WEEK 8 – WORK IN PROGRESS

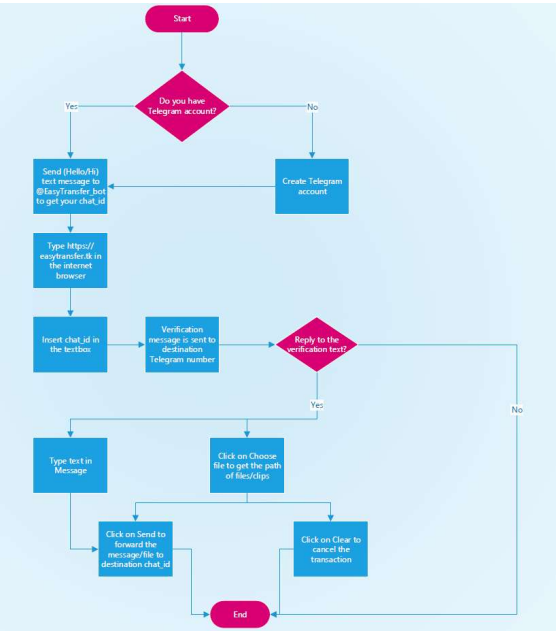
EASYTRANSFER WEBSITE

Work has been done:

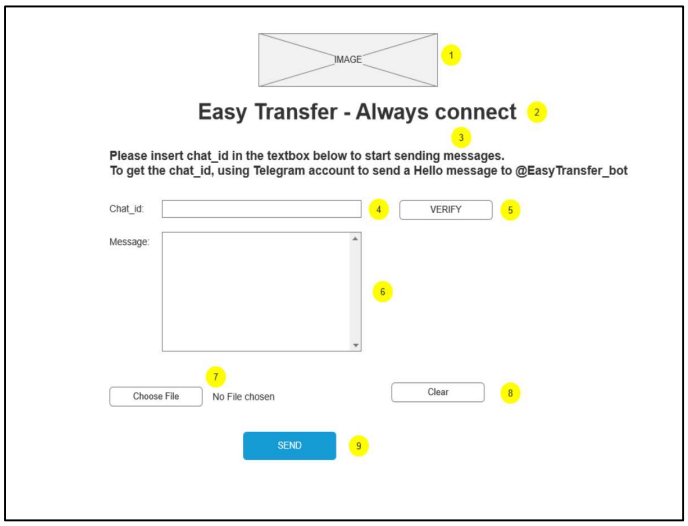
To ensure the website is responsive, functional and providing reliable and effective services, there is some work related to analysis, design and build completed. The outcomes of each task were submitted via email to supervisor (Dr Ben) on week 5 and week 7 and asked for feedback.

The benefits of planning stage and requirement management are to clearly define roles & responsibilities then streamline and document the development process. Moreover, the impact assessment of release candidates done at the planning stage will establish deliverables the rest of the application lifecycle.

- Requirement analysis
- User Interface specification
- Functional and technical specification
- Design walkthrough – Process flow

