Sprint 2 Planning

Intelligent Search

Sidhant Chitkara, Alexander Meyer, Apoorva Parmar,
Yash Pujara, James Shao, Michael Vieck

SPRINT OVERVIEW

The goal for this sprint to to create a “production ready” product. We will be expanding the search functionality for Dining courts and starting work on an early working prototype for citybus live data. Additionally, a time will be spent improving the CSS / functionality of the website. User accounts and preferences, user stories carried over from Sprint 1, will also be completed during this sprint.

There are a few challenges for this sprint. The dining side of the website almost has all of it’s functionality completed and the remaining few to be added won’t be too difficult. The majority of the challenged will be on the citybus side. We will have to create a web-service hosted by CityBus to get the live bus data and translate it to our database. Furthermore, we will have to be able to accommodate multiple requests which may involve threading our backend requests. The front-end will have the technical challenge of mapping the busses, along with predicted movement while we are waiting on updated live data to come in. This sprint will require a lot of math and calculations on the back-end’s part while the front-end will have to figure out a good way to map the bus routes and provide an intuitive UI for the users.

Scrums will be held during the original class times - Tuesday and Thursday starting at 3:00pm. The meetings will ideally be 10-15 minutes and led by James. Team coding sessions will also be held once a week, most likely on a Friday with the exact time to be determined.

CURRENT SPRINT DETAILS

*Note: Identical tasks are listed again if they are relevant to multiple user stories.

Task Team	Team Members	Total Hours
Front End	Apoorva, Michael	20.5
Tokenizer	Yash, Sidhant	1
Services	Alex	16
Data Processing	James, Alex	9
Database	James	4

Server	James	2
*Note: Tasks are assigned to teams and members can choose what task they are working on. **Note: Team Members are free to choose their Task Team and can choose to change teams.		

1. **Users would like to search for a specific item in a specific dining court *for any date* then return the item card. *From Sprint 1***

Team	Hours	Description
Services	2	Create functions that can convert the returned API a to JSON format and pass to the front end.

Acceptance Criteria: Successful search of *"Does [DINING COURT] have [ITEM NAME] on [DATE]"* or *"Is [ITEM NAME] at [DINING COURT] [DATE]?"*. The query will return whether this item is being served at the dining court.

2. **Users would like to search for a specific item across all dining courts *for any date* then return what dining courts have that item. *From Sprint 1***

Team	Hours	Description
Services	2	Create functions that can convert the returned API data to JSON format and pass to the front end.

Acceptance Criteria: Successful search of *"Where can I find [ITEM NAME] [DATE]"* or *"Is [ITEM NAME] being served [DATE]?"*. This query will return a list of Dining courts that the food is being served at today. This query will only work for TODAY this sprint.

3. **Users should be able to create an account then log in with the credentials used and be able to parse the user information and start a user session. *From Sprint 1***

Team	Hours	Description
Front End	2	Parse returned data and create user session

Acceptance Criteria: Does the login page work. Can users create an account and then log in. Is the user's information, such as saved preferences saved.

4. **Users would like to be able to log out *From Sprint 1***

Team	Hours	Description
Front-End	1	Add a logout button to the right of the search bar or in the nav menu and route user from the current page.

Acceptance Criteria: On User logout, the current page redirects to the landing page

5. **Users would like to be able to change user settings such as their password, name, and dietary restrictions and then have the results update immediately *From Sprint 1***

Team	Hours	Description
Front-End	3	Create a settings page that the user can change preferences in.

Acceptance Criteria: Does the preferences page save changes? Change user information and reload - is the new information displayed?

6. **Users would like to be able to set favorited items by "pinning" the item or dining court card and then it would display under favorites *From Sprint 1***

Team	Hours	Description
Front-End	1	Add a favorites button to all the cards.
Front-End	5	Set preferences on backend, only if item is pinned and also create a favorites page where user can see pinned items.
Front-End	3	Add a way for users to see all the pinned items on the landing

Acceptance Criteria: Can the User click on the item / dining court and add it to their favorites. Check if the item / dining court shows under the favorites section.

7. **Users would like to be able to set user food preferences (allergies, dietary restrictions) under User Settings and then food item cards that match those preferences will not be displayed. *From Sprint 1***

Team	Hours	Description

Front-End	4	Create a settings page that the user can change preferences in.
Front-End	1	Set the user's preference with the API
Services	1	Check user prefs and return ones that pass

Acceptance Criteria: Does the preferences page persist changes? Search for menu that contains items that should be hidden based on allergies and see if item card is shown or not.

