



Part C: Final Project

Group: Team 7

Jiaqi Zhou 46299660

Yeonjin Kim 46226509

Mengxuan Zhang 45603040

Yawen Zhang 45908983

Johnny Chi 33328690

Table of Contents

Describes the final product.....	3
Detailed description of the final project.....	3
Description of implementation.....	11
Final product link	18
Discussion of issues and major challenges:	19
Challenge 1: How to divide up the code tasks and make the teamwork efficient?.....	19
Challenge 2: The API for the main function is different from what we learned in class.....	20
Challenge 3: We think only showing the location is not good enough.....	23
Challenge 4: The default colour of leaflet map limits the colour theme of our website	24
Challenge 5: Light themes and low-contrast colours that are unfriendly to people with vision impairment or colour blindness.....	24
Challenge 6: The algorithm of routing function	28
A response to the feedback	31
Reflection	37
The Likes and Dislikes.....	37
Successful and Unsuccessful Aspects.	37
Improvements.....	38
Chose not to Implement.	38
Reference	40

Describes the final product

Detailed description of the final project

The final product of our website includes various types coral reefs that are listed on our map, the data is retrieved Queensland Government Data Portal. Our team's goal is to build a user-friendly website for all ages. Our website has been very successful but challenging due to time constraints and obstacles that presented itself during the project.

The purpose of the website is to make user-friendly and easy to navigate. Main objective is to let the users find all information on one page about coral reefs and locations. The website is designed for users ranging from average day user to professionals who specialise in coral reefs, the API dataset retrieved from the Queensland Government Data Portal that displays different types coral reef in Queensland. At the present time, there are a lot of websites that provide various types coral reefs with beautiful photos and eye-catching design. The problem with these websites they have so much undesirable information and buttons and it makes it harder for the user to navigate properly. Also, most of the website have too many advertisements on each page than the required information about the coral reefs. Therefore, we created our website as simple as one button click, this allows users to find all the information by clicking on the map.

The interactive map provides all the detailed information about the coral reef and how to travel there. As a result, our website only takes a few steps to get to their required information on one page. By designing like this, users will not waste their time to explore other pages or endless searches to get to their required information. Hence, our website purpose is to provide users to find a particular coral reef or reefs in Queensland. Also, we have imbedded routing which allows users to users to plan their trips to different coral reefs by themselves.

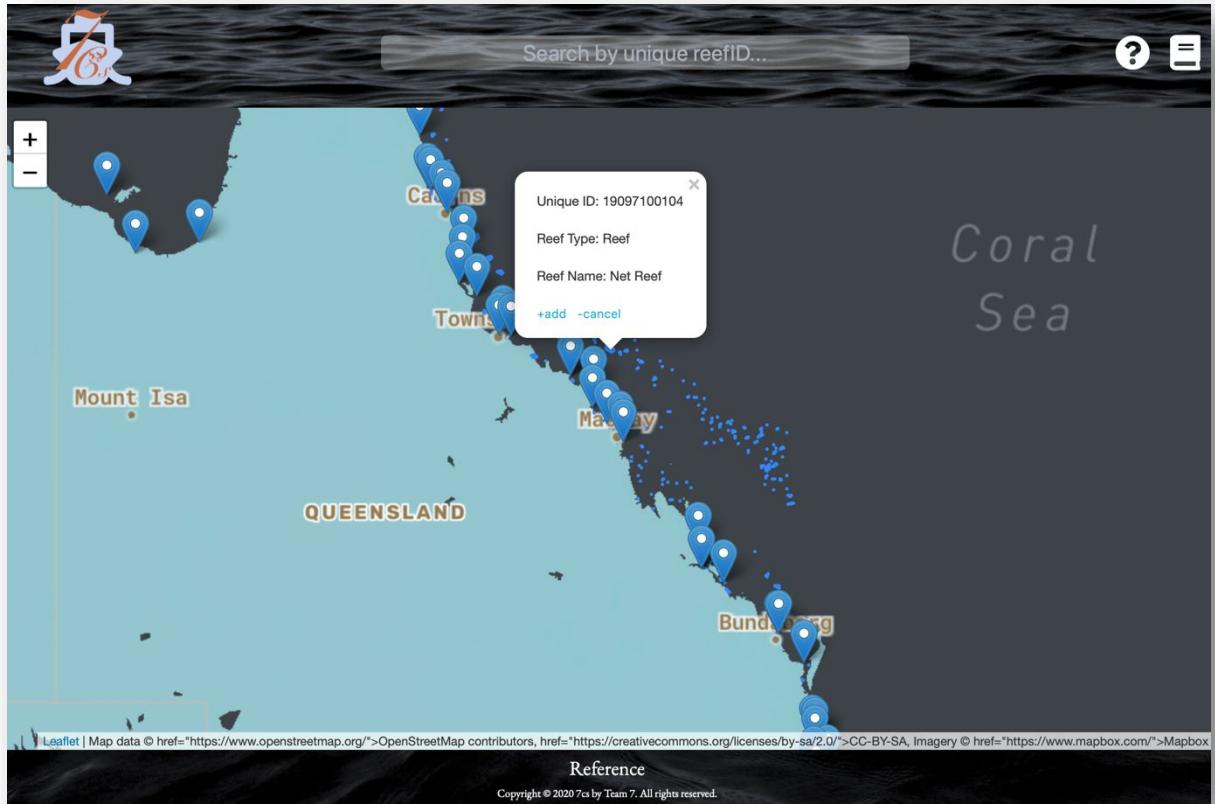


Figure 1: API dataset for Coral Reef on the map on “Home Page”

Our main target audiences are professional tourists and academics who have a lot of experience with the coral reefs. Therefore, they will want to choose where they want to go and will create their own routes by adding the destinations on the website. Our website provides users with more flexible routes and more detailed information on the map compare to other websites. We are also targeting people who love scuba diving because they want to explore different types of reefs.

Chai Choi-sum



"I love water and fish. I have big aspirations and I want to go snorkelling and scuba diving in Australian waters to see reef and then I'll meet my wife."

Age: 67
Occupation: Retired
Family: Married
Location: Australia, Perth
Gender: Male
Character: Carefree

Personality



Loving Stubborn Fighter
Conscientiousness Courageous

Bio

He comes from poor background but with happy life. He immigrated from Asia to Australia 30 years ago as refugees. He has 3 sisters and 1 brother; he worked very hard everyday just to put food on the table. He worked as a labourer, a day labourer and had his own business. His dad passed away when he was 12 years old. He is the eldest child in the family and he has to be strong for his mother and his siblings. He studied hard each day so one day he can provide his mum and siblings a better life. He has a brother who wants to prove these guys with his skills alive. He has a wife most of time. His last life or his previous from Australia and became Engineer for CPI Contractors. Once he got into firm, he worked hard to become lead engineer after 3 years. His 2nd life goal was to marry my sweetheart whom he met at university, she was his Princess who would come to him every day. They got married and they have a girl and 1 boy. They all become doctors, lawyers and environmentalist. He is the happiest dad in the whole world with a beautiful wife, loving and caring children. And finally, He was able to go scuba diving all around the world at the age of 60.

His last life is now to live happy and quiet life with my wife and travel around the world with her. He wants the memories to last eternity until they meet in the after-life. Also, his last goal is to catch the biggest tuna in the world and then taste the fresh and delicious meat. That would make my life complete.

Goals

1. He has reached his goal to retirement. He hopes he can live another 33 years to see his 100 milestones.
2. Love his wife until death do them apart.
3. To see the beautiful underwater world!
4. To be role model for his grand-children.

Frustrations

1. Old age, wish he could live up to 200 years old.
2. Sharks, he hates sharks because when he goes swimming in the ocean. He always worried that he will get killed by a shark.
3. Wars and terrorism in the world.

A day in the life

His day starts with me waking up at 4:30 am. Kisses his beloved wife. Quick pray to his God to watch over his wife and children and one day there will be World Peace. Once his prayer is done, he will go for 1 hour walk at his local park and meet all his local neighbourhood friends and greet them with love and happiness. After his wonderful walk, he would take nice dip in his pool and have a swim for 1 hour.

Once he is done, his lovely wife always makes him salmon, poached eggs on toast. Around 10-11 am he would go searching internet for different coral reefs and fishes. Then if I really like it, I will make booking for wife and myself to go on weekend trip scuba diving and snorkelling.

After 3-4 hours of scuba diving and snorkelling, he will go home and will make wonderful sushi or beer battered fish and chips for my family. After dinner, his wife and him will sit at the lounge room and talk about their wonderful life and memories. These are the memories they want to carry with them each night before they go to bed at 9 pm. This is their life each day and they are thankful everyday that they are alive and living well.

Figure 2: Non-professional target audiences.

Our second target audience are people of all ages who have no experience (tourist). This group of audience are interested in visiting coral reefs for short term vacation. So, our website is built very simple and easy to access all the information on one page. Also, the website shows a guideline for the users on how to use the website such as how to find coral reefs and how to add destination points where they want to visit. Therefore, the users can view various coral reefs on the map to get detailed information, if the users decide if they want to explore the coral reef they will click on "Add" hyperlink to add to their route.

Jason Smith



"I love coral reef; I want to see all kinds of coral reef in the world."

Age: 24
Occupation: Student
Location: QLD, brisbane
Gender: Male

Personality



Gentle Brave Helpful
Adventurous Independent

Bio

Jason is a student in UQ. His major in marine biology. When he was young, he lived nearby the sea. Every morning he used to go to the sea with his father. Sometimes they could see pieces of coral reef and collect them at home. He wants to see all kinds of coral reefs in his life; this is the reason why he chooses to study marine biology. Now, he works in the café shop every day in order to earn enough money to buy diving equipment. He plans to travel to the Great Barrier Reef during the summer or winter holidays, where the coral reef is very famous. He wants to take many pictures of the coral reef to tell people to protect the marine environment.

Goals

- Get a driving license
- Master underground photography skills
- Photograph different types of the coral reef

Frustrations

- He was worry about the coral reef right now. They are very fragile. They are very vulnerable to damage and influence from humans and the Marin environment.

Preferred Applications

National Geographic
WWF Instagram

A day in the life

He goes to bed early and wakes up early, around 6 am. After having breakfast, he tries to keep fit by jogging nearby the river or swimming.

He makes a lunch box before 8.30 and then takes the bus to school. He usually gets to school at nine o'clock.

Every time after class, he goes to the library to read a book related to the Marine environment or Marine animals.

Every Thursday and Friday, he needs to go to work in the café shop. This is his weekday's life. At the weekends, he always goes surfing on the Golden Coast, which can help him to familiar sea.

Figure 3: Professional target audiences.

To understand the dataset of the website, first we need to analysis the data. The data in this resource is too in-depth which is complex for most users. Therefore, we decided only to use three fields for information popup such as “Unique ID”, “Reef Type” and “Reef Name”. These three fields are displayed on the map so users can choose their preferred spots based on these fields of information. The data are sorted by JavaScript and API to provide the information more accurately for the users.