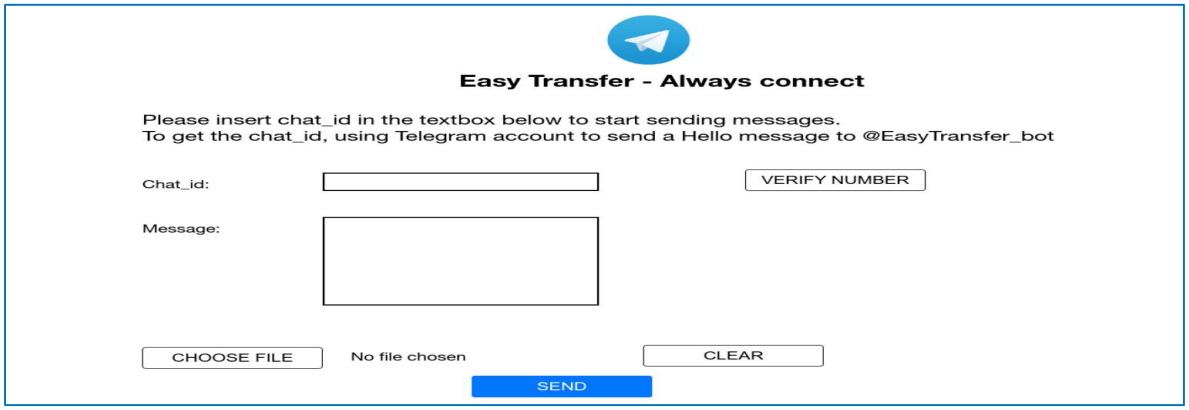


- Wireframe



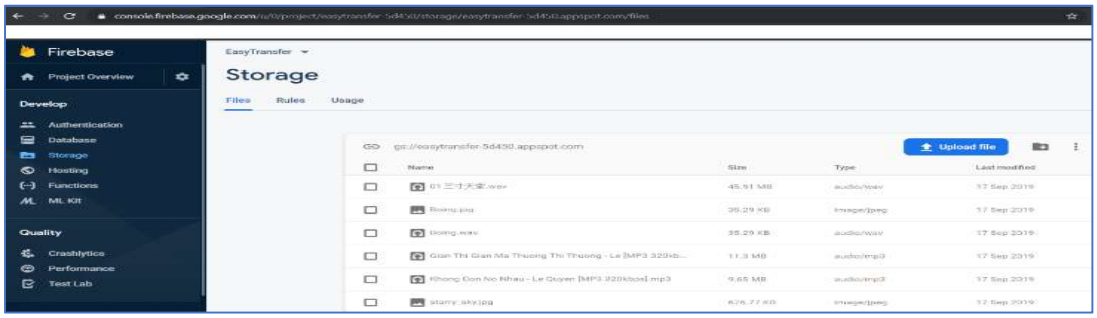
No	Element	Type	Description
1	Image		Logo of website.
2	Headline	Text	Title of website.
3	Content	Text	Brief summary of how to use website.
4	Chat_id	Number (9 digits)	Unique chat_id which is relevant to Telegram account of receiver. It must contain 9 digits to enable VERIFY button.
5	Verify	Button	Check if the chat_id is valid and exits by sending a text messages to Telegram account of receiver. Receiver needs to reply with Yes to confirm they wish to receive. After the successful verification, enable the SEND button.
6	Message field	Text	Enter the text message.
7	Choose File	Button	Open the file location and allow users to select the preferred file.
8	Clear	Button	Remove the file chosen previously.
9	Send	Button	Send Message or File to Telegram account of receiver.

- Prototype



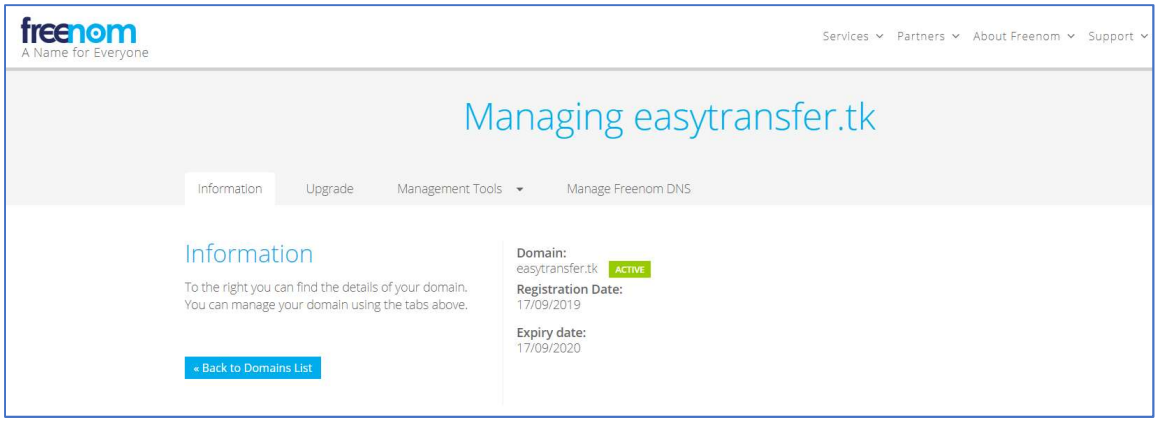
- Environment set up:

Firebase, one of Google products, has been utilized for temporary storage. All attached files will be uploaded to Firebase account. File will be automatically deleted after 2 days.

