

## System Requirement Specifications (SRS)

# 1 Table of Contents

<b>1 Table of Contents</b>	<b>1</b>
2 Problem Statement	3
3 Overview	3
3.1 Background	3
3.2 Overall Description	3
4 Investigation & Analysis Methodology	4
4.1 System Investigation	4
4.2 Analysis Methodology	4
4.2.1 Feasibility study and requirements elicitation	4
4.2.2 System analysis and requirements specification	5
4.2.2.1 Analysis of the problem using object-oriented techniques	5
4.2.2.2 Scope and Limitations	5
4.2.2.3 Object-oriented design using UML	6
4.2.2.4 Use Case Model	6
4.2.2.5 Prototyping	6
5 Constraints	7
5.1 Scalability	7
5.2 Data and Function Mapping	7
5.3 Proprietary hardware and software	7
5.4 Batch updates vs. (close) Real-time updates	7
5.5 Project Schedule	7
6 Operational Requirements	8
6.1 Help Desk Support	8
6.2 Application Services and Technical support	8
6.3 Administration Features	8
6.4 Live update of database	8
6.5 System hardware fail over and routine back-up	8
7 Functional Requirements	9
8 Input Requirements	10
8.1 Location Category	10
8.2 Search Keyword	10
8.3 Plan File	10
9 Process Requirements	11
9.1 Database Transaction	11
9.2 Data integrity	11
9.3 Data validation	11
9.4 Performance	11
9.5 Data repository	11
10 Output Requirements	12
10.1 Interactive Map	12

	System Requirement Specifications	
10.2 Location Details		12
10.3 Plan Viewer/Export		12
11 Hardware Requirements		12
11.1 Network		12
11.2 Client Computers		13
11.3 Production support systems		13
12 Software Requirements		13
12.1 Client Operating Systems		13
12.2 Client Application		14
12.3 Network system		14
12.4 Mainframe system		14
12.5 Licenses		14
13 Deployment Requirements		15

## 2 Problem Statement

Many young adults in Singapore meetup after work or study to socialize. To meet up for a meal or visit travel destinations to socialize, many people seek the internet for help, often in the form of websites and mobile apps. Hence, the development of such softwares has become a large industry today. On the internet, there are many planner websites such as Wanderlog and Inspirock that perform planning of travel itineraries. These websites allow their users to select multiple destinations to visit and helps them create a plan to maximise the time they have on holiday or on leave. On the other hand, there are also numerous mobile applications and websites that help users decide the best locations to go to enjoy a meal, or hang out when they are free. Search applications such as Yelp, Tripadvisor and Foursquare often make use of search algorithms that find food and travel destinations based on user search preferences like their current location, dietary preferences, costs and user reviews. However, there are no applications or services found on the internet that provide the search functionality of search applications, while offering the itinerary planning capabilities of planner websites. As a result, young adults that are busy with work or study have to consult both search applications and planning websites to help them plan a meet up for their friends or family. This is an inefficient process that affects the social lives of young adults in Singapore as well as around the world.

## 3 Overview

### 3.1 Background

Socializing is an important part of a young adult's life. A social atmosphere provides a person with a conducive environment to relax from a hectic work-life and even helps form relationships with other people[1]. Whether with friends or family, it is very important for young men and women to be able to meet up and participate in social events to encourage social development and relieve stress. For some, social events even provide emotional and mental support when in a difficult point in their lives. In these times, people often turn to their friends or family, and often partake in social events to restore their mental wellbeing. In a study conducted by the National Youth Council of Singapore, 85% of Singaporean youths would prefer simply having dinner with family for such support [2].

### 3.2 Overall Description

The “Go Where GaiGai” application is a website designed to make searching and planning for food and leisure experiences quick and simple. The website allows users to search for food and entertainment destinations, by accepting search preferences such as location and category. The website's algorithm will use these user inputs to dynamically search for said destinations and display them on a map. The data of these locations are pulled from Yelps public API.

The website will then allow the user to add selected destinations into a plan. The website will allow users to import and export their plans in order to save them as files of plain text format.