FID	SORT_GBF_LABEL_ID	SUB_NO	CODE	UNIQUE_I	FEATURE_GBR_NAM_FEAT_NAM	QLD_NAM_X_LABEL	GBR_ID	LOC_NAM	LOC_NAM_X_COORD	Y_COORD	Area_HA	GlobalID	SHAPE_Lei	SHAPE_Area
1	21752 21-752	100	21-752-10	2.18E+10	104 U/N Reef Reef	21-12215	21752	U/N Reef (U/N Reef f	152.3432	-21.9199	20.02624	{82192105	0.016461	1.74E-05
2	99416 99-416	100	99-416-10	9.94E+10	104 U/N Reef Reef	10-10535	99416	U/N Reef (U/N Reef f	142.1942	-10.4215	3554.084	{7019D15	0.33844	0.002917
3	21118 21-118	100	21-118-10	2.11E+10	104 U/N Reef Reef	U/N Reef 21-1185	21118	U/N Reef (U/N Reef f	151.2748	-21.2016	458.8209	{53938111	0.078906	0.000397
4	99497 99-497	100	99-497-10	9.95E+10	104 U/N Reef Reef	U/N Reef 10-11345	99497	U/N Reef (U/N Reef f	142.6127	-10.6097	101.3304	{513D03E	0.074427	8.33E-05
5	25004 25-004	100	25-004-10	2.5E+10	102 Little Woo Island	Little Woo 25-10145	25004	Little Woo Little Woo	153.02	-25.3188	{F9CDC110	0.024993	1.22E-05	
6	12100 12-100	100	12-100-10	1.21E+10	103 U/N Cay Cay	U/N Cay 12-1005	12100	U/N Cay (U/N Cay (I	143.6962	-12.7812	{F65D821	0.009297	5.58E-06	
7	99509 99-509	100	99-509-10	9.95E+10	104 U/N Reef Reef	10-11245	99509	U/N Reef (U/N Reef f	143.4717	-10.6241	36.37208	{F6E937D	0.020788	3.00E-05
8	20284 20-284	100	20-284-10	2.03E+10	104 Hesket Re却 Reef	U/N Reef 20-2845	20284	Hesket Re却 Hesket Re却	149.4813	-20.9356	8.013739	{7BBD3C	0.01559	6.95E-06
9	99439 99-439	100	99-439-10	9.94E+10	104 Alert Patch Reef	Alert Patch 10-10465	99439	Alert Patch Alert Patch	142.3536	-10.485	415.1164	{80E6161	0.098751	0.000341
10	20153 20-153	100	20-153-10	2.02E+10	104 U/N Reef Reef	U/N Reef 20-1535	20153	U/N Reef (U/N Reef f	150.5124	-20.095	36.73548	{A6800DA	0.025789	3.16E-05
11	23033 23-033c	103	23-033-10	2.3E+10	108 Egg Rock (Rock	Egg Rock 23-0338	23033C	Egg Rock (Egg Rock (I	151.1017	-23.198	{061F58D	0.002557	4.56E-07	
12	12029 12-029	100	12-029-10	1.2E+10	104 U/N Reef Reef	U/N Reef 12-0295	12029	U/N Reef (U/N Reef f	143.9106	-11.9964	12.56314	{E6BC056	0.012683	1.04E-05
13	22846 22-846	100	22-846-10	2.28E+10	104 U/N Reef Reef	U/N Reef 22-10205	22846	U/N Reef (U/N Reef f	152.5716	-22.1134	7.328857	{0A1CF34	0.010569	6.37E-06
14	20060 20-060h	108	20-060-10	2.01E+10	104 Long Islani Reef	U/N Reef 20-060H	20060H	Long Islani Long Islani	148.8557	-20.325	2.073936	{CEA346E	0.009253	1.79E-06
15	21124 21-124	100	21-124-10	2.11E+10	104 U/N Reef Reef	U/N Reef 21-1245	21124	U/N Reef (U/N Reef f	151.128	-21.2571	532.7763	{825838D	0.08860	0.000462
16	11134 11-134	100	11-134-10	1.11E+10	102 Pirie Islet Island	Pirie Islet 11-1345	11134	Pirie Islet (Pirie Islet (142.8988	-11.5884	{888C001E	0.015407	4.46E-06	
17	19061 19-061	100	19-061-10	1.91E+10	104 Jacqueline Reef	Jacqueline 19-0615	19061	Jacqueline Jacqueline	148.9038	-19.2139	579.7361	{F6A9DE1	0.101113	0.000498
18	21254 21-254	100	21-254-10	2.13E+10	104 U/N Reef Reef	U/N Reef 21-2545	21254	U/N Reef (U/N Reef f	152.3474	-21.3508	186.7297	{2CF6346E	0.077335	0.000161
19	11133 11-133	100	11-133-10	1.11E+10	104 U/N Reef Reef	U/N Reef 11-1335	11133	U/N Reef (U/N Reef f	142.8438	-11.5637	508.955	{B9524B0	0.11618	0.00042
20	22136 22-136	100	22-136-10	2.21E+10	103 U/N Cay Cay	U/N Cay 22-1365	22136	U/N Cay (U/N Cay (I	152.6489	-22.2488	{006500F4	0.007292	3.80E-06	
21	21529 21-529	100	21-529-10	2.15E+10	104 U/N Reef Reef	U/N Reef 21-5295	21529	U/N Reef (U/N Reef f	152.173	-21.8669	672.1661	{9AC9A94	0.101136	0.000584
22	23810 23-810	100	23-810-10	2.38E+10	102 Compigne Island	Compigne 23-8105	23810	Compigne Compigne	151.2585	-23.7858	{CE35DBA	0.033676	4.82E-05	
23	19180 19-180	100	19-180-10	1.92E+10	104 Gargoyle F Reef	Gargoyle F 19-1805	19180	Gargoyle F Gargoyle F	149.6834	-19.7409	541.1358	{887900A4	0.092014	0.000466
24	99402 99-402	100	99-402-10	9.94E+10	104 U/N Reef Reef	U/N Reef 10-10015	99402	U/N Reef (U/N Reef f	143.9322	-10.374	26.88274	{F42BFBDF	0.020015	2.21E-05
25	21423 21-423c	103	21-423-10	2.14E+10	102 Eagle Islet Island	Eagle Islet 21-423C	21423C	Eagle Islet Eagle Islet	150.1948	-21.978	{ABA0082I	0.004283	1.10E-06	
26	21603 21-603	100	21-603-10	2.16E+10	104 U/N Reef Reef	U/N Reef 21-10575	21603	U/N Reef (U/N Reef f	151.6062	-21.2605	26.30616	{72BD17E	0.174111	2.28E-05
27	11051 11-051	100	11-051-10	1.11E+10	104 U/N Reef Reef	U/N Reef 11-0515	11051	U/N Reef (U/N Reef f	143.0732	-11.3705	16.22602	{29540923	0.01417	1.34E-05
28	20814 20-814	100	20-814-10	2.08E+10	106 U/N Rock Rock	U/N Rock 20-8145	20814	U/N Rock U/N Rock	148.5057	-20.0781	{AE960C2I	0.002879	3.09E-07	
29	20069 20-069	100	20-069-10	2.01E+10	106 U/N Rock Rock	U/N Rock 20-0695	20069	U/N Rock U/N Rock	149.114	-20.1941	{744430E	0.001395	9.85E-08	
30	18101 18-101	100	18-101-10	1.81E+10	104 Judith Wrij Reef	Judith Wrij 18-1015	18101	Judith Wrij Judith Wrij	147.9513	-18.8997	3964.614	{1737C4B1	0.234648	0.003402
31	21345 21-345	100	21-345-10	2.13E+10	104 Ridge Reef Reef	U/N Reef 21-3455	21345	Ridge Reef Ridge Reef	149.6382	-21.6748	15.46106	{606F992A	0.028785	1.35E-05
32	16026 16-026	100	16-026-10	1.6E+10	104 Tongue Re Reef	Tongue Re 16-0265	16026	Tongue Re Tongue Re	145.7926	-16.3209	26007.88	{82A2474	0.075126	0.022001
33	99516 99-516	100	99-516-10	9.95E+10	102 Murangi Is Island	Murangi Is 10-10285	99516	Murangi Is Murangi Is	142.4927	-10.6954	{21EFB2CC	0.006018	2.13E-06	

Figure 4: Data source form data.qld.gov.au

The intended function of the website includes the map on “Home Page” with real time interaction. This allows the users to explore on the different types coral reef and received useful information about that reef. Also, if the users know the reef “Unique ID” they can type in “Unique ID” in the search bar which will bring up the location of the reef.

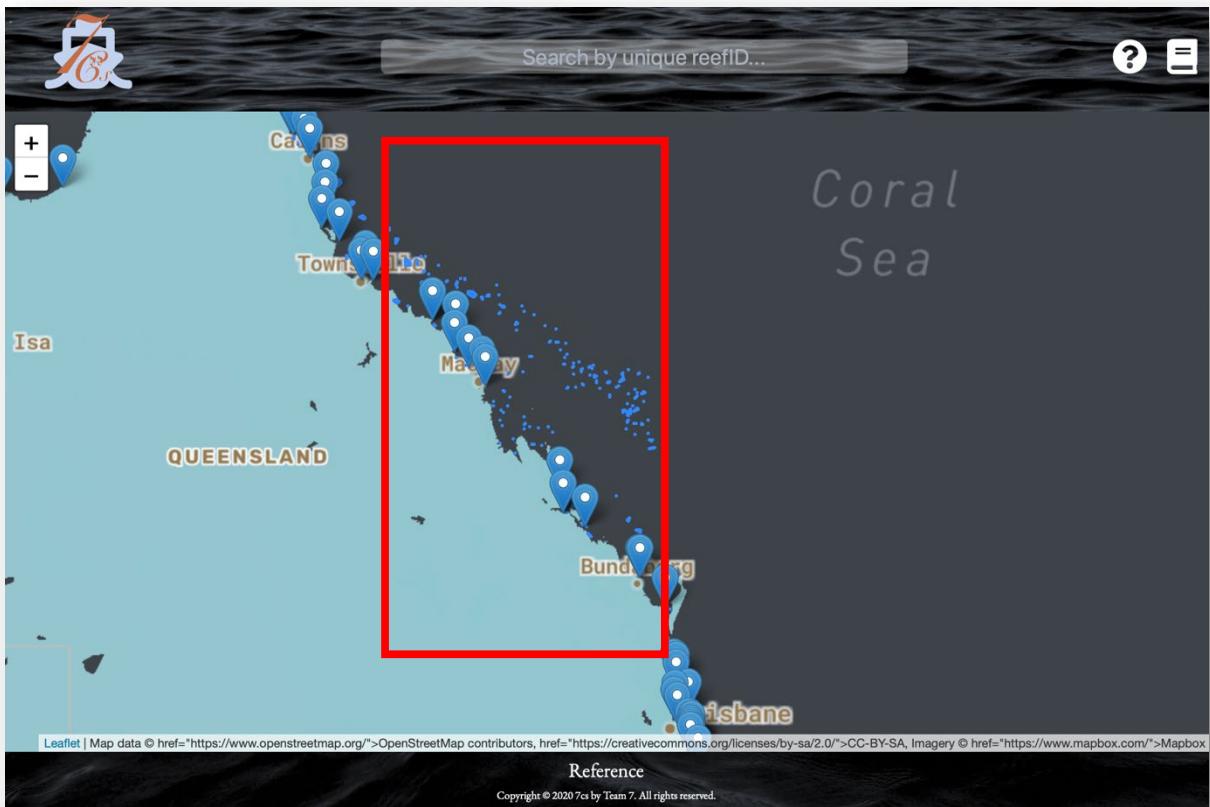


Figure 5: Home Page.

Users also can add their own routes where they want to travel. User's click on particular reef on the map, it will then show detailed information about the coral reef and then give the user an option to add or cancel the destination route. Clicking “+add” button will allow the users to add the place as their own route, it shows the alert which reminds the user wants to add or not. after that, the added places will be connected as a line on the map. If users want to make amendments to their routes they can also cancel a particular route by pressing “– cancel” hyperlink, a prompt will pop up to ask the user if they want to cancel their route.

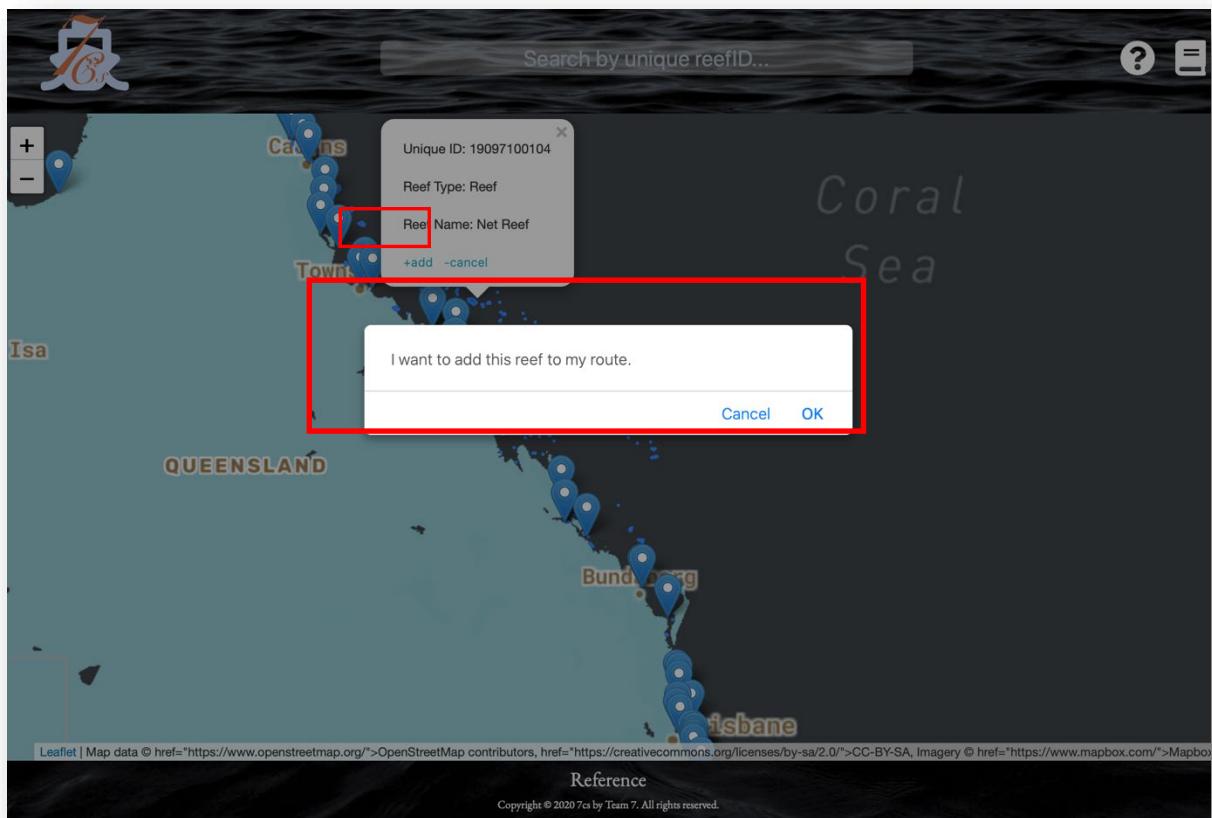


Figure 6: Add own route.