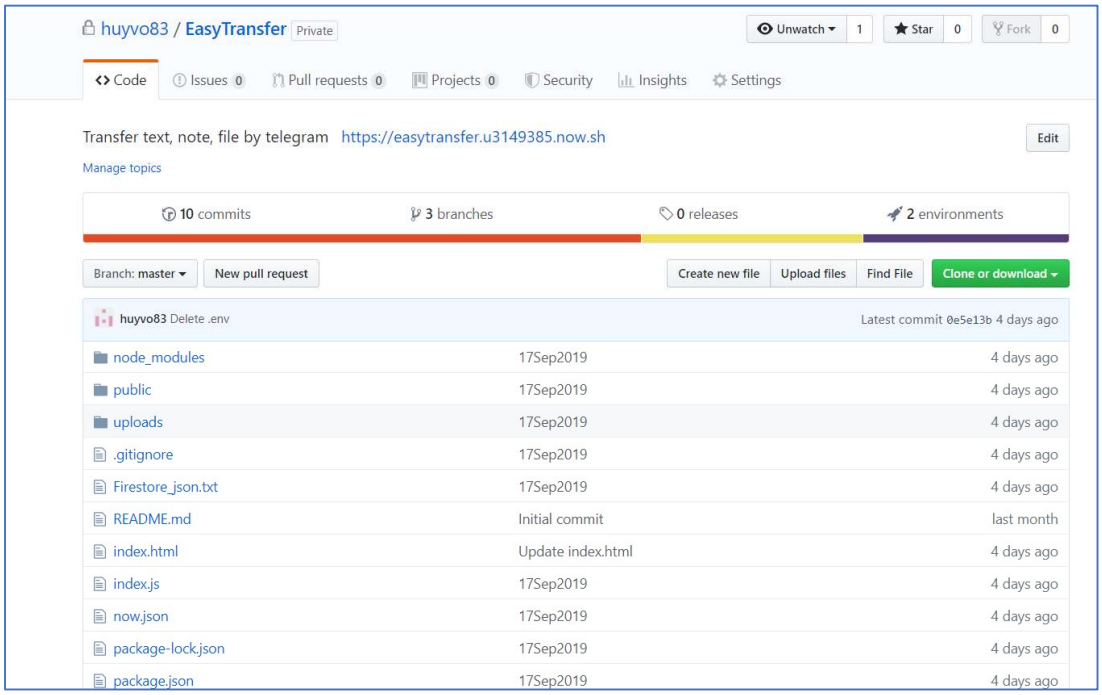


Host: Zeit.com

Domain: Dot.tk



Code versioning: GitHub (https://github.com/huyvo83/EasyTransfer)



Terminal to build and upload to Zeit.com: Node.js

- Function summary

1) Receiver uses Telegram account in their mobile phone to send “Hello/Hi” message to EasyTransfer_bot, webhook will trigger start_bot function in Zeit (index.js) and Zeit server will send Telegram message with chat_id back to receiver.

```
app.post('/start_bot', function(req, res) {
```

2) User inserts receiver’s chat_id on web browser (index.html) to request verification. Zeit server (index.js) will send message (Do you want to receive message from EasyTransfer?) to Telegram account of receiver to verify.

```
app.post('/verify', function(req, res){
```

3) Receiver answers “Yes”, the web browser will enable the “Submit” button.
(Before the verification is done, “Submit” button was disabled.)

```
$("#btnSubmit").prop("disabled", false);
```

4) User wants to send attachment to receiver: Zeit server will call function to prepare necessary parameters to initialize Firebase account to be ready for file upload.

```
firebase.initializeApp({
```

5) If user just sends text-only message to receiver, Zeit server will detect and use a different function.

```
function sendMessage(url, reply, chat_id, res) {
```

If user sends attachment or both (message and attachment), Zeit server will call the below function.

```
function sendDoc(url, reply, chat_id, file_url, res) {
```

Both above functions will communicate with the methods provided in Telegram API (https://core.telegram.org/bots/api)

```
let telegram_url_message = "https://api.telegram.org/bot" + process.env.TELEGRAM_API_TOKEN +"/sendMessage";
let telegram_url_doc = "https://api.telegram.org/bot" + process.env.TELEGRAM_API_TOKEN +"/sendDocument";
```

Work still needs to be completed:

The below tasks are pending to be done for the next few weeks and I am confident to progress them on time. Those are required as part of testing, documentation and maintenance. It helps traceability of change from requirements through to product solution, risk identification & mitigation and improved test outcomes through functional testing.

