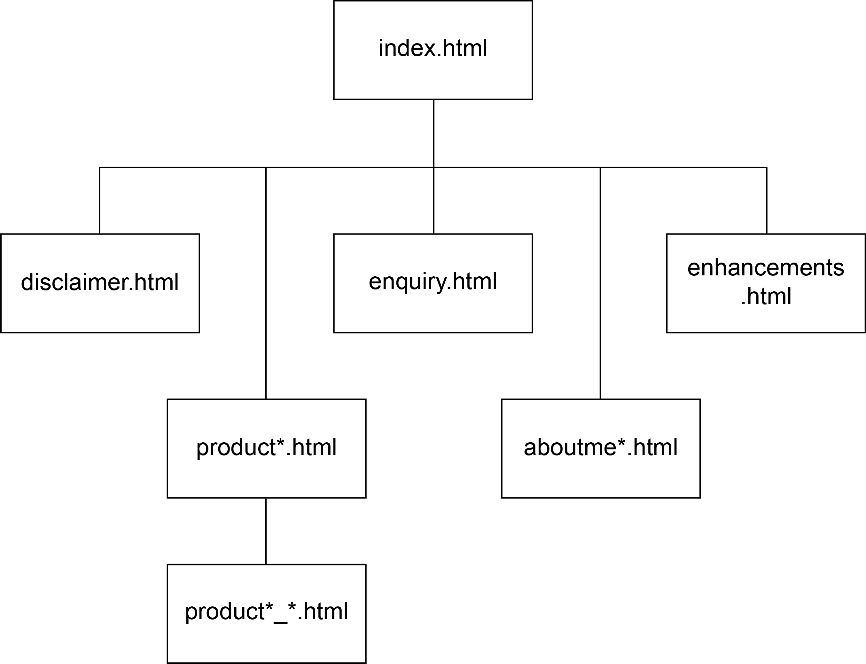


When designing the website, the team agreed upon a layout that will be used throughout all the pages of the website. It features a header, a navigation bar, the main content area, and the footer. The header contains the Glacier logo which can be used to link to the home page. The navigation bar contains another Glacier logo that can be used to link to the home page and is set to be sticky so that it shows up on the top of the webpage at all time. The footer will contain links that go to any page on the website and another Glacier logo on the bottom right corner.

Here are some images of the sample design that the team came up with prior to the development of the website. These designs were created using Adobe Illustrator.

## Website Structure



The website uses a hierarchical structure for the linking of the web pages. This is to categorize similar content under the same branch so that the user will not get too confused while viewing the website. Although the website is structured like this, the user may use the links in the footer to go to any page they want to, without having to first navigate to certain sites, with the exception of the product\*\_\*.html pages.

# End Product

## Actual User Interface

The website uses various web design principles in order to create an appealing and effective website.

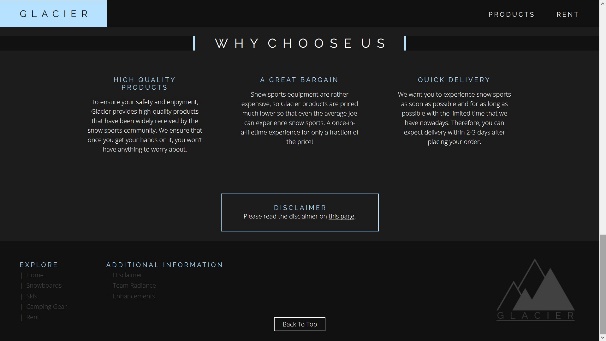
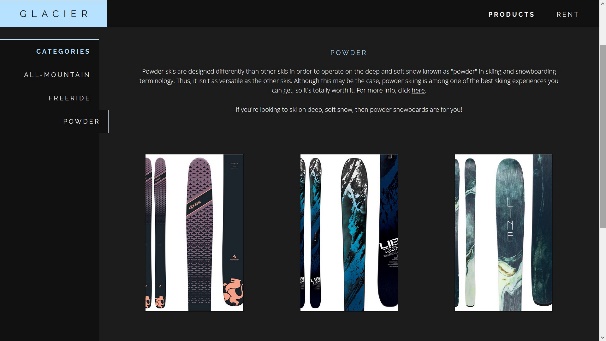
Firstly, the website uses a set color scheme, which consists of the darker and lighter shades of grey, white, and an icy blue color. These combination of colors help to provide contrast where needed, such as highlighting titles within the webpage to draw attention to itself. Since Glacier is a snow sports equipment rental service, the white and icy blue helps to reinforce the idea that the user is on a website that is related to snow sports.