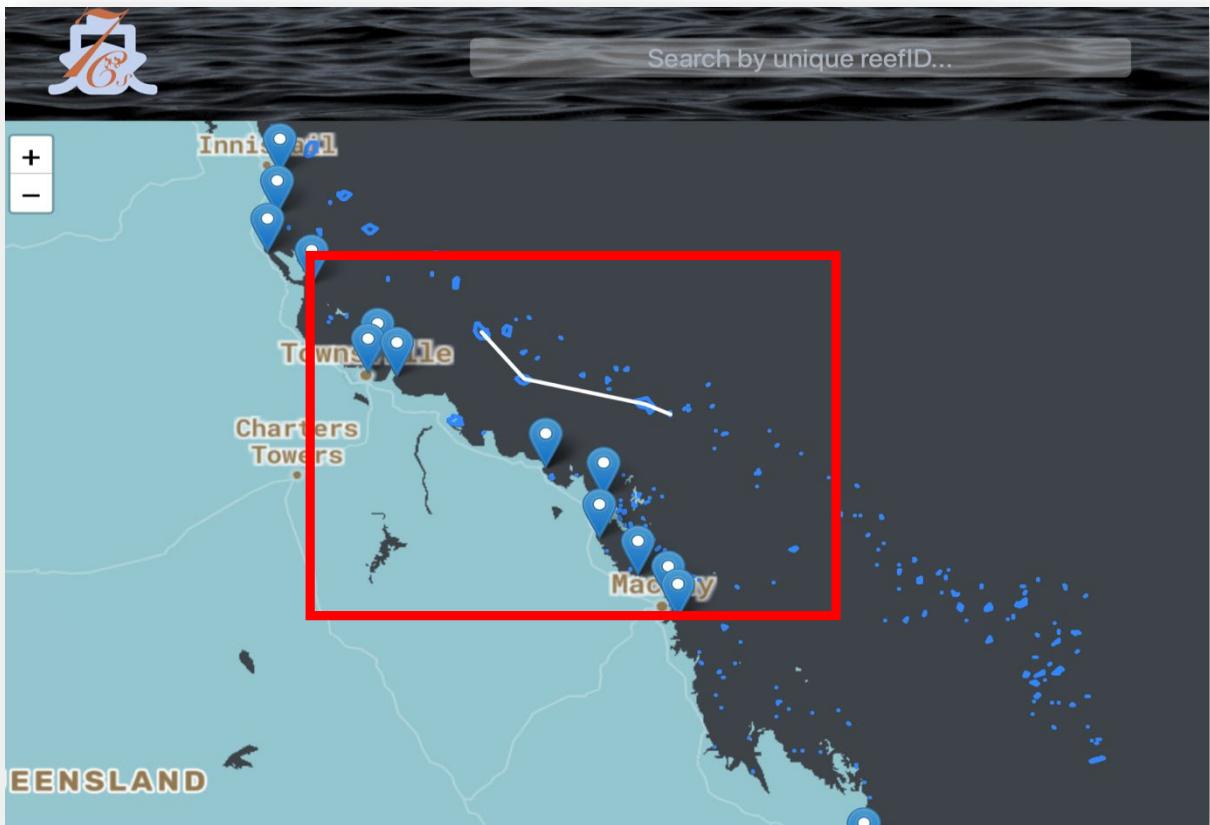


Figure 7: Add place connected as a line on the map.

Furthermore, we added “Question Mark” icon shown in figure below. The function of the icon is to explain to users on how to interact with the website. A lot of people who are not computer illiterate may not be to understand how the website works, thus we added this function for users who need guidance or help.

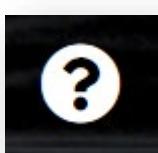


Figure 8: Instructions and help function.

The disk icon shown below, takes the user to another page giving brief explanation of types coral reefs and how our coral reef is been threatened and destroyed. We want our users to understand how our coral reef maybe gone in decade if we do change our habits and ways in society.



About Coral Reef

[What is a coral reef?](#)
[Types of Coral Reef Formations](#)
[What are the main threats to coral reefs?](#)

What is a coral reef?

Coral reefs are made up of colonies of hundreds to thousands of tiny individual corals, called polyps. These marine invertebrate animals have hard exoskeletons made of calcium carbonate, and are sessile, meaning permanently fixed in one place. Polyps grow slowly, forming different shapes and sizes depending on their species. Coral reefs provide an important ecosystem for marine life, offering food and shelter among their crevices and branches for animals including fishes, mollusks, sea urchins and sponges.

[Back to top](#)

Types of Coral Reef Formations

There are three main types of coral reefs – fringing reefs, barrier reefs, and atolls. Patch reefs and bank reefs are also important natural systems

[Back to top](#)

What are the main threats to coral reefs?

1. Climate change: Corals cannot survive if the water temperature is too high.
2. Overfishing: This affects the ecological balance of coral reef communities, warping the food chain and causing effects far beyond the directly overfished population.
3. Unsustainable coastal development: Tourist resorts and other coastal infrastructure have been built directly on top of reefs or close enough to them to cause significant damage.
4. Pollution: Urban and industrial waste, plastics, sewage, agrochemicals, and oil pollution are poisoning reefs.
5. Destructive fishing practices: These include cyanide fishing, blast or dynamite fishing, bottom trawling, and muro-ami (banging on the reef with sticks). Bottom-trawling is one of the greatest threats to cold-water coral reefs.

[Back to top](#)

Figure 9: Brief explanation of coral reef.

Description of implementation

```
<h1><a name="top"></a>About Coral Reef</h1>
<ul>
    <li><a href="#how">What is a coral reef?</a></li>
    <li><a href="#how1">Types of Coral Reef Formations</a></li>
    <li><a href="#how2">What are the main threats to coral reefs?</a></li>
</ul>
```

```
<div class="main">
    <h1 name="how1" id="how1">What is a coral reef?</h1>
    <p> Coral reefs are made up of colonies of hundreds to thousands of tiny individual corals, called polyps. These marine invertebrate animals have hard exoskeletons made of calcium carbonate, and are sessile, meaning permanently fixed in one place. Polyps grow slowly, forming different shapes and sizes depending on their species.

    <br>Coral reefs provide an important ecosystem for marine life, offering food and shelter among their crevices and branches for animals including fishes, mollusks, sea urchins and sponges.

    </p>
    <a href="#top" class="button">Back to top</a>
</div>
```

Figure 10: Reef Information Page code.

On the reef information page, we add several internal links. If the users are interested in that particular topic, it will directly take them to the relevant information by clicking on the link. This saves the user time and effort so they do not need to scroll down the page manually, and they can quickly find the right information.

We also added a hyperlink at the top page so it would take the user to right content. And when the user finished reading the information we added “Back to top” button allowing users to go back to the top page. These added features can increase user experience and overall satisfaction.

```
.login-form-1 li a{  
    color: ■rgb(215, 244, 252);  
    font-size: 20px;  
    font-family: 'Roboto Slab', serif;  
    text-shadow:1px 1px 1px □rgb(75, 66, 52);  
}  
  
.login-form-1 li a:hover{  
    color:■rgb(255, 255, 255);  
    font-size: 23px;  
    text-shadow:2px 2px 2px □rgb(54, 54, 54);  
    transition: 0.3s;  
}
```

Figure 11: Hyperlink and Back to Top code.

```
.main a:hover {  
    background-color: ■#7687a5;  
    transition: 0.4s;  
}  
  
.main a:active {  
    background-color: ■#2a5f2c;  
    box-shadow: 0 5px □#666;  
    transform: translateY(4px);  
}
```

Figure 12: Hover and font size code.

In the Cascading Style Sheets (CSS) section, when users hover their mouse over a question that they are interested in, the question's font size will grow, and the colour of fonts and shadows will change accordingly. We have changed the button colours to indicate to the user that they can click in this area. This part CSS will make easier for our audiences to use our site by showing small interactions to help them, such as older or colour-blind users.

We use JavaScript to implement the route window. This allow us to access the API and to select useful data. By doing this we realize we could generate the search function and

routes.

```
$('.button1').click(function() {
  if ($.button1).css('display') == "none") {
    $('#infor').show();
  } else {
    $('#infor').hide();
  }
});

$('.button').click(function() {
  if ($.button).css('display') != "none") {
    $('#infor').show();
  }
});
```

Figure 13:Route API code.

This section mainly uses the "If" statement in JavaScript to influence the guide's display and hiding. In the first part, the ".button1" is the class of the "Continue" button in the guide. When the users enter our web page for the first time, the "if" statement will determine the "Continue" button's display. If the button's display is none, the guide will popup automatically. In contrast, the guide will hide after clicking the "Continue" button.

The second part is about the icon in the upper right of the web page. After the users click this icon, the "if" statement will determine the icon's display. The guide will then popup if the display of the icon is none.

```
var myMap = L.map("map").setView([-21, 149], 5.5);
L.tileLayer("https://api.mapbox.com/styles/v1/mengxuanzhang/ckg0jawvb0we919o7a7wv5aqv/tiles/256/{z}/{x}/{y}@2x?
access_token=pk.eyJ1IjoibWVuz3h1YW56aGFuZyIsImEiOijja2cwoXMwZjMyNTFqMnpwOTI0aWQ4YXUwIn0.9ynMtyT-AXaXa8j2qEqEdg", {
attribution: "Map data © href=\"https://www.openstreetmap.org/\">OpenStreetMap contributors, href=\"https://creativecommons.org/licenses/by-sa/2.0/\">CC-BY-SA, Imagery © href=\"https://www.mapbox.com/\">Mapbox",
maxZoom: 18,
id: 'mapbox/streets-v11',
tileSize: 512,
zoomOffset: -1,
accessToken: 'your.mapbox.access.token'
}).addTo(myMap);
var layerGroup = L.layerGroup().addTo(myMap);
var latlngArray = []
```

Figure 14:Accessing Leaflet Code.

In the main function, we first access the leaflet map API then initialise the centre point and the size. After that, we will set it to the main page. Also, we have create some global values that we will use in the following functions, as shown in the image.

```
var data = {  
    where: "1=1",  
    outFields: "*",  
    geometryType: "esriGeometryPolygon",  
    resultRecordCount : 300,  
    returnGeometry : true,  
    f: "pgeojson"  
}
```

Figure 15: iterateRecords code.

```
$.ajax({  
    url: "https://services8.arcgis.com/l11QQ2mI4WMXIXdm/ArcGIS/rest/services/Great_BARRIER_Reef_features/  
    FeatureServer/0/query",  
    data: data,  
    dataType: "jsonp",  
    cache: true,  
    success: function(results) {  
        L.geoJson(results, {onEachFeature: iterateRecords}).addTo(myMap);  
        console.log("Got reefs")  
    }  
});
```

Figure 16: iterateRecordsTide code.

Then JavaScript sets the GeoJson information and reef shapes onto the map. This will call the “iterateRecords” function after accessing the reef information API successfully by making an ajax request.

```
var data = {  
    where: "1=1",  
    outFields: "*",  
    geometryType: "esriGeometryPolygon",  
    resultRecordCount : 300,  
    returnGeometry : true,  
    f: "pgeojson"  
}
```

Figure 17: Ajax and API code.

```
$.ajax({
  url: "https://services8.arcgis.com/l11QQ2mI4WMXIXdm/ArcGIS/rest/services/Great_BARRIER_Reef_features/
  FeatureServer/0/query",
  data: data,
  dataType: "jsonp",
  cache: true,
  success: function(results) {
    L.geoJson(results, {onEachFeature: iterateRecords}).addTo(myMap);
    console.log("Got reefs")
  }
});
```

Figure 18: Ajax and API code.

Similarly, JavaScript makes an ajax request to access tide information API and calls the “iterateRecordsTide” function.

```
function iterateRecordsTide(data) {
  var currentTime = Math.floor(Date.now()/1000)
  var Time = Math.ceil((currentTime-86400) / 600) * 600

  $.each(data.result.records, function(recordKey, recordValue) {
    var recordSeconds = recordValue["Seconds"];

    if (recordSeconds == Time) {
      var longitude = recordValue["Longitude"];
      var latitude = recordValue["Latitude"];
      var recordTime = recordValue["DateTime"];
      var recordPrediction = recordValue["Prediction"];
      var recordSite = recordValue["Site"]
      var marker = L.marker([latitude, longitude]).addTo(myMap);
      popupText = "<p>Site: " + recordSite + "</p><p>Predicted Water Level: " + recordPrediction + "</p><p>Date &
      Time: " + recordTime + "</p>";
      marker.bindPopup(popupText).openPopup();
    }
  });
}
```

Figure 19: bindPopup and openPopup code.

In the “iterateRecordsTide” function, a selection is required for the most useful and straightforward information since the tide data is from a real-time dataset. After consideration and discussion, we have decided to choose only the site name, the predicted water level, and the time the prediction is made. We have filter the data by getting the

system's current time and looking for the data closest to 24 hours before.

```
function iterateRecords(feature, layer) {
    layer.layerID = feature.properties.UNIQUE_ID;
    layer.longitude = feature.properties.X_COORD;
    layer.latitude = feature.properties.Y_COORD;

    layer.bindPopup("<p>Unique ID: " + feature.properties.UNIQUE_ID + "</p><p>Reef Type: " + feature.properties.FEAT_NAME +
        "</p><p>Reef Name: " + feature.properties.GBR_NAME + "</p><button class = 'popupButton' id = '" + feature.
        properties.UNIQUE_ID + " addButton'>+add</button><button class = 'popupButton' id = '" + feature.properties.UNIQUE_ID +
        "+cancelButton'>-cancel</button>", {closeOnClick: true, autoClose: false}).openPopup();

    layer.on('popupopen',function(){
        $("#" + layer.layerID +"addButton").click(function(event) {
            event.preventDefault();
            console.log("add");
            if (confirm("I want to add this reef to my route.")) {
                addLatlng(layer.latitude, layer.longitude);
            }
            layer.closePopup();
        });

        $("#" + layer.layerID +"cancelButton").click(function(event) {
            event.preventDefault();
            console.log("cancel");
            if (confirm("I want to cancel this reef from my route.")) {
                cancellatlng(layer.latitude, layer.longitude);
            }
            layer.closePopup();
        });
    });
}
```

Figure 20: addLatlng and cancellatlng code.