## 4 Investigation & Analysis Methodology

### 4.1 System Investigation

“Go Where GaiGai” is a standalone website designed with the aim to allow users to search for food and entertainment destinations easily. The target users of the website are people looking to plan food and leisure experiences within Singapore. Such people include young adults and adults busy with work or study, and tourists in Singapore without a fully planned itinerary. When the user opens the website, they will see a blank map of Singapore. There will be a navigation bar on the top of the website where a user will be able to select their category of location. They can choose to specify categories such as dining locations, gardens, bars etc. There will also be a search bar for users to type keywords that might help their search such as names of towns in Singapore to narrow down the search of the website.

Once the user begins the search, the website's algorithm will search for related locations and display five locations that best fit the user's search preferences. These locations will be displayed as map pins on the map of Singapore. Each pin on the map can be clicked to expand a panel displaying basic information of each location. The basic information displayed includes user reviews, the name and address of each location. The information from which the location and basic information of each location is acquired from the website's back-end database. The data from the back-end database is pulled from Yelp's public API.

The user can also select locations and add them to the planner on the website. The website will store each plan as a plain text file. The user can save their created plans by exporting the plain text files. The files can then be reused by importing it back into the website for viewing purposes.

### 4.2 Analysis Methodology

#### 4.2.1 Feasibility study and requirements elicitation

In order to ensure the practicality and feasibility of “Go Where GaiGai”, a detailed survey and interview will be conducted on 56 people. These 56 people were made up of university students and employed adults. There will be a similar number of the two groups of surveyees when conducting this interview to maintain the consistency of results. The survey will contain questions designed to find out how important meeting up for social events are to busy people, and to find out what are the common difficulties faced when trying to plan for certain events. The information from this survey will serve as a point of reference from which the requirements of the website can be elicited. The development team will be able to design the software in such a way that seeks to target the pain points identified from this survey.

## 4.2.2 System analysis and requirements specification

### 4.2.2.1 Analysis of the problem using object-oriented techniques

An external view of the model of the website will be developed using Unified Modeling Language (UML). The project will adopt a class-based language to perform object-oriented techniques for design. Classes include the different pages of the website, the map drop points when they are on the map, as well as other functions on the website.

Some desired features of the system include:

- The ability to view a map of Singapore
- The ability to query locations in the back-end database based on location and category.
- The ability to perform a dynamic search based on the user's search preferences.
- The ability to add location information onto a plan.
- The ability to import and export a plan in plain text file format.

### 4.2.2.2 Scope and Limitations

Scope:

1. The website is primarily designed to be displayed on a computer screen.
2. The user can input a location in the search bar, or select a category from a drop down field.
3. The website can perform a search using the user's input, and display the locations on the map.
4. The website's map will utilize location information from Yelp's public API. The map will receive this information from Yelp every four hours automatically.
5. The user can add location into a planner.
6. The planner can import and export the planner in a plain text file format.

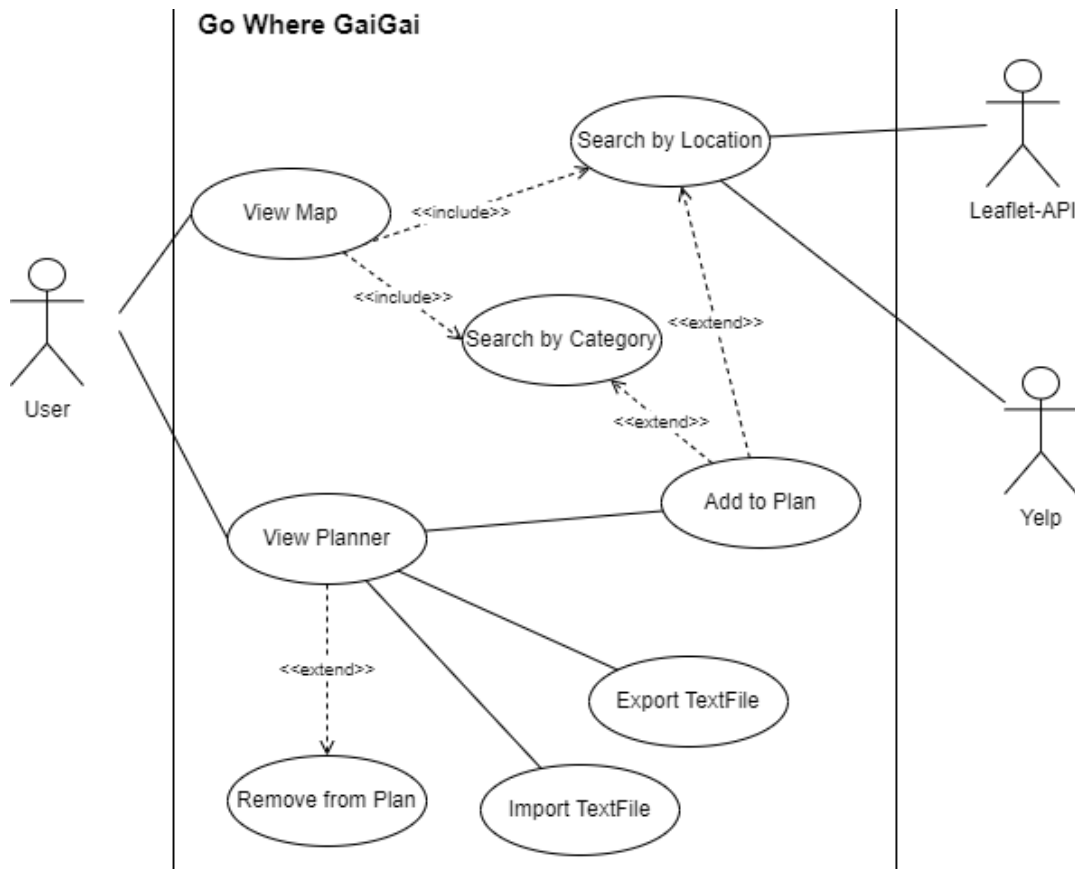
Limitations:

1. If the website fails to establish a connection to Yelp's public API, the information reflected on the map will be outdated. This might cause user's to be viewing incorrect information.
2. The system will be prone to performance issues when the batch update of data from Yelp occurs. This is due to a large inflow of information each time the system polls the API for data. Users might experience this performance issue when searching for locations at this time.
3. The website might be prone to crashes in the case where multiple users are concurrently using the application. Load testing will help to overcome this issue.

### 4.2.2.3 Object-oriented design using UML

A detailed classed based approach to object-oriented design for the website will be developed. UML will be used again for the graphical representation and documentation of the design. It will primarily be focused on the front-end website, and how the different pages and components interact with each other. It will generally describe how the front-end graphical user interface interacts and uses data from the back-end.

### 4.2.2.4 Use Case Model



### 4.2.2.5 Prototyping

The prototype of "Go Where GaiGai" will be implemented with a class-based approach. The prototype will be a working example of the major features of the whole website. It will include a functioning map, as well as the use of Yelp's public API to be able to pass location data onto the map as map pins. The prototype will ensure the front-end user interface is able to interact seamlessly with the back-end database. After approval from the Project Manager and QA Manager, the prototype will be used to gather early feedback from beta-testers to ensure that the project is headed in the right direction.

## 5 Constraints

### 5.1 Scalability

The website scales well to increasing system demands. System demands come in the form of location data pulled from Yelp's public API. The website's search algorithm was designed to only display a certain number of map pins on the map of Singapore. This prevents excess elements displaying on the website. The search algorithm is also designed to search dynamically based on user preferences, to prevent brute force searching on the back-end database. Error handling is also limited to few anticipated or common errors.

### 5.2 Data and Function Mapping

New functions can easily be added to the main website. The BrowserRouter library from Reactjs allows for the simple mapping of features onto a new page by defining the new page URLs in the source code of the main page. Features can then be added into these new pages.

However, changing the source of the location data requires an API which can produce the same parameters such as longitude, latitude, user reviews, user review count, etc.

### 5.3 Proprietary hardware and software

"Go Where GaiGai" website only requires a working browser as specified under Software Requirements. No proprietary hardware or software is required to function properly.

### 5.4 Batch updates vs. (close) Real-time updates

Updates of the map information between the back-end database and front-end will happen at intervals in batch updates. This is because there is a large amount of information polled from Yelp's public API each pull.

### 5.5 Project Schedule

The complete project execution will span three months, with implementation of the prototype and major features taking up two months. The project will begin in the fourth week of August 2022, and the deployment and final release of the product will happen at the end of October 2022. The full Gantt Chart can be found in the Project Proposal.