- Promotional poster
- Code review
- Test case/strategy
- User acceptance testing
- End-user training document

Change in production plan: My initial intention was to deploy a website with two ways of sharing. However, due to time constraint, the scope of the project is to release a website product transferring information from desktop to Telegram phone account.

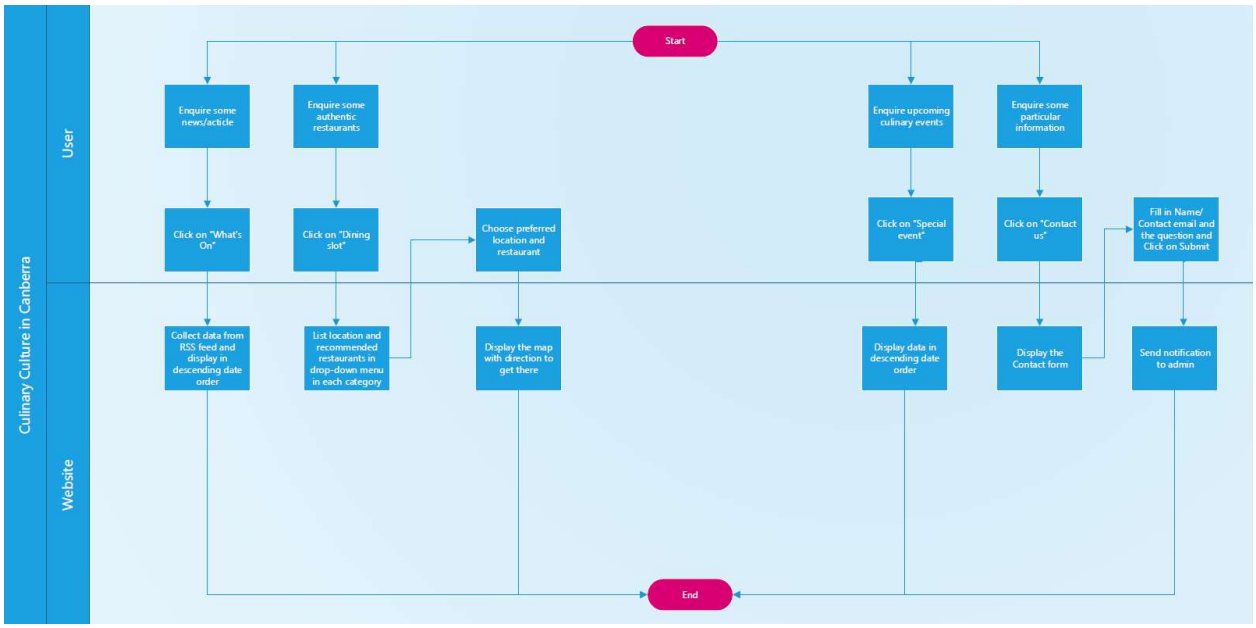
CULINARY CULTURE IN CANBERRA

Work has been done:

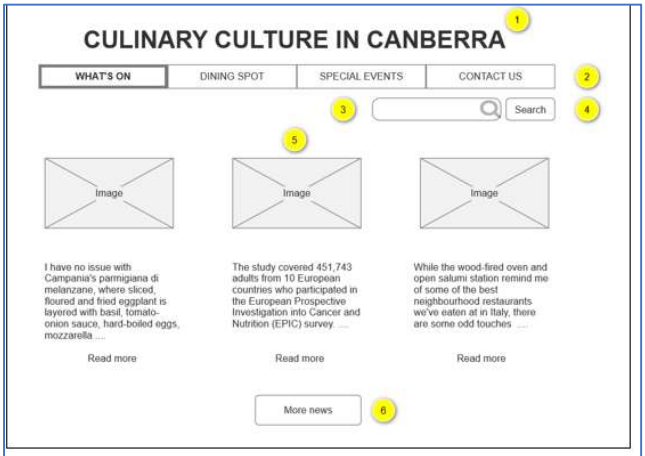
To ensure the website is responsive, functional and providing reliable and effective services, there is some work related to analysis, design and build completed. The outcomes of each task were submitted via email to supervisor (Dr Ben) on week 5 and week 7 and asked for feedback.