In the “iterateRecords” function, we first set the layer ID to the instance's unique id so that the specific layer can be reached to get other parts to work. The “bindPopup” and “openPopup” functions of the layer are used to set the popup information and to achieve the click-to-popup. We set the popup window “autoClose” for the false searching function. When the popup information shows on the page, the users can use add and cancel buttons to plan the destination and generate their route. Once the user confirms to add/cancel a reef, the “addLatlng/cancellatlng” function will be called forth.

```
function addLatlng(latitude,longitude) {
    var rooffAdded = false;
    for (let index = 0; index < latlngArray.length; index++) {
        const element = latlngArray[index];
        if (latitude == element[0] && longitude == element[1]) {
            alert("This reef has been added.");
            rooffAdded = true
        }
    }

    if (!rooffAdded) {
        var latlng = [latitude,longitude];
        latlngArray.push(latlng);
        latlngArray.sort(([a,b],[c,d]) => b-d || a-c)
    }

    console.log(latlngArray);
    drawLine(latlngArray);
}
```

Figure 21: addLatlng code.

In the “addLatlng” function, we determine whether the reef has been added by checking the latitude-longitude array elements going through a for loop. And if it has been added, an alert will be thrown. Otherwise, we will add it into the latitude-longitude array and sort the array by longitude descending. Sorting is for the routing function showing by the “drawLine” function to be called.

```
function cancelLatlng(latitude,longitude) {
    var cancelIndex = -1;
    for (let index = 0; index < latlngArray.length; index++) {
        const element = latlngArray[index];
        if (latitude == element[0] && longitude == element[1]) {
            cancelIndex = index;
        }
    }

    if (cancelIndex != -1) {
        latlngArray.splice(cancelIndex, 1);
    }

    console.log(latlngArray);
    drawLine(latlngArray);
}
```

Figure 22: cancelLatlng code.

In the “cancelLatlng” function, we can determine whether the coordinate is in the array or not. If it is in the array, then we remove it. The “drawLine” function will then be called.

```
function drawLine(latlngs) {
    layerGroup.clearLayers();
    L.polyline(latlngs,{color: 'white'}).addTo(layerGroup);
}
```

Figure 23: drawLine and L.polyline() code.

As the latitude-longitude array is sorted, the “drawLine” function clears all the lines to remove the cancelled points and draw it again by the “L.polyline()” function of leaflet with the new points. It shows the straight line between all the spots added.

```
$('#search-by-reefID').keyup(function() {
    myMap.eachLayer(function(layer) {
        layer.closePopup();
    });

    if (this.value.length > 1 ) {
        let input = this;

        myMap.eachLayer(function(layer) {
            if (typeof layer.layerID == 'string' && layer.layerID.includes(input.value)) {
                layer.openPopup();
            }
        });
    }
});
```

Figure 24: closePopup code.

When the user clicks the search bar, all the popups will be closed first by calling each layer and closing any popups on that layer. Once the user enters greater than or equal to than two numbers of the unique ID, all the reefs with the numbers in the unique ID will pop-up a window to show the information. We choose two because searching by only one number is meaningless. The data loading process is timely and may leads to lagging problem. We check the layer ID (set as the unique ID) by if statement and popup those including the input numbers.

Third party libraries/frameworks/code

1. Leaflet - Mapping features
<https://leafletjs.com/>
2. Mapbox - overrides map style
<https://www.mapbox.com/>

Final product link

<https://deco7180teams-pfc01t01.uqcloud.net/>

Discussion of issues and major challenges:

Throughout the whole development process, we encountered one challenge after another, so we produced various versions of the website to overcome these challenges.

Challenge 1: How to divide up the code tasks and make the teamwork efficient?

Version 1.0: There is only a basic structure with very poor design.

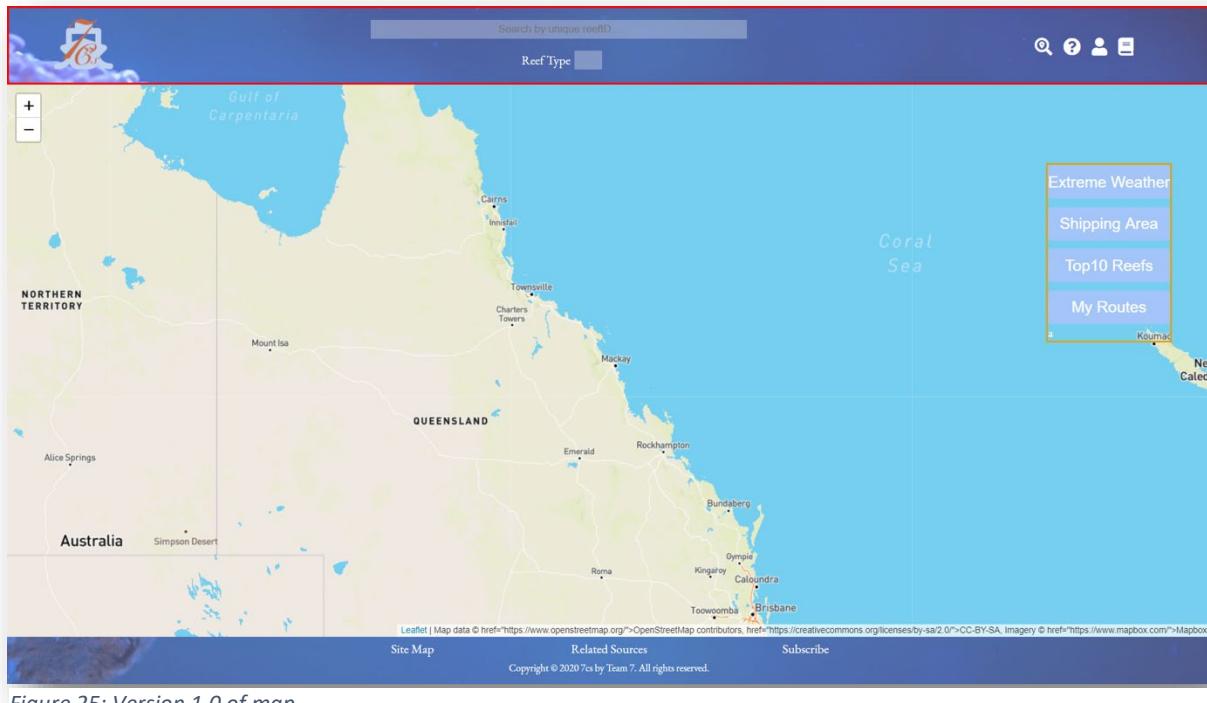


Figure 25: Version 1.0 of map.

At this stage, we have just finished Part A and we started to work on the implementation of the website. The challenge we encountered at this point is that we do not know how to divide up the code tasks and keep the teamwork efficient.

We discussed about it and we decided to implement the basic structure of the whole website first, regardless of any aesthetics or design style. This allows some of us to adjust the visual organization of the entire website and some of us to develop the functionality of the website at the same time.

Once the website was completed. We looked back on the decision that we made at this point; it seems that we made a very wise decision. Because we have separated the visual presentation and the functional part of the website, which means we can develop these two parts at the same time. And when a member had an issue trying to resolve a particular code, we would solve that problem together before going back to work on each person's own part. Based on this strategy, we can quickly communicate and modify the code and push the process of the project.

Challenge 2: The API for the main function is different from what we learned in class

Version 1.1: We successfully display the locations of reefs on the map.

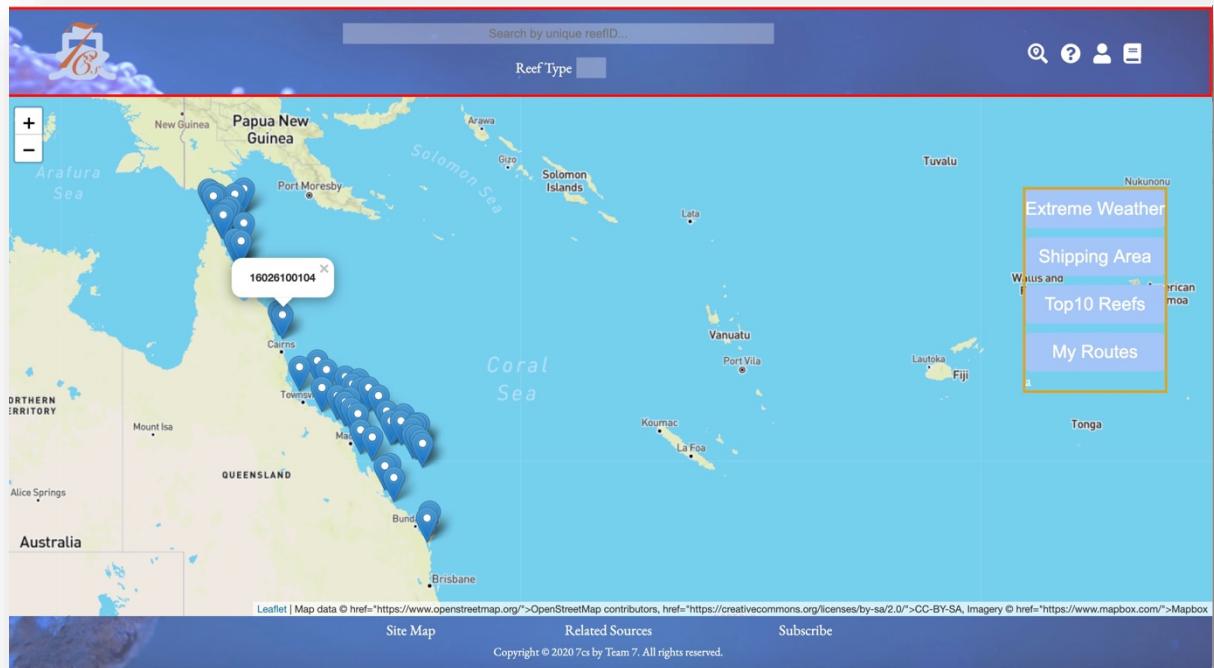


Figure 26: Implementation of reef locations.

Since we have learnt how to put markers on the map in our workshop, we then applied that method to our main function which is displaying the location of reefs on the map by markers.

However, we encountered some challenges and the issue was the function was not that easy as we had imagined. Because the API for the main function is different from what we have learned in class. The API is ArcGIS REST API and it serves GeoJSON, which we did not

know anything about before.

The screenshot shows a map of South America and Africa with a search bar at the top right. Below the map are three tabs: Overview, Data (selected), and API Explorer. Under the API Explorer tab, there is a date range (4/15/2020 - 4/15/2020), a Feature Layer link, and a Custom License link. To the right, there are Download and APIs buttons. A sidebar titled 'Query URL' contains links for Full Dataset, GeoService (with a URL), and GeoJSON (with a URL). The main area has sections for Query, Where (No Active Filters), and Out Fields (checkboxes for OBJECTID, SORT_GBR_ID, LABEL_ID, SUB_NO, CODE, UNIQUE_ID, FEATURE_C, GBR_NAME, FEAT_NAME, QLD_NAME, and X_LABEL).

Figure 27: API code from ArcGIS.

So, we decided to work on this function together. We learned by Googling related resources and reading the official documentation provided by ArcGIS. After many attempts, we managed to implement this feature. Although this function was realized by one of us at last, the discussion on this function and exchange of ideas in the team during the process helps a

lot.

```
function iterateRecords(results) {
    console.log("hello");

    var myMap = L.map("map").setView([-21, 149], 5.5);

    L.tileLayer("https://api.mapbox.com/styles/v1/{id}/tiles/{z}/{x}/{y}?access_token=pk.eyJ data © href="https://www.openstreetmap.org/">OpenStreetMap contributors, href="https://creativecommons.org/lice

        maxZoom: 18,
        id: 'mapbox/streets-v11',
        tileSize: 512,
        zoomOffset: -1,
        accessToken: 'your.mapbox.access.token'
    }).addTo(myMap);

    $.each(results.features, function(recordID, recordValue) {
        var x_coord = recordValue["attributes"]["X_COORD"];
        var y_coord = recordValue["attributes"]["Y_COORD"];
        // console.log(y_coord)
        var marker = L.marker([y_coord, x_coord]).addTo(myMap);

        popupText = "<strong>" + recordValue["attributes"]["UNIQUE_ID"] + "<br>";
        marker.bindPopup(popupText).openPopup();
    });
}

$(document).ready(function() {
    console.log("hello2");
    var data = {
        where: "1=1",
        outFields: "*",
        geometryType: "esriGeometryPolygon",
        resultRecordCount : 100,
        returnGeometry : true,
        f: "json"
    }

    $.ajax({
        url: "https://services8.arcgis.com/ll1Q02mI4WMXIxdm/ArcGIS/rest/services/Great_BARRIER_Reef_features/FeatureServer/0/query",
        data: data,
        dataType: "jsonp",
        cache: true,
        success: function(results) {
            console.log("GGEZG");
            iterateRecords(results);
        }
    });
});
```

Figure 28: Implementing API via ArcGIS.

Challenge 3: We think only showing the location is not good enough.

Version 1.2: We change the way we implement the main function.