8. **As a user, I would like to be able to find foods around, greater than, or less than a certain calorie level with optional specification of dining court, meal time, date, and item.**

Team	Hours	Description
Tokenizer	1	Recognize calorie search
Front End	3.5	Get calorie search information from JSON object and filter. Users can then change their search parameters dynamically
Data Processing	1	Return calorie search information in the JSON object

Acceptance Criteria: Be able to search *"What can I get at Ford that is under 500 calories tomorrow for lunch."* Date, location, mealtime should all be queryable. User should be able to change the calorie values on the page after searching

9. **As a user, I would like to be able to find the shortest time to my destination. *Moved to Backlog***

Team	Hours	Description
Front-End	5	Create a bus page displaying all bus routes
Services	3	Create a way to search for all possible transfers within one stop to reach the destination. Craft SQL query
Data Processing	3	Find the shortest possible path out of these searches.
Database	2	Setting the tables. Create a script to update live data

Acceptance Criteria: User should be able to enter their current location and intended destination and have route options returned to them. The routes should be displayed on a map.

10. As a user, I would like to change my password if forgotten.

Team	Hours	Description
Front-End	3	Create a settings page that allows users to can change password

Acceptance Criteria: Does the preferences page save changes? Logout and login, does it work?

11. Users would like to be able to use the page without enabling unsafe scripts *Moved to Backlog*

Team	Hours	Description
Server	2	Create SSL certificate and enable without breaking the server

Acceptance Criteria: Can user search / load page and see information without enabling unsafe scripts

12. On login, check favorites and return alert

Team	Hours	Description
Data Processing	1	Check dining courts for favorited items
Front-End	2.5	Display Item cards / food items that have been favourited on the favourite page

Acceptance Criteria: Set food to favorites, logout and log back in. Food alert should pop up.

13. As a user, I would like to be able to search for food by meal name and time of the day

Team	Hours	Description
Data Processing	1	Get token and create SQL call

Acceptance Criteria: Be able to search "What is at Ford for Dinner today." Also be able to specify food items, etc...

14. As a user I would like to view a bus route of my choice.

Team	Hours	Description
Front-End	2.5	Display a specific bus route based on user choice
Services	3	Make a query that finds the route and returns all stops and buses on line
Data Processing	1	Process data in clean way for front end to read
Database	2	Setting the tables. Create a script to update live data

Acceptance Criteria: Users can filter bus routes on bus landing page. Live bus data is shown for routes that are selected

15. As a user I would like to view all buses / bus routes.

Team	Hours	Description
Front-End	5	Display all bus routes on the default bus landing page
Services	3	Make a query that finds the route and returns all stops and buses
Data Processing	1	Process data in clean way for front end to read
Database	2	Setting the tables. Create a script to update live data

Acceptance Criteria: On default, the bus landing page displays all routes using live data. The routes can be filtered as per user story 14.

16. As a user, I would like to be able to plan my trip in the future. *Moved to Backlog*

Team	Hours	Description
Front-End	3	Display bus routes according to the planned trip.
Services	2	Create way to query static data to find best possible route at certain time

Data Processing	1	Return to front end via rest call in clean JSON format
Database	1	Setting there tables for static data

Acceptance Criteria: User can enter starting location, intended destination, and starting time and routes will be returned based off of static / live data (depending on time). The routes will display on the map.

17. As a user, I would like the find the closest bus stop *New*

Acceptance Criteria: User can enter starting location and the closest bus stop will be returned

18. As a user, I would like to find the closest bus stop on a specific route *New*

Acceptance Criteria: User can enter starting location and intended route and closest stop will be returned

19. As a user, I would like to get incoming bus times at a specific stop

Team	Hours	Description
Front-End	3	Display bus routes according to the planned trip.
Services	2	Create way to query static data to find best possible route at certain time
Data Processing	1	Return to front end via rest call in clean JSON format
Database	1	Setting there tables for static data

Acceptance Criteria: User can enter starting location, intended destination, and starting time and routes will be returned based off of static / live data (depending on time). The routes will display on the map.

NON-FUNCTIONAL TIME CONSUMING TASKS

1. CSS improvements to a production ready state. While not extremely high up on the priority list, it will be something to work on if time permits.

2. Write unique file names for lexer and set up job to delete files every day. This will make sure that multiple requests won't overwhelm the lexer and have it overwrite a file before the file has been read and processed.

REMAINING BACKLOG

Backlog ID	Functional Requirements
1	As a user, I would like to be able to search for specific cuisines (food styles).
2	As a user, I would like to sort food items based on popularity, meal type and cuisine type.
3	As a user, I would like to be able to use location service to find the closest dining court.
4	As a user, I would like to have notifications sent to my phone about my favorite foods being served.
5	As a user, I would like to have a profile image.
6	As a user, I would like to be able to find the nearest bus.
7	As a user, I would like to find the estimate bus time.

Non- Functional

Backlog ID	Non-Functional Requirements
1	As a user, I would like to have a fast response time.
2	As a system admin, I would like to be able to scale the project up, if need be.
3	As a user, I would like to have access to this on Mobile.
4	As a developer, I would like to be able to add more APIs in the future.
5	As a developer, I want to be able to handle a large volume of traffic at any given point in time.

Sprint 2 Planning

6	As a user, I want a fast and secure way to log in.