The benefits of planning stage and requirement management are to clearly define roles & responsibilities then streamline and document the development process. Moreover, the impact assessment of release candidates done at the planning stage will establish deliverables the rest of application lifecycle.

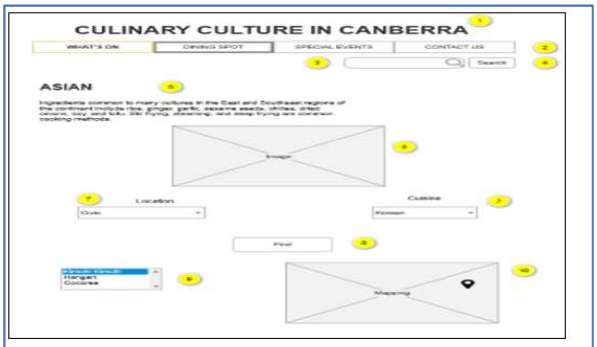
- Requirement analysis
- User Interface specification
- Functional and technical specification
- Design walkthrough – Process flow



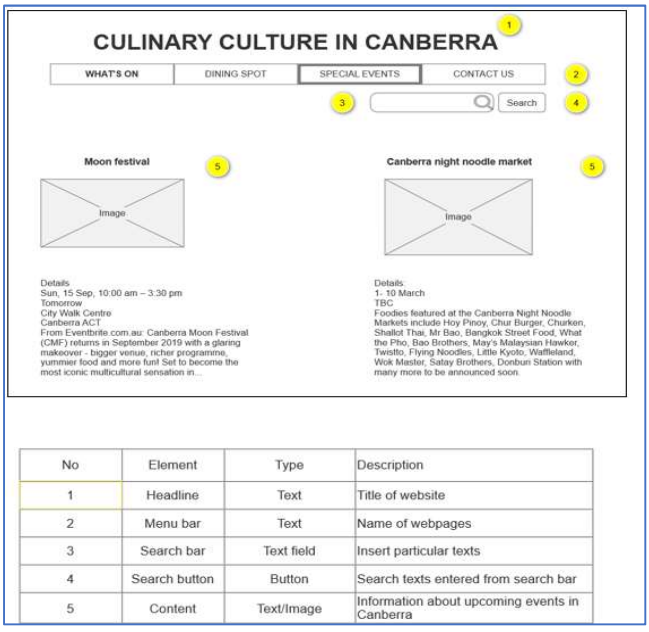
- Wireframe



No	Element	Type	Description
1	Headline	Text	Title of website
2	Menu bar	Text	Name of webpages
3	Search bar	Text field	Insert particular texts
4	Search button	Button	Search texts entered from search bar
5	Content	Image and Text field	Data retrieved from RSS feed/ API
6	More news	Button	Display more news/articles



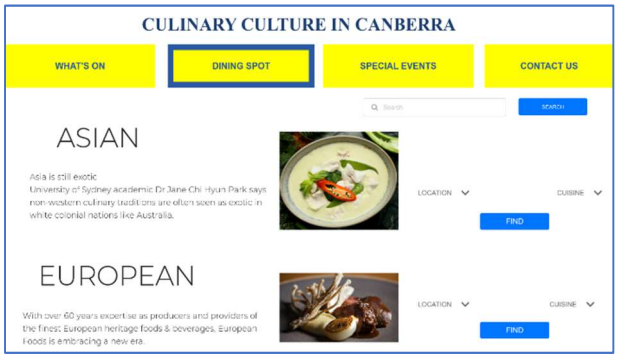
No	Element	Type	Description
1	Headline	Text	Title of website
2	Menu bar	Text	Name of webpages
3	Search bar	Text field	Insert particular texts
4	Search button	Button	Search texts entered from search bar
5	Content	Text field	Information about each culture (categorised into 4 cultures)
6	Image	Image	Image of some food
7	Location/ Cuisine	Drop-down list	4 main locations in Canberra and scoped 2 cuisines
8	Find	Button	Search the restaurants revelant to the location and cuisine
9	List	List box	Return the list of restaurants
10	Map	Image	Direction to go to restaurant