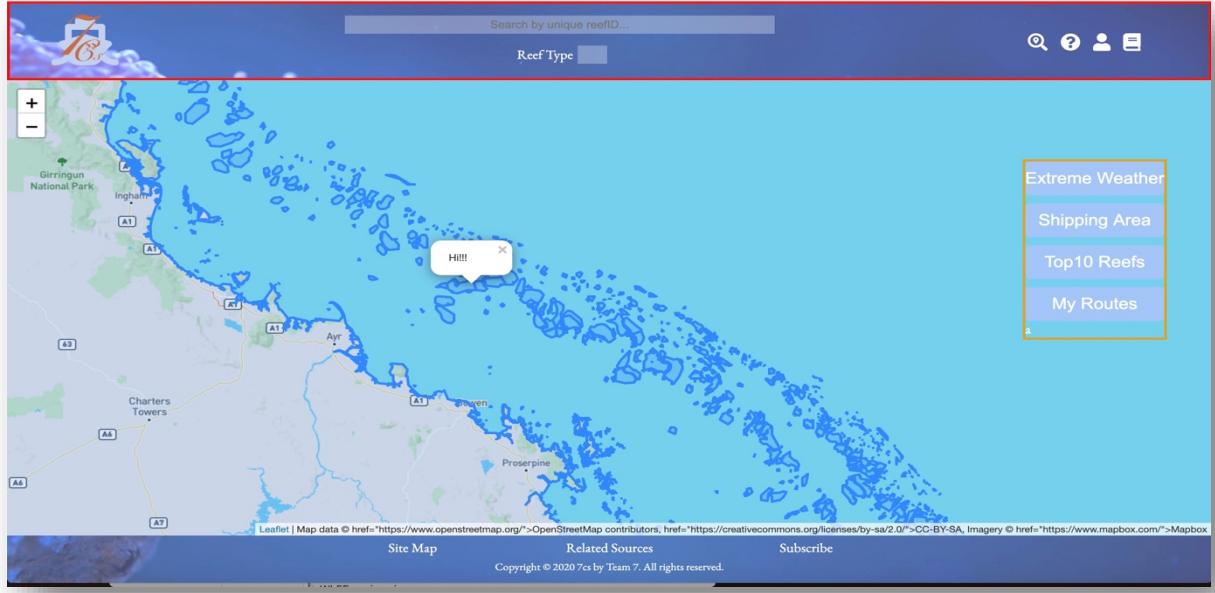


Figure 29: Successful intergration reefs on map.

After we successfully displayed the location of reefs on the map, we encounter a problem. And that problem is the dataset for tide level only provides the coordinates of the detection site, which means that we could only represent the tide level on the map by markers. However, we have already used markers to show the locations of reefs. This would make our website looks messy with too many markers. Furthermore, we found that if we only show the location of reefs on the map, it provides too little information to the user.

Luckily, we found that the dataset for reefs includes the information of the shape of each reef. So, we decided to show every reef's shape on the map instead of just showing the coordinates. After further exploration of GeoJSON, we succeeded by showing the shape of each reef on the map. This allow users to see the size and shape of each reef directly on the map, which made our website more useful.

```
$(document).ready(function() {
    console.log("hello2");
    var data = {
        where: "1=1",
        outFields: "*",
        geometryType: "esriGeometryPolygon",
        resultRecordCount : 100,
        returnGeometry : true,
        f: "pgeojson"
    }

    $.ajax({
        url: "https://services8.arcgis.com/l1l0Q2mI4WMXIXdm/ArcGIS/rest/services/Great_BARRIER_Reef_features/FeatureServer/0/query",
        data: data,
        dataType: "jsonp",
        cache: true,
        success: function(results) {
            console.log("GGEZG");
            L.geoJson(results).addTo(myMap);
        }
    });
});
```

Figure 30: GeoJSON code.

Challenge 4: The default colour of leaflet map limits the colour theme of our website

Version 2.0: We use map box to generate map with our own style.

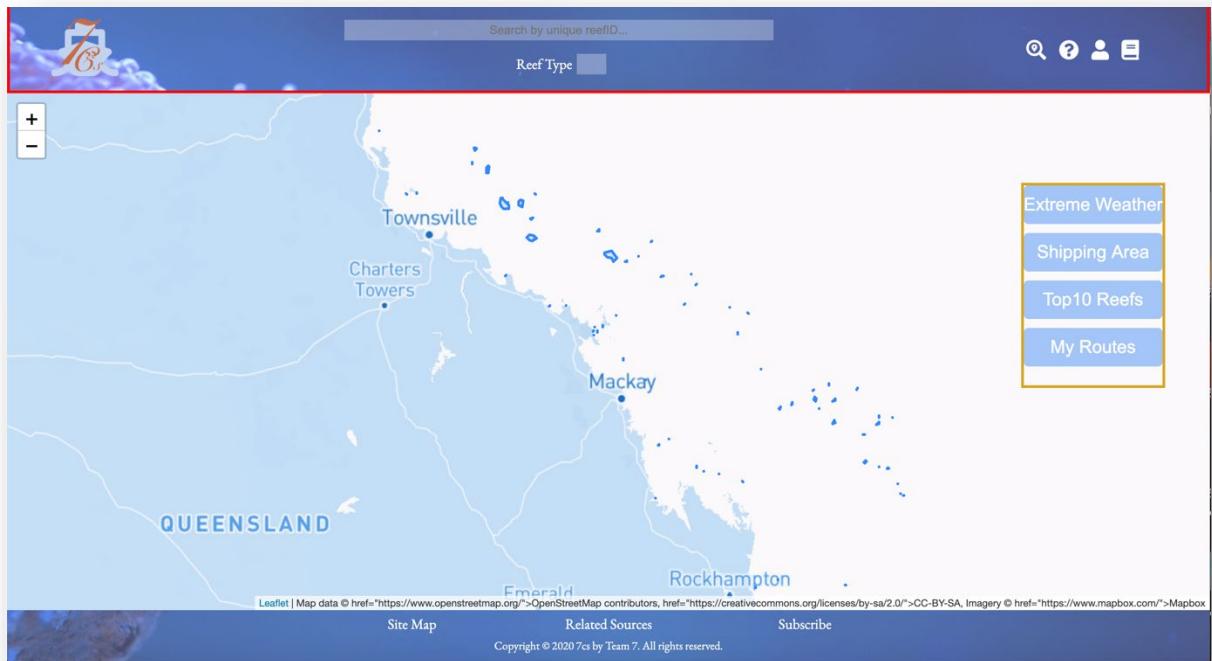


Figure 31: Trial using Map Box.

After we implemented the main function of our website. We started to think about aesthetics. We encountered the challenge that the default color of leaflet map made us unable to adjust the color of our website too much, since the map made up almost 80% of the index page. We discussed about it and tried to find some methods to adjust the colour of the map. After doing some research and experimentation, we succeeded to use a “mapbox” style created by ourselves along with leaflet map. Then we started experimenting with various map styles.

Challenge 5: Light themes and low-contrast colours that are unfriendly to people with vision impairment or colour blindness.

Version 2.1: We tried several different colour themes and tested in the group.

After we made the version 2.0, one of our group members who is highly myopic said she felt that light blue color theme was not good. Because she has severe dry eye syndrome, her eyes started to be uncomfortable after using the website for more than 10 minutes. And another member of our group has colour blindness, he felt he could not see things clearly with the light blue color theme.

So, we tried several different color themes and tested them out in our group.

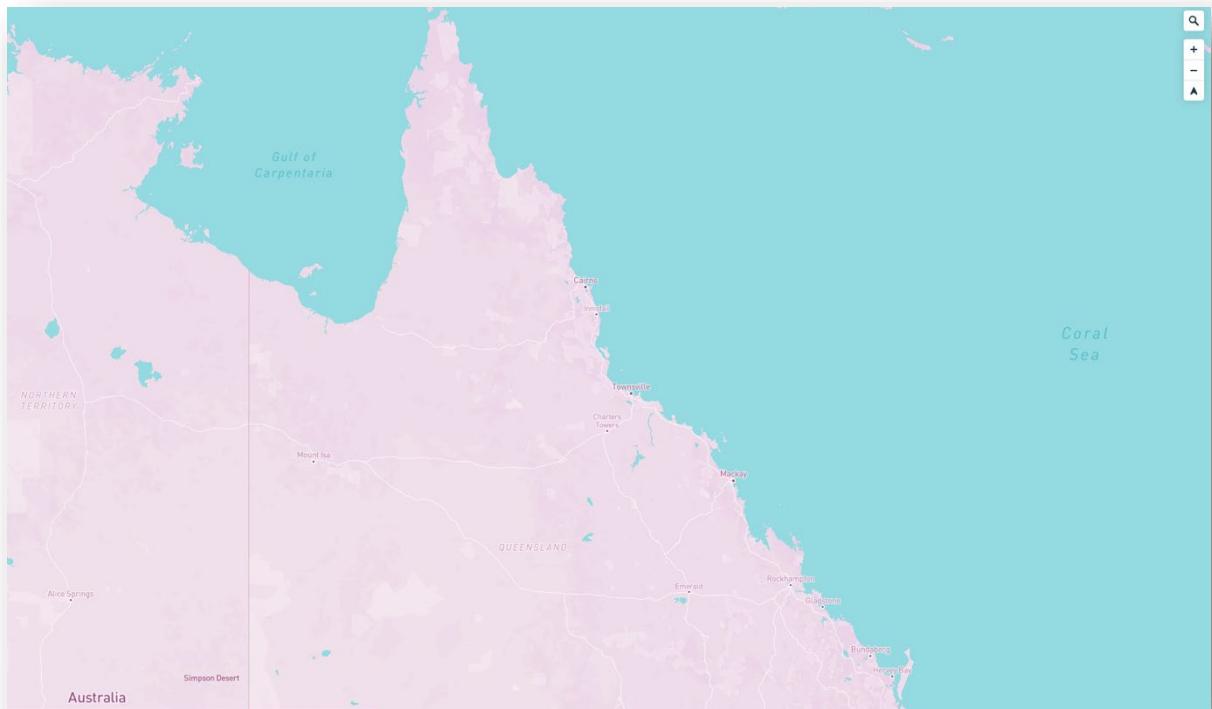


Figure 32: Different types of colour themes, light colour theme.

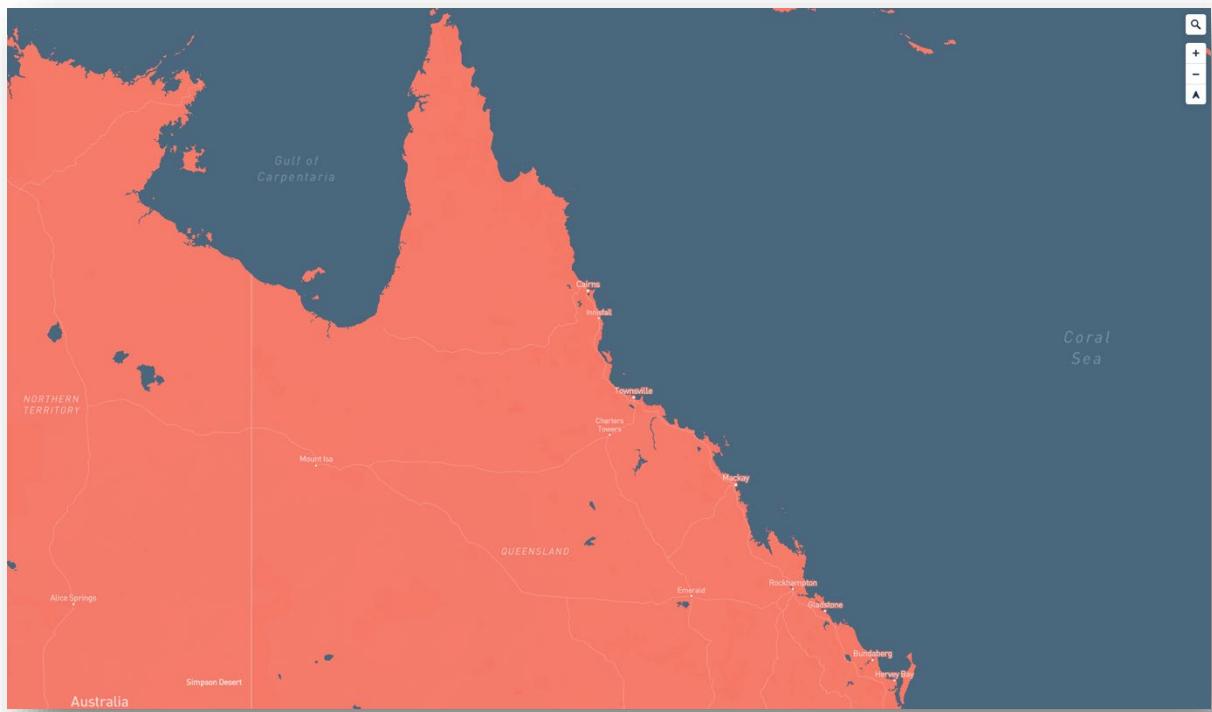


Figure 33: Different types of colour themes, medium dark colour theme.

Test results show that the person who is highly myopic feels more comfortable with dark colour theme especially when she needs to use the website for more than 10 minutes. And the person who has colour blindness feels comfortable when there are few colors on the entire page. The strong contrast between colour makes it easier for user to use the website. Therefore, we decided to use dark color for most part of our website and make sure the color contrast is obvious enough.

```
.main h1{
    font-size: 30px;
    font-family: 'Courgette', cursive;
    text-align: left;
    color: ■rgb(200, 246, 252);
    text-shadow: 3px 3px 3px □rgb(43, 43, 43);
}

.main p{
    padding-left: 25px;
    color: ■#ffffff;
    font-family: 'EB Garamond', serif;
    text-shadow: 1px 1px 1px □rgb(56, 56, 56);
}

.main a{
    padding: 10px 20px;
    border: none;
    background-color: ■rgb(171, 198, 199);
    color: ■white;
    text-align: center;
    text-decoration: none;
    display: inline-block;
    cursor: pointer;
    border-radius: 10px;
    box-shadow: 0 4px 8px 0 □rgba(0, 0, 0, 0.3), 0 6px 20px 0 □rgba(0, 0, 0, 0.3);
    margin-left: 85%;
}
```

Figure 34: Trying different types of colour code.