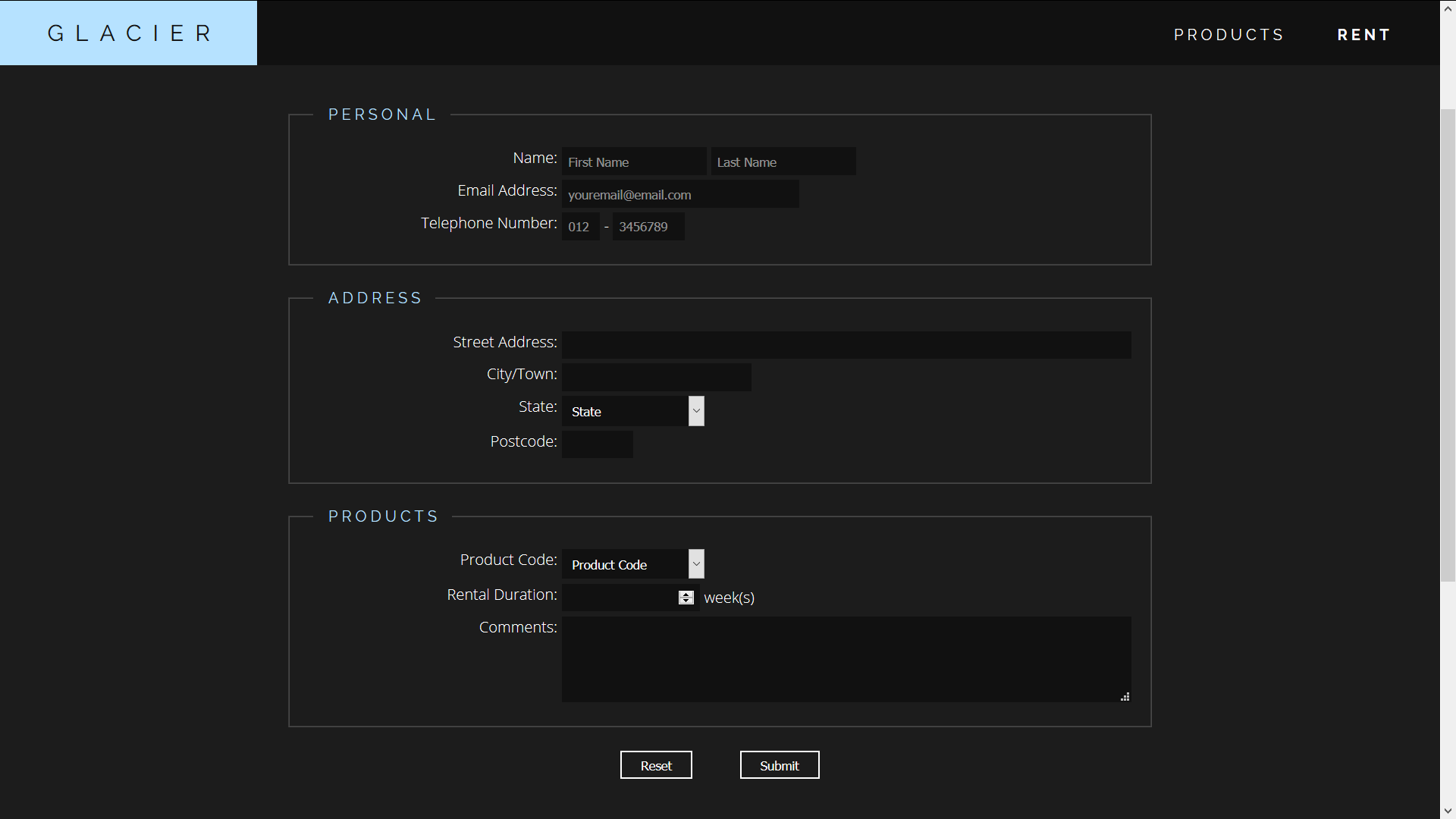
Secondly, the website uses a maximum of two fonts, which are “Raleway” and “Open Sans”. In this website, the “Raleway” font is used exclusively for titles, as the design of the font is rather stylish. All titles are also set to uppercase in order to draw more attention to the titles. “Open Sans” is used for body text as it is a simple and an easily readable font. In addition to this, different heading levels are used to indicate the hierarchy of the content, as large-sized titles typically refer to an entire category, while smaller ones refer to sub-categories.