## 6 Operational Requirements

### 6.1 Help Desk Support

System users will have 24x7 access to telephone assistance for questions that are technical in nature, such as slow or sluggish system response time, incompatible browser features, application errors, system downtime inquiries, etc. Contact information of developers and engineers will be available on the “Contact Us” section of the team wiki. Bugs can also be reported via email found in said section.

### 6.2 Application Services and Technical support

Programmers and application developers will have access to source code to address bugs or system enhancements as deemed necessary. Network Administrator and DBA support is also required to maintain a 24x7 system uptime.

### 6.3 Administration Features

Since users are not required to create accounts to access features of “Go Where GaiGai”, there are no System security and access levels in the system.

However, only authorized system administrator(s) and database engineers will be able to access the back-end database stored in the system.

### 6.4 Live update of database

The website is able to run on local copies of information data in an excel file format. However, the website must be able to continuously and automatically poll Yelp’s public API for location information. This is required as information of locations such as user reviews are constantly changing, hence affecting the search algorithm performed by “Go Where GaiGai”. This entails that the public API must be live and working in order for the website to function smoothly.

### 6.5 System hardware fail over and routine back-up

Computer operations center will handle system hardware tasks such as database back-up and rollback, hardware maintenance, fail over, scheduled system patches and maintenance. Appropriate protocols for when to perform rollbacks and fail overs will be produced.

## 7 Functional Requirements

### 7.1 Search attractions

7.1.1 Users must be able to search by categories.

7.1.1.1 User selects the category from the dropdown menu that includes but not limited to Bars, Museums, Hawker Centre, etc.

7.1.1.2 The system will then query the database for results and return the top 5 attraction based on the category selected.

7.1.1.3 The system will then highlight the drop points of the returned result.

7.1.2 Users must be able to search by location.

7.1.2.1 Users must be able to input the location or the postal code .

7.1.2.1.1 System must check if the input is valid.

7.1.2.1.2 If the input is invalid, an error message must be displayed.

7.1.2.2 The system will then query the database for results and return the top 5 attraction around the location.

7.1.2.3 The system will then highlight the drop points of the returned result.

7.1.3 Users must be able to search by categories and locations concurrently.

7.1.3.1 System must be able to check if both input has been entered.

7.1.3.1.1 If both inputs have been entered, the system must validate both the inputs.

7.1.3.1.1.1 If either of the inputs is invalid, an error message must be displayed.

7.1.3.1.2 If both input has been validated, system will then query the database for results and return the top 5 attraction based on the category selected as well as attractions near the location.

7.1.3.1.3 The system will then highlight the drop points of the returned result.

7.1.3.2 If only either input has been entered, the system will follow either 7.1.1 or 7.1.2.

### 7.2 Interactive Map

7.2.1 Users must be able to zoom in and out of the map.

7.2.2 Users must be able to reset the map.

7.2.3 Users must be able to click on the drop points.

7.2.3.1 Drop points must show the picture of attraction.

7.2.3.2 Drop points must show the name of the attraction.

7.2.3.3 Drop points must show the ratings of the attraction.

7.2.3.4 Drop points must show the categories of the attraction.

7.2.3.5 Drop points must show the address of the attraction.

7.2.3.6 Drop points must show the reviews of the attraction.

7.2.3.7 Drop points must allow the user to add the attraction to the plan.

## 7.3 Planner

7.3.1 Users must be able to edit the planner.

7.3.1.1 Users must be able to edit any of the attractions in the planner.

7.3.1.2 Users must be able to delete any of the attractions in the planner.

7.3.1.3 Users must be able to reorder any of the attractions in the planner.

7.3.2 Users must be able to import a planner.

7.3.2.1 System must validate the text file .

7.3.2.2 If valid, a successful message must be displayed.

7.3.2.3 If invalid, an error message must be displayed.

7.3.2.2 Details of the imported planner must be displayed on the planner page.

7.3.3 Users must be able to export a planner.

7.3.3.1 If export is successful, a successful message must be displayed.

7.3.3.2 If export is not successful, an error message must be displayed, informing user why it is not successful.

# 8 Input Requirements

## 8.1 Location Category

The first method a user can use to input their search preferences. Locations fall under different categories such as gardens, bars, zoos, and cuisines, i.e., singaporean, taiwanese and italian. A drop-down list on the homepage of the website will display these options for the user to select before they click and begin the website's search algorithm.