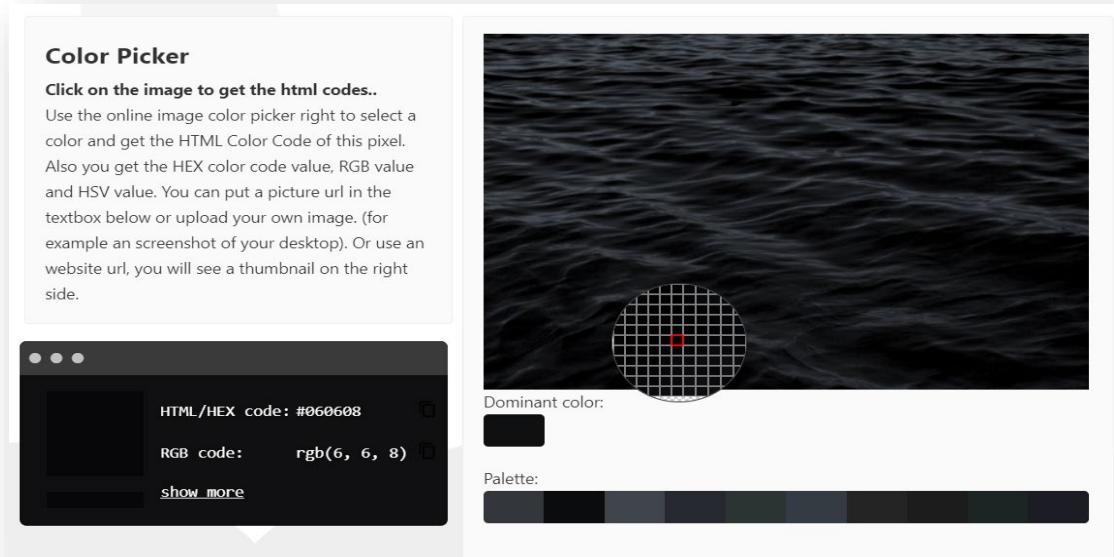


Figure 35: Image colour picker.

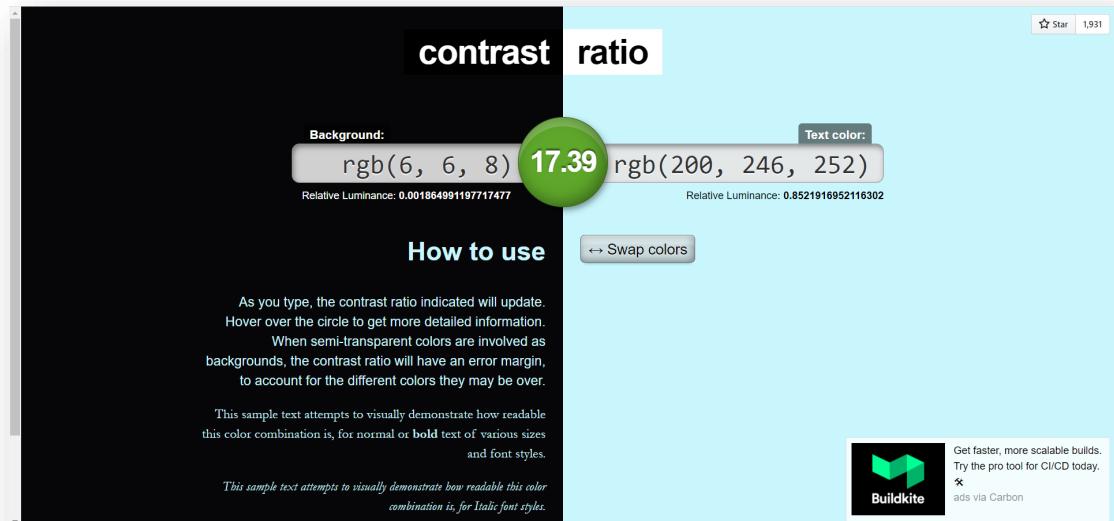


Figure 36: How to use contrast ratio control.

A dark-grey background and a light blue or white font are used on the overall colour palette and the web page's selection. The contrast ratio is measured on professional websites at 17.39, which altogether indicates that the web pages' contrast ratio is quite obvious and user-friendly. Aesthetics is a factor in the colour palette of the site. However, the most important element is the website can help more users through the elaborate design.

Considering that the reefs are all in the sea, users will spend most of time looking at the sea when using our website, so we decided to use dark grey for the sea and a contrasting light blue for the land, instead of the traditional map where the water is blue and the land is light beige, to make our users feel more comfortable when using the website, especially those

with visual impairments or colorblind.

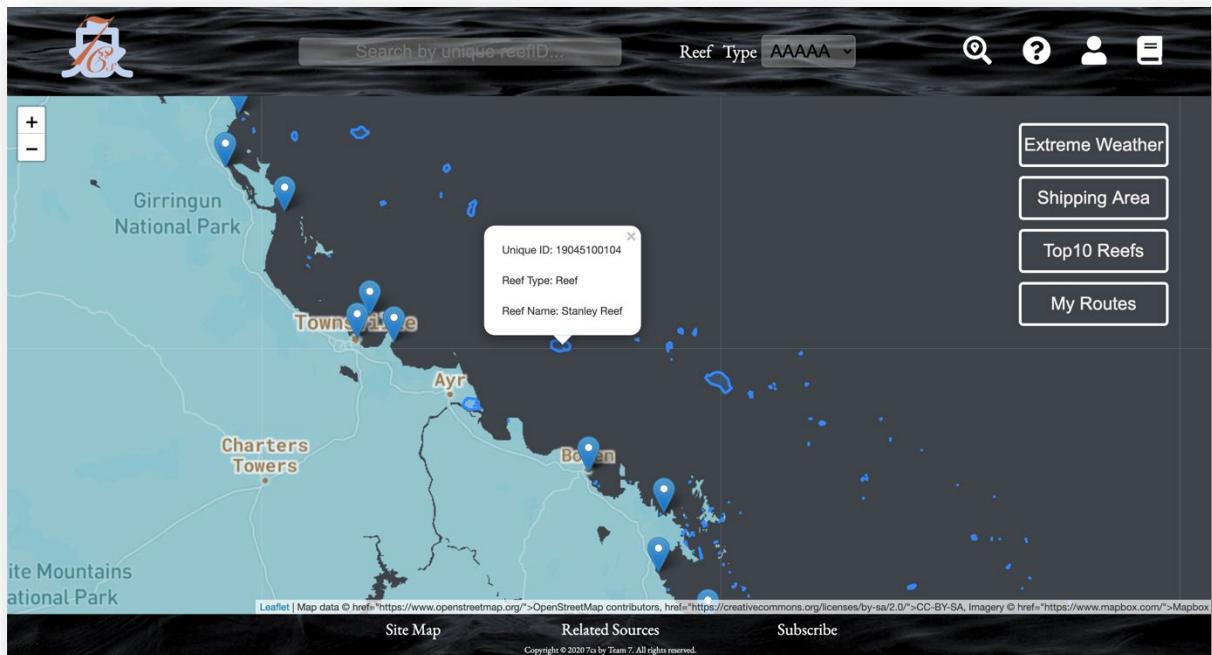


Figure 37: Home page, 1st iteration of popup.

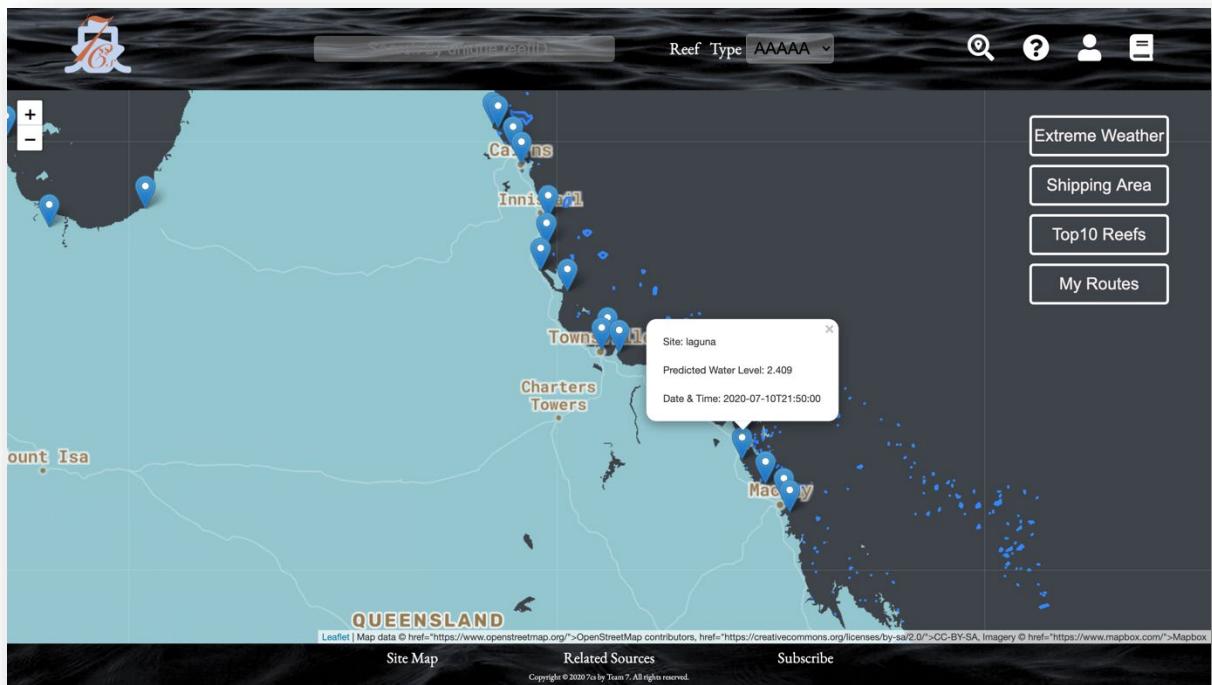


Figure 38: Home page, 1st iteration of tide levels popup.

Challenge 6: The algorithm of routing function

Version 3.0: We tried different ways to display the route.

At this stage, we started to implement the routing function. At first, our algorithm was very simple. When user clicked “+add” in the popup window, the reef’s coordinate would be added to an array. And when user clicked “-cancel”, the reef’s coordinate would be deleted from the array. This algorithm only checked duplicate coordinates and joined these coordinates in the order in which they were added by the user.

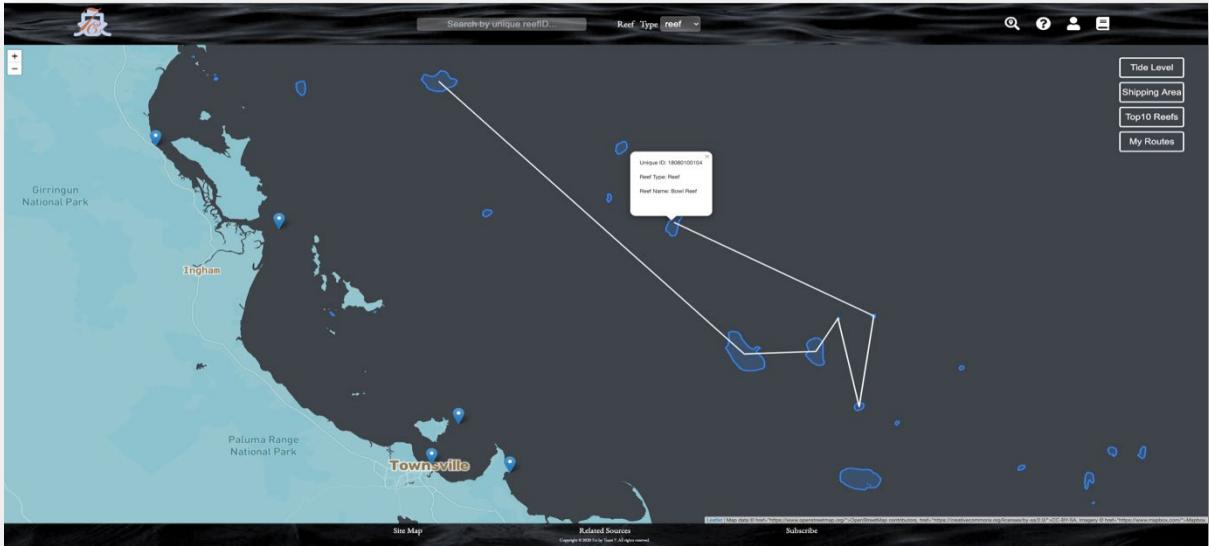


Figure 39: Home page, 1st iteration of routing function.

However, this algorithm caused some issues. For example, the user might get a route like the following one.



Figure 40Home page, 2nd iteration of routing function.

So, we tried to figure out how to generate a route planned by the user on the map like Google Map. After doing some research, we found a plugin of leaflet called Routing Machine. However, when we tried to use this plugin, an issue arose that Routing Machine could not generate a route on the sea, it could only generate a route on the land.



Figure 41Home page, 3rd iteration of routing function.

