NPS2001C
Milestone 1

Chong Zhao Yang, Shun Pyae Phyo,
Ho Jia Wen, Jane, Han Xin Ping

1. Which real world problem have you chosen to work on? Why is it an important problem?
We have chosen to work on the problem of NUS students and staff not knowing the wait times for food, particularly during lunch hours. A study conducted by SMU has found that customer satisfaction decreased from 71.5 to 50.4 points after waiting for more than 20 minutes (SMU, 2017). Moreover, according to the Ministry of Manpower, the minimum lunch break that is provided to employees is at least 45 minutes (Ministry of Manpower, n.d.). Taking all these into account, the problem of not knowing how long the queue is would result in customers being unsatisfied with the service. Students and staff would also attend lessons and return back to work late respectively should they underestimate the wait times for the food. Therefore, this makes the problem of not knowing the wait times for food important.

2. What will your app do and how will it help solve or mitigate the problem?
The main function of our app would be to allow customers to order their food, view their queue number, the number of people before them in the queue, and the expected wait time for their food. A prerequisite of this would be to integrate the queues and menus of all the food stores in each canteen within NUS onto a single digital platform. The app would mitigate the problem stated above by allowing customers to view the expected wait time for food from the respective stalls. This then allows customers to make informed decisions regarding which stall they want to order from to minimize the wait time and allow customers to return back to work or attend lessons on time. Therefore, this decreases the lack of information with respect to the wait times and improves the outcome of returning back to work or attending lessons on time.

3. What is the central algorithm or class of algorithms that will enable this app to work?
The first central algorithm that will enable this app to work is that of menu management. This algorithm has three segments - display and updating of food items, sorting and filtering of food items and searching of food items.

The first algorithm within this central algorithm is that of display and updating of food items. The algorithm will take in the picture, name, and price of the food item and store them as elements in a dictionary. The algorithm will then display the picture, name and price of the food item. The algorithm would then initialize the availability of the food items as available. When an input is received by the store owner that the food item is no longer available, the algorithm would change to indicate that the food item is no longer available and that customers would not be able to order the food item.

The second algorithm will sort and filter food items depending on variables, particularly the expected wait time for the food. This allows customers to choose food items that are within their tolerable wait times. The wait times of each store will be stored in a list. When filtering for the wait time, the algorithm will look at the wait times of each store within the list. If the wait time is less than that input by the customer (E.g. the wait time is 10 minutes while the filter applied is less than or equal to 20 minutes), the algorithm will display the names of the stores which have a wait time of less than or equal to 20 minutes. Should the wait times be unsorted, the algorithm will also include a sorting algorithm in the form of quicksort.

The third algorithm will search food items based on the keywords input by the customer. The input by the customer (further which is defined as keyword) would be stored within a variable. The name and wait time of food items in each store will have already been stored as key and value respectively within a

dictionary. The algorithm would then look to see if the keyword is present in any of the keys within the dictionary.. If the keyword is in key, the algorithm would obtain the corresponding wait time and display both the name and wait time of the food item to the customer.

The next central algorithm that will enable this app to work is that of queue management. Each store will have a list that contains the customers waiting in queue. When a customer makes an order, they will be appended to the list. Should the list be unsorted, a sorting algorithm in the form of quicksort would be implemented to sort the list to sort based on order timing. A searching algorithm in the form of linear search would then be implemented to determine the customer's position within the queue (position of element within the list of active orders). The position of element will then be returned as a queue number for the customer to refer to.

In general, the wait times for customers will be given by 10 * Number of food items ahead of them in the queue.

Below are the resources that we have referenced in understanding our algorithm.

- https://www.w3schools.com/python/default.asp
- https://www.geeksforgeeks.org/
- https://realpython.com/sort-python-dictionary/#using-the-sorted-function

Basic coding ability is required to understand documentation that is involved in the process of algorithm formulation. There is no prerequisite in understanding algorithms that currently exist.

4. Under what circumstances do you expect users to use your app?

Our target audience is NUS students and staff members. Users will likely use the app while on the move, deciding where to go for meals based on real-time queue information, which is essential to beat the lunch crowd. We anticipate four potential limitations with this idea. Firstly, loading issues due to high user traffic during peak times might lead to slow loading or potential server issues. Investing in robust server infrastructure and optimizing the app for performance can help to curb loading problems. Secondly, providing live updates on queue lengths might pose a challenge due to the need for constant data refresh. We can circumvent this by implementing efficient data fetching mechanisms and consider a balance between real-time updates and system load. Thirdly, each order can have several items so if the estimated time is calculated by multiplying the number of orders with a chosen timing, instead of the number of items, the estimated wait time will be inaccurate. We can mitigate this by including number of items under the order number element and multiplying the number of food items by the waiting time instead . Lastly, some stalls may only input the order after food preparation, such as in the case of economic rice. This results in the estimated wait time not taking into account the people who are queuing to order. A way to mitigate this is by mandating individuals to make orders from food kiosks and then integrating the data from food kiosks and online orders and displaying the queue number once the food is ready for collection. This would then reduce the problem of lack of queue management across different digital platforms.

References

SMU (2017). Excessive waiting time for food causes F&B sector to lose return customers. *SMU Newsroom*. https://news.smu.edu.sg/news/2017/11/30/smu-study-excessive-waiting-time-food-causes-fb-sector-lose-return-customers

Ministry of Manpower (n.d.). Hours of work, overtime and rest day. *Ministry of Manpower Employment Practices*. https://www.mom.gov.sg/employment-practices/hours-of-work-overtime-and-rest-days