Thirdly, there are repeated elements throughout all the pages of the website. This is to reinforce the identity of the website, and to also not confuse the user. The header, navigation bar, disclaimer box, and the footer are contained within all of the pages, thus the user can quickly learn the positioning of these elements, and be able to use the website more efficiently. The website also uses the same color scheme and fonts throughout the website, so as to prevent the user from getting shocked by any sudden changes.

Fourthly, the website has a flexible layout, which means that it scales with the width of the device. All users are likely to have different devices with different device widths, therefore it is important for the website to adapt and change accordingly. Using CSS tools like float, flexbox and grid, the team was able to create a website that automatically rearranges the content to fit the device-width. Under a certain width, the website will even change the layout to fit mobile devices, with the use of media queries. This allows the user to view the website, regardless of the device used.

Last but not least, the website uses a user-controlled navigation system. There is a navigation bar on the top which links to various parts of the website, and a footer which contains links to all parts of the website. This way, the user may choose how they would like to navigate the website. This assists in keeping the user interested in the content of the website, rather than begin frustrated over finding links to other parts of the website.



## Enhancements

### Mobile-view CSS

In addition to the "style.css" that was specified as a requirement for the assignment, another "style-mobile.css" was added to style the website for devices with smaller screen resolutions, especially mobile devices. This required the team to fully understand their own code and alter the CSS for mobile view while maintaining a similar layout and feel of the website. This enhancement helps improve usability of the website as the layout of the website changes according the width of the device, so that the user may view the website in the most comfortable way.

To achieve this, the team used "media queries" in order to apply styles after a certain "breakpoint", in this the case, the device width of 768px. Using the code "@media only screen and (max-width: 768px)" sets up the CSS file to only apply the styles when the device width is below 768px. The team then proceeded to reuse the selectors from the original "style.css" file and alter the CSS to conform to the device width.

### Grid & Flexbox Layout

Aside from using "float" to create a flexible layout, the team also utilized grid and flexbox layouts. These are powerful tools used by modern web designers to create a layout that they have more control over. This enhancement helps to create a "responsive web design", which means that it scales with the size of the device, so that the user may have utmost comfort when viewing the website, regardless of the device.

To achieve this, the team used "display: grid" or "display: flex" according to their needs. These codes come with extra properties that can be altered, such as "justify-content" or "flex-wrap" for flexbox or "grid-template-columns" for grid. By altering these properties, the team was able to achieve a flexible layout with less effort.

### CSS Selectors

Apart from the selectors that were specified to be used in the assignment, such as the contextual selector, grouping selector, etc., the team also used other CSS selectors to style some of the elements in the website.

Among the selectors used is the pseudo element selector "::before". The team used this selector to create a custom bullet type with the "|" symbol. The team also used the pseudo class selector ":focus" to style the form inputs when they have been clicked on and "focused". This gives feedback to the user which helps improve usability of the website. Lastly, the team used id selectors that target elements with ids that contains a given word. It is written like so, [id\*="word"]. Using this, the team was able to style elements that were similar but also slightly different without having to create multiple classes for them.

### CSS Animations

In order to improve usability of the website, the team also incorporated CSS animations into the website.

CSS animations can simply be done by using the pseudo class selector ":hover" to target the desired element, and then applying the desired styles on that element when hovered over. The team also used the "transition" property to "slow down" the animation, as it is more appealing than instantly popping out. The team also made use of this ":hover" selector to create a dropdown menu for the navigation bar.

### Preview Page

When completing a form, it is very likely for the user to enter values that are inaccurate but fit the criteria of the input, like misspelling a name or enter the wrong number. Thus, it is better to provide the user a way of checking and confirming their inputs before submitting the form.

To achieve this, values from the form are stored in the sessionStorage. Using the sessionStorage, values can be transferred to other webpages for use, thus it is useful for this context. The values are then obtained and displayed in another page for the user to confirm their inputs.

### Mobile Navigation Bar

In order to create fluid and responsive website, the layout of the website should change and rescale in order to conform to the device width and height. This is the same for the navigation bar. On a mobile device, the long navigation bar that can be seen on a desktop or tablet will not be as suitable, thus a different navigation bar should be used for mobile devices with small screens.

Thus, to achieve this, a navigation bar that contains a button to further open a list of links was used for this website. In order to create this "drawer" functionality, JS was used to toggle the displaying of the links. When the button is clicked, the event will change the display of the list based on its current value. If it is open, then it will close and vice versa.