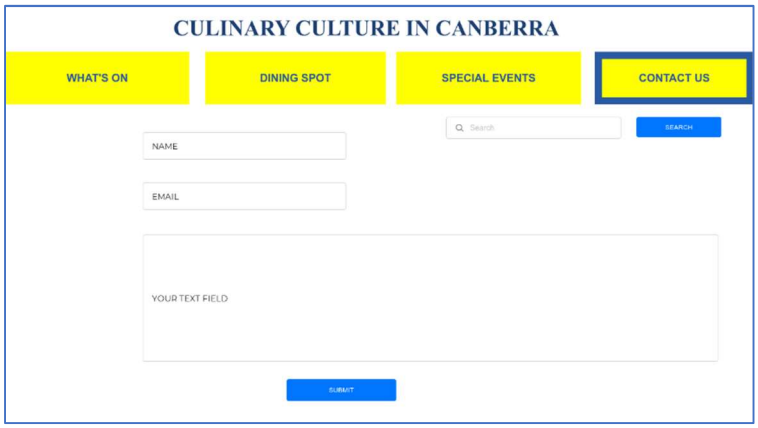


No	Element	Type	Description
1	Headline	Text	Title of website
2	Menu bar	Text	Name of webpages
3	Search bar	Text field	Insert particular texts
4	Search button	Button	Search texts entered from search bar
5	Content	Text/Image	Information about upcoming events in Canberra

No	Element	Type	Description
1	Headline	Text	Title of website
2	Menu bar	Text	Name of webpages
3	Search bar	Text field	Insert particular texts
4	Name	Text	Name of user
5	Email address	Text	Email address of user
6	Enquiry	Text field	Question/Feedback from user
7	Submit	Button	Send the form to admin's email