Therefore, we gave up using Routing Machine and went back to work on our own algorithm instead. We sort the reefs in the array list based on the rule “from left to right, then from top to bottom”, because the reefs are distributed by the coastline. When the user adds a reef to his route, the coordinate of that reef will be added to the array list and the array list will be sorted based on this rule, and then use “`L.polyline()`” of leaflet to draw the route.

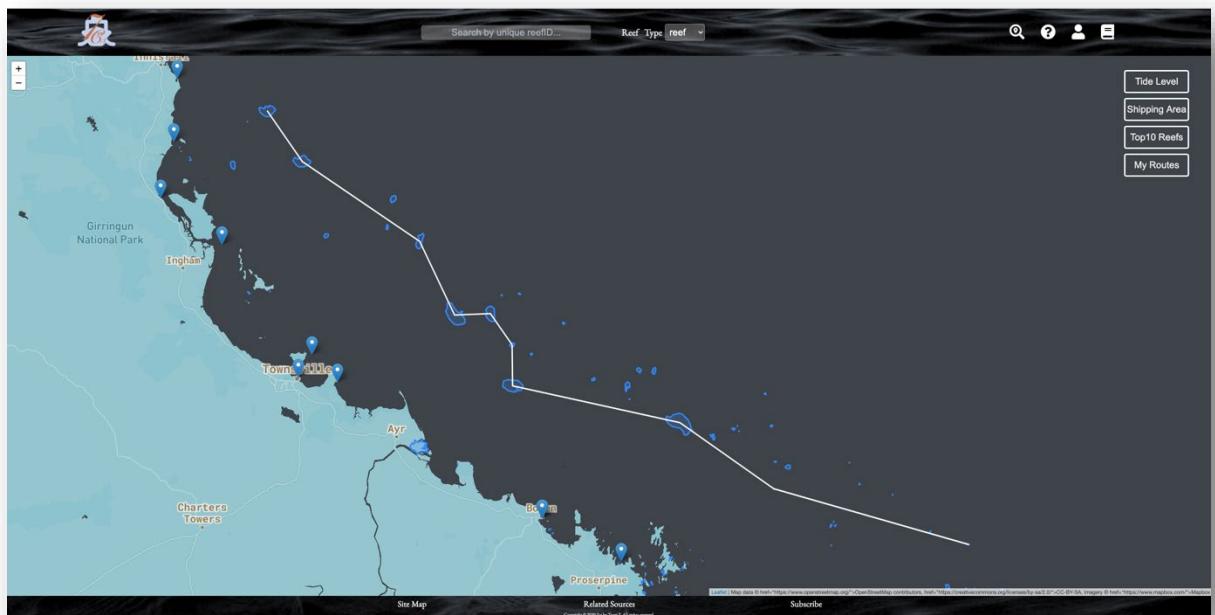


Figure 42Home page, 4th iteration of routing function.

A response to the feedback

Q1

In Part B, we designed four buttons for interactive operation.

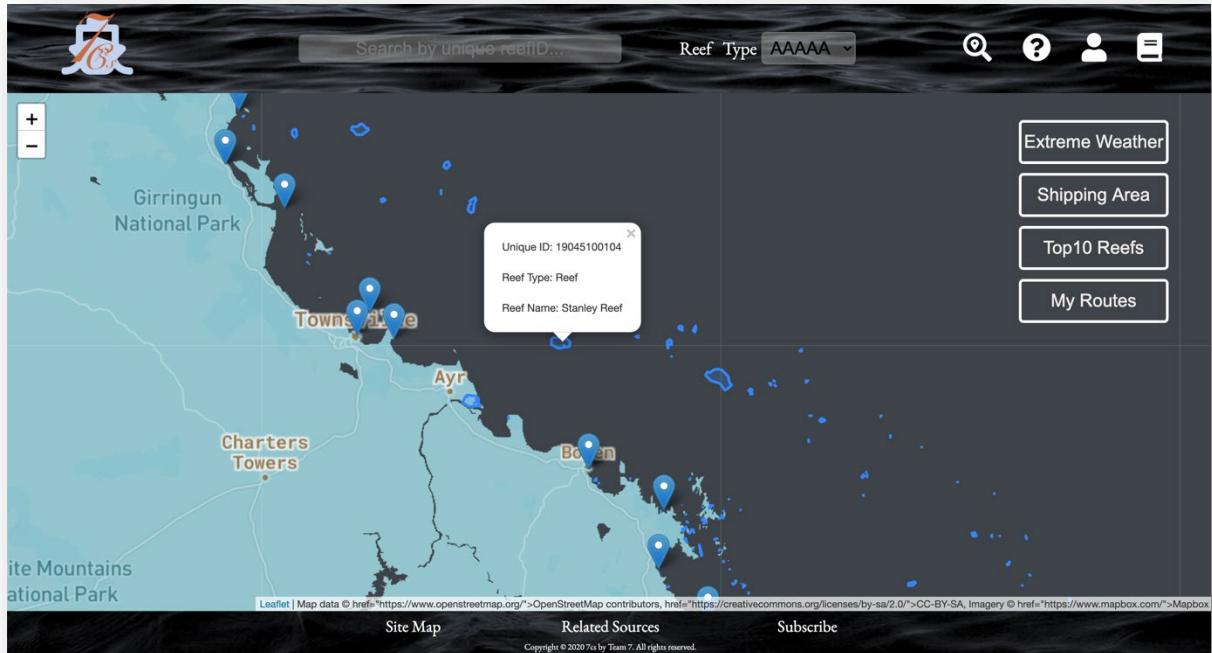


Figure 43:Home page, 1st iteration of button icons.

However, after the presentation we received a lot of critique from our peers to help us improve our website. The most critique we received was about the button design. The feedback from our peers thought we had too many buttons, this would make the web page become more clutter and less user-friendly. The second is about the size of font, the font size button is a little small, which could affect the user's usability.

Anonymous 15d

45855742 guangxin su

1. there are really many buttons to be known
2. i like the color style
3. yean I think i will have a try
but for 4: my grandparents, I don't think they are easy to use it

Anonymous 15d

It is a very good website. But maybe the font size it a bit small.

45652576 Zixin Wang

Add comment

Anonymous 15d

Qihui Chan 45980961

1. yes
2. I think so, it is proper for the theme.
3. 4
4. I guess so, but I guess the layout or the font size could be improved a bit.

Anonymous 15d

45693748- Yizhou Chen

2.The color you used in main page is amazing,I like it.
3. I am afraid that I won't use a website like this if I want to travel, since it is a little bit complicated, so I rate 2.
4. Honestly, I think it would be a little bit hard for them to distinguish all fonts on your website, since the font size on buttons are too small to read for the elder.

Add comment

Figure 44: Feedback and critique from peers.

In Part C, we removed the all the buttons and made an icon which is situated at the top right-hand corner of the website. We also increased the font size of the search Bar to

improve the readability.

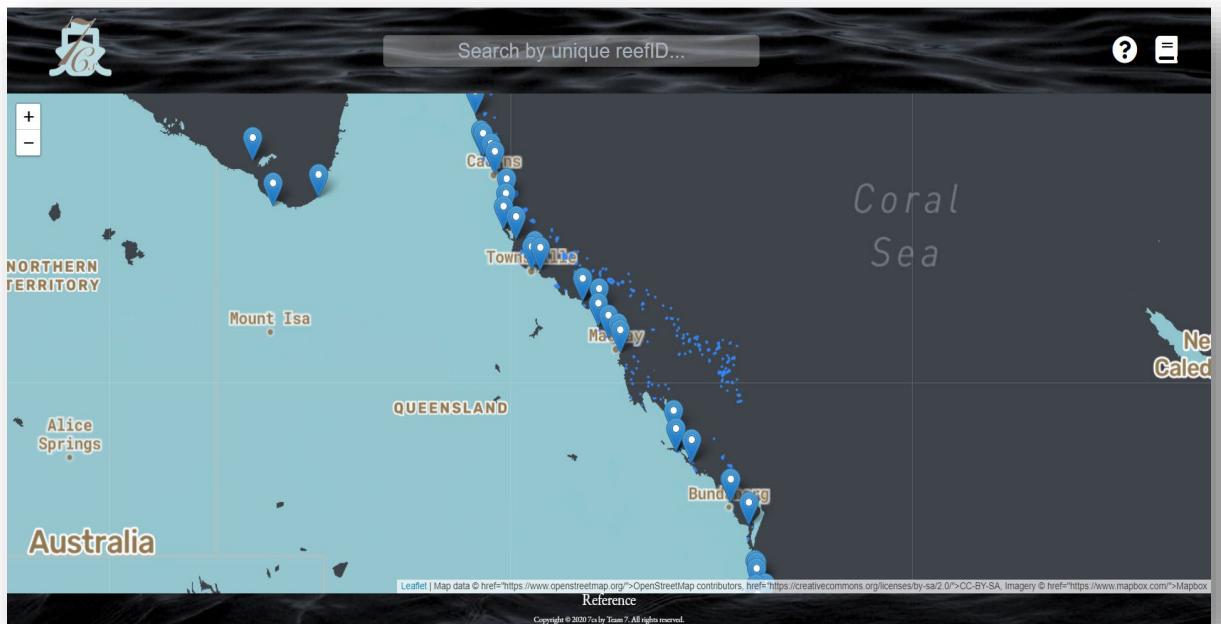


Figure 45: Home page, 1st iteration removal of buttons.

Q2.

We chose light blue and dark grey as the main colours of our website in Part B. After receiving critique feedback from our peers, they thought that the dark grey and black background would make the whole page too dark. Our peers suggested that we use a warm colour tone or something similar. After countless hours of testing and feedback, we decided to keep the original colour collocation because of the feedback we received and also to keep

our audiences happy.

 Anonymous 15d

Xinguo Feng 45850699

1. I think it's a bit complex. There are a few buttons and drop down menu.
2. I think it's a bit dark. I think light blue might be more positive.
3. Probably 3.
4. Not really. Like I said in my feedback for the 1st question, I think there are a few buttons that don't come very self-explanatory.

Runqi Wang 4576174

- 1 Very nice buttons and navigation bar, black and blue are very suitable choices for the display of site map distribution.
- 2 As what I mentioned before black and blue are pretty suitable colour for your website demo
- 3 ,3 I suppose cause I felt a little confuse about how to record my trip.
- 4 A little bit hard for grandparents generation I THINK..

 Add comment

Jiayang Li 44143862

- 1.I agree that the website is user-friendly and easy to navigate.
2. Personally, I think that will be better to change the grey colour into another light colour.
3. 4
4. It is difficult to answer, for some old people that is even hard for them to use the computer and internet. But I believe the functions of the website is pretty clear.

1. The moment the user enter the page, they may face a lot of buttons, which can be a bit confusing.

2. The theme of this dark style is very good.

3. I will give you 4. This website is really attractive.

4. It may be a little difficult to get older people to understand how the site works, they have a lot of information to deal with.

Figure 46: Feedback and critique from peers.

Another reason we chose these colours is because one of the team members is myopic. Their eyes will feel uncomfortable if they spend a long-time browsing website with bright or warm colour tone themes. On the contrary, pages with large colour contrast and dark themes will allow the user to use the pages for a long time. Additionally, this is one of the reasons why many programmers choose black instead of white as the background colour when they write their code. The second reason is that for people who are colour-blind, it is difficult for them to distinguish warm colours, such as red and pink. Considering the needs of these users, we finally chose this colour scheme.

We also received feedback from our tutors about the colour of our map. They suggested that landmark should be dark-grey and sea should be light-blue colour. We consider this issue before we design our website. However, after long consideration and research, we rejected this proposal because our purpose was to allow users to perform relevant operations on the ocean rather than on the land. For normal navigation app, they focus on the land. Conversely, we focus on the sea. If users stare at the light blue part for a long time, their eyes will feel tired.

Q4.

During our presentation, we mentioned to our peers whether they think that their grandparents could use our website proficiently. We chose them as a particular audience because a lot of elderly people are not good at using electronic devices and new technology. We want to make our website user-friendly and straightforward.

4. First of all, I believe that old people may find difficulties with dark colors because it's hard to see images in details and read the information. Furthermore, if you target on old people such as grandparents, then they may need a navigation guide everytime they use the website, since they are not familiar with technology as well as have mental problems related to memory.

4 I think for some people it may easy to understand the method of using main function but others may not. the guide is necessary.

4 They can understand all the functions by following some guides offered by website.

Figure 47: Feedback and critique from peers.

We got excellent feedback in which we added a simple guide that explained how to use our website. In the end, we designed a simple icon on the right side of the navigation bar to help

users review the information in the guide after they click on the icon.