### Display Change

Given a product, there may be multiple categories, and they may even have sub-categories in each category. This was the case for this website, and since the number of products in each sub-category was not enough to warrant the creation of another page, JS was used to display different sub-categories at a given moment.

To achieve this, the content of each sub-category were placed in seperate divs. These divs were then stored in a list in JS, along with buttons that correspond to these divs. When a certain button is clicked, the corresponding sub-category will be displayed while hiding the others.

### Regular Expressions

Regular expressions, or RegEx for short, are used to check strings for a match with a given pattern. These can be used in many different ways, such as to check input to see if they fit a certain criteria, or to find specific words in a long paragraph. RegEx is used in this website to validate form input based on certain criteria.

To do this, a simple ".match" method can be used to check a string according to a given pattern. The complex part is the RegEx, as it has its own syntax to form patterns that match different criteria.

### HTML Form Element

To access inputs in a form, the HTML Form Element was used. This is done by just indexing the document.forms attribute, like "document.forms["formname"]". This will return the form element which can then be used to access input values or other attributes. In this assignment, the form element was used to obtain values from the inputs, to manipulate the "onsubmit" event of the form and to also set values into the inputs.

# Contributions

|  |  |
| --- | --- |
| **Team Member 1** | Ian Ong |
| **Contribution (%)** | 30% |
| **List of Contributions** | HTML structure for enquiry.html, aboutme\*.html, disclaimer.html, and enhancements.html |
| CSS styles for enquiry.html, aboutme\*.html, disclaimer.html, and enhancements.html |
| Content (images and info) for product3.html and product3\_\*.html |
| JS for form validation in enquiry.html |

|  |  |
| --- | --- |
| **Team Member 2** | Jonathan Seng |
| **Contribution (%)** | 35% |
| **List of Contributions** | HTML structure for the header, navbar, footer and also index.html |
| CSS styles for the header, navbar, footer and also index.html |
| Mobile-view CSS |
| Content (images and info) for index.html, product1.html, product1\_\*.html, and enhancements.html |
| HTML and CSS bug-fixing |
| JS for mobile navigation bar and display change in product\*.html |

|  |  |
| --- | --- |
| **Team Member 3** | Karyn Chong |
| **Contribution (%)** | 35% |
| **List of Contributions** | Main web page designer, create sample designs with Adobe Illustrator |
| Design for Glacier logo |
| Structure for product\*.html and product\*\_\*.html |
| Styles for product\*.html and product\*\_\*.html |
| Content (images and info) for product2.html, product2\_\*.html |
| Photos for aboutme\*.html |
| JS for data transfer between pages (product\*\_\*.html, confirmation.html) |

# Group Reflection

During the development of the website, the team faced various challenges and also learned various valuable lessons.

One of the challenges were communication. In the beginning, it was difficult for the team to communicate ideas effectively and compile the work done by each individual. Every member had their own vision and interesting ideas for the website, but it is impossible to implement all ideas, therefore compromises have to be made. After much discussion and re-designing, the team came up with a design that everyone was content with. When compiling the work, at various times, there would be a mistake and little bits of code may be deleted which ends up breaking the website.

Another challenge faced during the development of the website were bugs within the HTML and CSS code. At various times, some of the members were stuck and not making any progress due to the lack of knowledge on HTML and CSS code. This was not too big of a problem as the members consulted and discussed with each other for a solution. If a solution was not found, the team consulted the Internet for information on websites such as StackOverflow and W3Schools.

This assignment taught the team various things about working in a team. Communication is very important, especially when someone has an idea or a problem that needs to be fixed. If a member makes any sudden changes to the code without the knowledge of other members, it could be very detrimental to the workflow, as bugs would suddenly appear and the team would have to search for the source of the problem within a rather large file.

Another thing the team learned is that the code should be kept as clean and neat as possible. This is so that bug-fixing and future changes can be made with as little time spent as possible. Comments are a great way to explain what a specific piece of code does, especially when there are multiple members working on the same thing.

There are definitely areas of improvement to increase the work efficiency of the team. The team could try to make use of apps or programs online that can help the team to collaborate and communicate effectively. The team should also communicate with each other more often so that the team can keep up with each other’s progress. The team could possibly even set up a standard for communication, such as “What updates were made, which lines, etc…” so that the team know exactly what changes were made, how and where.