## 8.2 Search Keyword

The user can also use search keywords to influence the search performed by “Go Where GaiGai”. This method however requires the user to type a string input into the search bar on the top of the homepage. The website will search the back-end database for any fields which contain the matching string input entered by the user. Therefore performing similarly the Location Category input, but allowing for greater flexibility in narrowing down search results.

## 8.3 Plan File

One of the major features of “Go Where GaiGai” involves the ability to plan the destinations of an outing and exporting it as a plain text plan file. This user input refers to the importing of the same plan file onto any client running the same website, to allow users to view the plan. The user will be asked to

upload the file and click an “import” button, following which the website will read the plain text file and display it on the website for viewing.

## 9 Process Requirements

The following are among the inherent requirements that “Go Where GaiGai” must be able to handle.

### 9.1 Database Transaction

Adhering to the ACID properties of a database transaction, the system must be able to send, receive and trigger transactions to the database system.

### 9.2 Data integrity

Data received from the API should not be tampered with as this program strives to give the most honest and latest review for an attraction. The database will ensure that the data received from the API are in the correct format as well as the correct data type.

Audit trails and logs are also set-up in case there is a need to trace and trace modification of data in the database.

### 9.3 Data validation

The programs provide data validation and useful error-handling messages. Should the user enter a wrong format for the location or upload a file with an incorrect format, the program must be able to handle the error and tell the user what the error is..

### 9.4 Performance

The program should provide concurrent usage to at least 100 users on a 24x7 basis. It should also be able to automatically poll Yelp’s Public API every 4 hours in order to get the latest information. Search results have to be returned within 1 second from searching.

### 9.5 Data repository

The system should store all the relevant information the user searches on. This will allow the systems to better recommend attractions for the user

## 10 Output Requirements

### 10.1 Interactive Map

The homepage of the website will contain a map of Singapore. This interactive map is where the user can find their current location, and where search results of the website's algorithm will be displayed as map pins. The user can drag to move the map and zoom in and out.

### 10.2 Location Details

The user can click on the map pins generated after the website completes its search. A panel on the side of the website will then display a picture and basic information of the location the user clicked on. Basic information includes name, address, user reviews and user review count of the location. From here the user can choose to add the location to a meetup plan.

### 10.3 Plan Viewer/Export

After creating a plan, the website is able to convert the plan into a plain text file for the user to save and export. This function will also serve as a viewer, allowing the user to view their plan on the website.

## 11 Hardware Requirements

### 11.1 Network

A stable and high-speed wireless or wired connection is required for the website to function. The recommended speed to ensure smooth functioning is 5-10 Mbps (megabits per second) upstream bandwidth.

## 11.2 Client Computers

The client computer hardware specification is as follows:

Minimum:

Processor: Single Core 1 GHZ or equivalent

Memory: 2 GB RAM

Recommended:

Processor: Dual Core 2 GHZ or equivalent

Memory: 2 GB RAM

## 11.3 Production support systems

As the back-end of “Go Where GaiGai” is pulled from and managed by Yelp’s public API, no infrastructure management is required. The project will however have a database storing copies of the data pulled from Yelp’s public API which can be accessed by the Database Engineer when necessary.

# 12 Software Requirements

## 12.1 Client Operating Systems

- UNIX (any flavor)
- MAC
- Windows

## 12.2 Client Application

The Client will be able to work on any Java and Javascript compatible browser. Suitable browsers as of September 2022 include:

- Google Chrome versions: 63- 104, 105, 106- 108
- Microsoft Edge versions: 79- 104, 105
- Safari versions: 11.1- 15.5, 15.6, 16.0- TP
- Mozilla Firefox versions: 67- 130, 104, 105- 106
- Opera versions: 50- 89, 90

## 12.3 Network system

Network software and protocols in order for systems to communicate:

- TCP/IP
- HTTP
- HTTPS
- FTP

## 12.4 Mainframe system

- IBM Gateway
- DB2 database

## 12.5 Licenses

Valid licenses are required to run software from third party vendors:

Application Development Tools	Node.js	MIT License Agreement
	React	MIT License Agreement
	Github	MIT License Agreement
	npm	Artistic License 2.0
	phpMyAdmin	GNU General Public License 2
Back-end and Database	Python	PSF License Agreement

## 13 Deployment Requirements