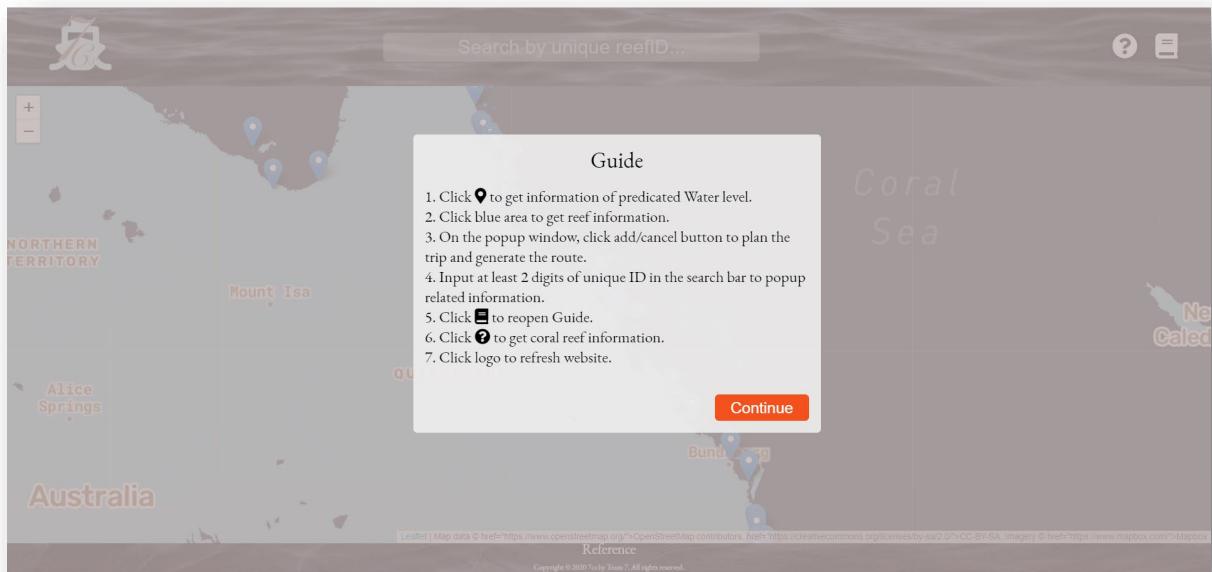


Figure 48: Guide and information page.

Reflection

The Likes and Dislikes.

Our team is delighted and happy with the design, layout and user-ability, even though it was not as elegant as we hoped it to be. Our team really liked how the website looks and feels. The website interface has seamless and smooth appearance. Our goal is to make user-friendly and accessible website for all ages.

What we liked about our website was the darkness feel of the map. We received a lot of feedback from users who loved the dark theme of the map. We had particular reason for choosing this colour theme. Colour Blindness or Colour Vision Deficiency (CVD) affects approximately 1 in 12 men and 1 in 200 women in the world. With that in mind, we decided to choose the dark theme that would be easy on people's eyes who have Colour Blindness or CVD (Colour Blind, 2019). Another reason why we chose dark water theme was to give the user a sense of deep water feeling. For example, when you go scuba diving and as you go deeper and deeper into the bottom of the sea floor the water gets darker and darker, so we wanted our users to have that warm and dark feeling.

Another feature we loved about our website is the simplicity of the interface with only couple icons and one search bar. People do not like websites that have a lot of icons, buttons and images that clutter the website, it makes the user feel lost and sometimes anguished because of the amount of information that is thrown at their face.

Our team agreed the search bar did sometimes have slight lag/delay due to the amount of information that is been retrieved from the API databases because we set the minimum limit to two character to start search engine for the Unique Reef ID.

Successful and Unsuccessful Aspects.

Overall, our team was very satisfied with the final design and achieving our main objectives. The smoothness of the website, the colours and lay of the map was total success. Aesthetically the map and icons are pleasing to the eyes, especially if the users have Colour Blindness or CVD. The reefs were highlighted with nice blue colour behind dark background making it easier for the user to identify that particular reef. The routes were another success, letting the user to create their own routes from reef to reef. We created the shortest route from point A to B to C and so on by incorporating the SQL database code in unique way to make this happen.

Another success was the removal of buttons. We just implemented search bar for users to find particular reef ID (this feature is mainly used by experts and academics who specialise studies in coral reefs). Also, we implemented real time reef locations where users can click on that particular reef and find the information that they require.

We did regret we could not get routing to work like Google Maps. We could get the routing to work by using latitude (shortest distance), whereas Google Maps lets the user to add waypoints the way they want it instead of using shortest distance.

Improvements.

There were some improvements that we wanted to be implemented but due to lack of time they were not implemented in this website.

1. At the bottom of the page of the website we will add “Related Searches” link. This is an external link to other websites that caters coral reef tours. By adding this feature, we can generate income by promoting their businesses that organises coral reef tours.
2. We originally wanted to have of lush-green colour for Australia map and blue for the ocean on our website. At the top right-hand corner there will be button that will switch to colour blind mode for people with Colour Blindness or CVD (the map colour would be black and white as colours).
3. More sophisticated routing feature allowing users to choose the route the way they want it. For example, the user can choose their 1st route which maybe 20km away and the 2nd route which is 10km away and 3rd route is 40km away. Our routing feature is by shortest distance using latitude, so the 1st route would be 10km, 2nd route 20km and 3rd route 40km. Our routing feature will not allow users to choose the way they want to go.
4. Adding language feature to the website, a lot of users will be tourist and English may not be their first language. Therefore, adding different languages will be beneficial to us because we will have more users using our website.
5. Social media icons, having social media icons will let our business connect with our customers and easy way to share information.
6. Login page for returning customers. Some customers want to know which coral reefs they have visited. By having this feature returning customers can view and update their routes for future travel.

Chose not to Implement.

Our team chose not to integrate the “Storm Warning Icon”, “Colour Layers” and “Tide Button”. We could not add the storm colour layers onto our map due lack of datasets to support it. Without the relevant API dataset for storm warning we could not implement the storm layer colours into our website shown in figure below. Also, we could not implement tide button because tide button is incoherently connected to storm warning.

Additionally, adding real time radar over the map will make the map slow and sluggish due to information been streamed from API databases. Also, the website will be cluttered with clouds and rain making the website less appealing. As a result, we decided to remove “Storm Warning Icon”, “Colour Layers” and “Tide Button” from the website. In the future we

more relevant data, we will be able to implement into our website.

Parameter	Code	Produces	Use for
unknown (no parameter)	<code>{{Storm colour unknown}}</code>	<code>c0c0c0</code>	a storm with unknown wind speed
	<code>{{Storm colour }}</code>	<code>c0c0c0</code>	
Classification used in Atlantic and East-Central North Pacific basins			
cat5 . 5	<code>{{Storm colour cat5}}</code>	<code>#ff0000</code>	a storm with at least 157 mph (252 km/h) 1-minute sustained winds (Category 5 hurricane or equivalent)
cat4 . 4	<code>{{Storm colour cat4}}</code>	<code>#ff8200</code>	a storm with 130–156 mph 1-minute sustained winds (Category 4 hurricane or equivalent)
cat3 . 3	<code>{{Storm colour cat3}}</code>	<code>#ffcc40</code>	a storm with 111–129 mph 1-minute sustained winds (Category 3 hurricane or equivalent)
cat2 . 2	<code>{{Storm colour cat2}}</code>	<code>#ffe775</code>	a storm with 98–110 mph 1-minute sustained winds (Category 2 hurricane or equivalent)
cat1 . 1	<code>{{Storm colour cat1}}</code>	<code>#ffffcc</code>	a storm with 74–95 mph 1-minute sustained winds (Category 1 hurricane or equivalent)
storm . TS subtropical . SS	<code>{{Storm colour storm}}</code>	<code>00faf4</code>	a storm with 39–73 mph 1-minute sustained winds (tropical storm or equivalent)
depression . TD subdepression . SD	<code>{{Storm colour depression}}</code>	<code>66baff</code>	a storm with under 38 mph 1-minute sustained winds (tropical depression or equivalent)
potential . PT	<code>{{Storm colour potential}}</code>	<code>80ccff</code>	a tropical disturbance with a high chance of becoming a tropical cyclone within 48 hours, and poses a threat to land
Post-tropical . POST Remnant low . RL	<code>{{Storm colour post-tropical}}</code>	<code>cccccc</code>	a system that was a tropical cyclone and has since become a post-tropical cyclone or a remnant low
Classification used in Western North Pacific basin			
typhoon . TY	<code>{{Storm colour typhoon}}</code>	<code>#ffaf9a</code>	a storm with 84 knots (119 km/h; 74 mph) 10-minute sustained winds and above
nwpsevere . strong . STS	<code>{{Storm colour nwpSTS}}</code>	<code>ccffff</code>	a storm with 48–63 knots (59–87 km/h; 56–72 mph) 10-minute sustained winds
nwpstorm . TS	<code>{{Storm colour nwpstorm}}</code>	<code>00faf4</code>	a storm with 34–47 knots (43–57 km/h; 39–54 mph) 10-minute sustained winds
nwpdepression . TD	<code>{{Storm colour nwpdepression}}</code>	<code>66baff</code>	a storm with under 34 knots (43 km/h; 39 mph) 10-minute sustained winds
Classification used in North Indian Ocean basin			
sprcyclstorn . SUCS	<code>{{Storm colour sprcyclstorn}}</code>	<code>#ff0000</code>	a storm with at least 138 mph (222 km/h) 3-minute sustained winds
esvrcyclstorn . ESCS	<code>{{Storm colour esvrcyclstorn}}</code>	<code>#ffcc40</code>	a storm with 104–137 mph 3-minute sustained winds
vsrvcyclstorn . VSCS	<code>{{Storm colour vsrvcyclstorn}}</code>	<code>#ffffcc</code>	a storm with 74–103 mph 3-minute sustained winds
svrcyclstorn . SCS	<code>{{Storm colour svrcyclstorn}}</code>	<code>ccffff</code>	a storm with 55–73 mph 3-minute sustained winds
niocyclone . cyclstorn . CS	<code>{{Storm colour cyclstorn}}</code>	<code>00faf4</code>	a storm with 39–54 mph 3-minute sustained winds
deepdepression . DD	<code>{{Storm colour deepdepression}}</code>	<code>66baff</code>	a storm with under 35–38 mph 3-minute sustained winds
niodepression . D	<code>{{Storm colour niodepression}}</code>	<code>80ccff</code>	a storm with under 35 mph 3-minute sustained winds
nioland . landdepression . land . LD	<code>{{Storm colour land}}</code>	<code>80ccff</code>	a storm system that has formed over land
Classification used in South Pacific and Australian basins			
Fiji5 . Aus5 . A5	<code>{{Storm colour Fiji5}}</code>	<code>#ff0000</code>	a storm with at least 127 mph (205 km/h) 10-minute sustained winds
Fiji4 . Aus4 . A4	<code>{{Storm colour Fiji4}}</code>	<code>#ffcc40</code>	a storm with 99–126 mph 10-minute sustained winds
Fiji3 . Aus3 . A3	<code>{{Storm colour Fiji3}}</code>	<code>#ffffcc</code>	a storm with 75–98 mph 10-minute sustained winds
Fiji2 . Aus2 . A2	<code>{{Storm colour Fiji2}}</code>	<code>ccffff</code>	a storm with 55–74 mph 10-minute sustained winds
Fiji1 . Aus1 . A1	<code>{{Storm colour Fiji1}}</code>	<code>00faf4</code>	a storm with 39–54 mph 10-minute sustained winds
spdepression . TD low . TL	<code>{{Storm colour spdepression}}</code>	<code>66baff</code>	a storm with under 39 mph 10-minute sustained winds
disturbance . DI	<code>{{Storm colour disturbance}}</code>	<code>80ccff</code>	a poorly-organised storm with under 39 mph 10-minute sustained winds
Classification used in South West Indian Ocean basin			
vintense . VITC	<code>{{Storm colour vintense}}</code>	<code>#ff0000</code>	a storm with over 132 mph (215 km/h) 10-minute sustained winds (Very Intense Tropical Cyclone)
intense . ITC	<code>{{Storm colour intense}}</code>	<code>#ffcc40</code>	a storm with 104–132 mph 10-minute sustained winds (Intense Tropical Cyclone)
swioltc . TC	<code>{{Storm colour swioltc}}</code>	<code>#ffffcc</code>	a storm with 74–103 mph 10-minute sustained winds (Tropical Cyclone)
swiosts . STS	<code>{{Storm colour swiosts}}</code>	<code>ccffff</code>	a storm with 55–73 mph 10-minute sustained winds (Severe Tropical Storm)
mtstorm . sub . MTS	<code>{{Storm colour mtstorm}}</code>	<code>00faf4</code>	a storm with 39–54 mph 10-minute sustained winds (Moderate Tropical Storm)
swiodepression . swsubdep . TD	<code>{{Storm colour swiodepression}}</code>	<code>66baff</code>	a storm with under 39 mph 10-minute sustained winds
swioldisturbance . subdisturbance	<code>{{Storm colour swioldisturbance}}</code>	<code>80ccff</code>	a poorly-organised storm with under 39 mph 10-minute sustained winds
Z00W . DI			
Historical values (do not use)			
shem5	<code>{{Storm colour shem5}}</code>	<code>#ff0000</code>	
shem4	<code>{{Storm colour shem4}}</code>	<code>#ffcc40</code>	
shemsvertc	<code>{{Storm colour shemsvertc}}</code>	<code>#ffe775</code>	
shem3 . shemtc	<code>{{Storm colour shem3}}</code>	<code>#ffffcc</code>	
shem2	<code>{{Storm colour shem2}}</code>	<code>ccffff</code>	
shem1	<code>{{Storm colour shem1}}</code>	<code>00faf4</code>	
shenedepression . extratropical	<code>{{Storm colour shenedepression}}</code>	<code>66baff</code>	
	<code>{{Storm colour extratropical}}</code>	<code>cccccc</code>	

Figure 49: Storm Colour Template.

Reference

Colour Blind Awareness (n.d.) <https://www.colourblindawareness.org/colour-blindness/#:~:text=What%20is%20colour%20blindness%3F,most%20of%20whom%20are%20male>.

Wikipedia The Free Encyclopedia. Template: Storm Colour (2019) https://en.wikipedia.org/wiki/Template:Storm_colour

Figure 35: Image of colour picker. <https://imagecolorpicker.com/en>
Figure 36: Image of colour contrast ratio. <https://contrast-ratio.com>