- Prototype

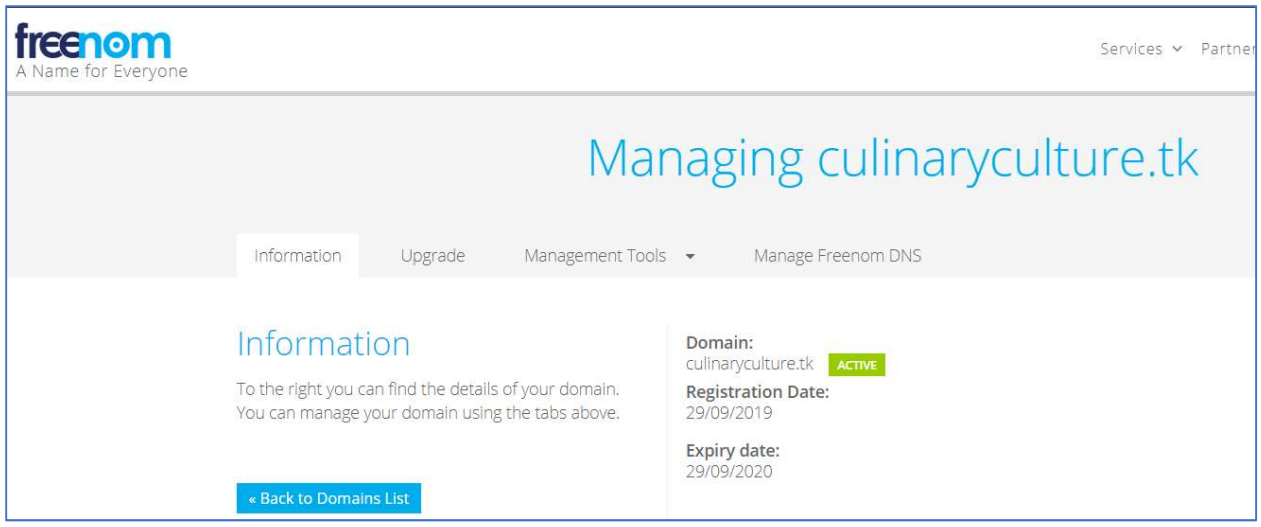




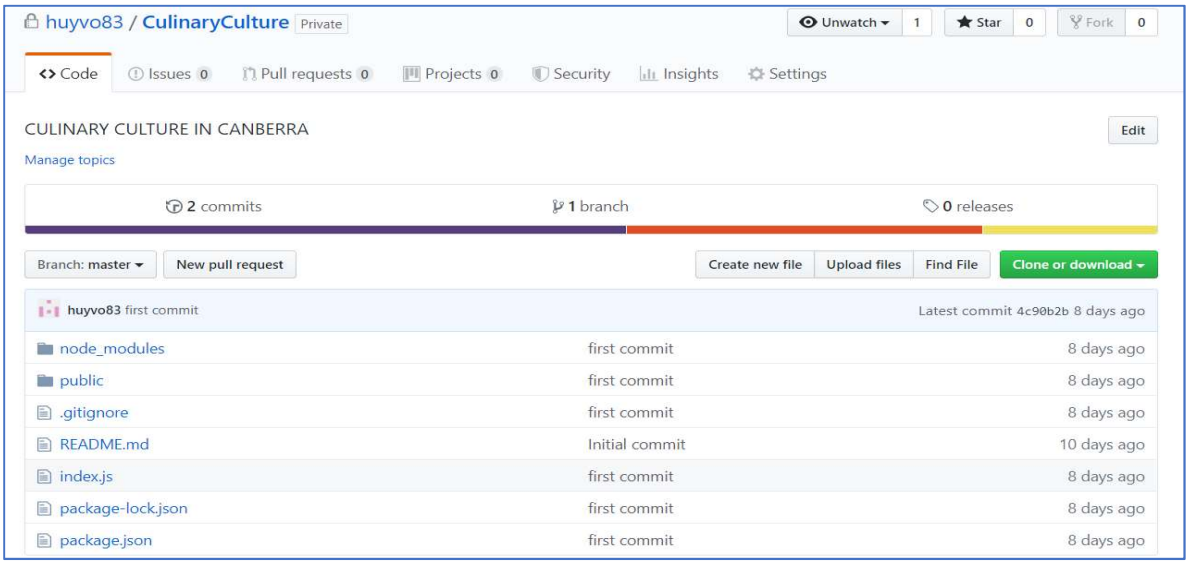
- Environment set up:

Host: Zeit.com

Domain: Dot.tk



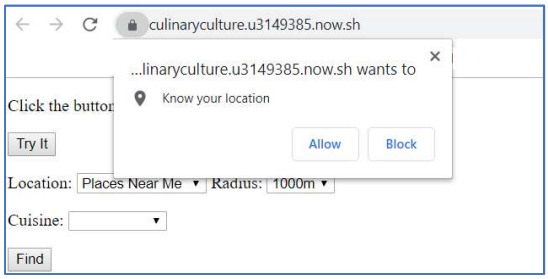
Code versioning: GitHub (https://github.com/huyvo83/CulinaryCulture)



Terminal to build and upload to Zeit.com: Node.js

- Completed functions:

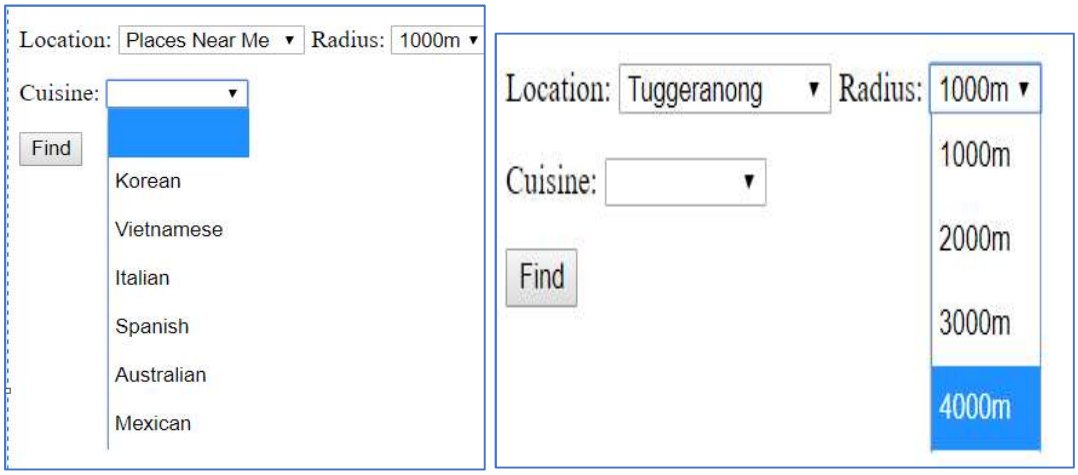
1) Detect current location of user and return the restaurant with location “Places Near Me”.



2) Use Places API (in index.js) to find the relevant restaurants with pre-selected location, radius and cuisine.

```
let near_by_call = '
https://maps.googleapis.com/maps/api/place/nearbysearch/json?type=restaurant&key=' + process.
env.GOOGLE_API_TOKEN ;
```

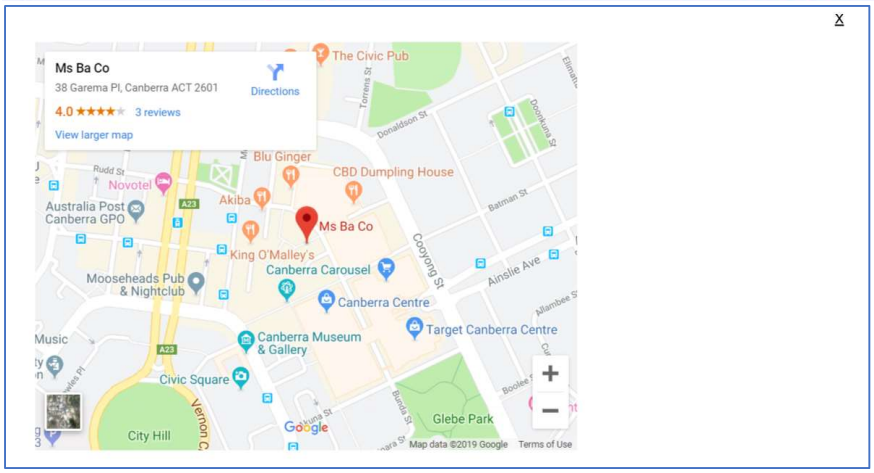
```
findLocation(userInput, near_by_call,res);
```



3) Return the direction map when user clicks on their preferred restaurant.

```
function showEmbedMap(place_id){
    var Iframe_Open  = '<iframe width="600" height="450" frameborder="0" style="border:0" src="
    https://www.google.com/maps/embed/v1/place?key=AIzaSyBfEZy8-4n1bfWcL9haLlAjTioWvLZW_M4&q=place_id:';
    var Iframe_Close = '"></iframe>';
    $(".popup-inner-body").append(Iframe_Open + place_id + Iframe_Close);
```

```
.click(function(){
    showPopup();
    showEmbedMap(findResult.results[i].place_id);
```



Work still needs to be completed:

The below tasks are pending to be done for the next few weeks.

- Functions:
    - What's on webpage (use News API)
    - Dining spot: 3D images
    - Special event webpage
    - Contact us webpage
    - CSS and HTML for layout
- Promotional poster
- Code review
- Test case/strategy
- User acceptance testing
- End-user training document

Change in production plan: At the first stage of wireframe and prototype design, I thought it would be good to have each category of culture seperately. However, after considering the convenience and usefulness in real life for user (where most of the cases, they prefer to use smartphone to open the website) and timeframe, I decided to combine all categories of culture under one cuisine drop-down box and more specifically, there will be 6 options – Korean, Vietnamese, Italian, Spanish, Australia and Mexican. This function can be a perfect candidate for future enhancement when more time and resources are allocated.