Team 11

Purdue Eats

Team Members:

Aniket Agnihotri

Anisha Sinha

Eric Thompson

Mark Jin

Sean Joo

Vaastav Arora

Sprint 2 Planning Document

Overview

During this sprint, we hope to successfully allow the user to upload their schedule as well as a profile picture associated with their account. We also aim to let them track their dining dollar transactions and meal swipes as well as view the nutritional information of the items that they eat and get notified of their preferred items. One significant portion of this sprint on the backend is that we will be implementing the logic for predicting user meals as well as wait time at each dining court. Overall, this sprint has a lot of challenging components for each team member and it will take a lot of communication and hours to accomplish.

Scrum Master: Anisha Sinha

Meeting Schedule: Tuesdays/Saturdays @ 2:00pm, Thursdays @ 1:00pm

Risks and Challenges:

Despite our successful past sprint, we had difficulties thoroughly testing our implementations due to time constraints. It is important to be aware of the difficulty we had in the past and to be more proactive this sprint. Moreover, for most of our screens, we included dummy data instead of properly updating and fetching the correct real-time information, and this may require significant rewriting and reimplementation of components to accommodate real data in the current sprint. Finally, another challenge will be to properly train our GNN module to predict meals as well as train our XGBoost model to accurately predict wait times. The performance of both models is largely data-dependent and requires extensive tuning to bring them to acceptable accuracy levels. It is important to fulfill these challenges and the rest of the user stories as most of our components now and in the future rely on these implementations.

Current Sprint Detail

User Story #1

As a user, I would like to be able to see the correct menu items so that I can view what is offered at dining courts, record the right meal preferences, and record my meals.

#	Description	Estimated Time	Owner
1	Implement a function to fetch the right menu items with their allergen information based on the dining court.	2 hr	Anisha
2	Create a UI component to render the correct menu items based on the dining court on the menu page.	2 hrs	Anisha
3	Create a UI component to render the correct menu items based on the randomly selected items in the databases for meal preferences.	2 hrs	Anisha
4	Create a UI component to render the correct menu items based on the dining court on the record meal page.	2 hrs	Anisha
6	Manually test rendering of the correct menu items for menu, meal preferences, and record meal page.	1 hr	Anisha
7	**Total hours**	9 hrs	

Acceptance Criteria:

- Given that the menu items and allergens are fetched correctly, the correct items will be rendered on the menu page for each dining court.
- Given that the menu items are fetched correctly, the correct items will be rendered on the record meal page.
- Given that the menu items are fetched correctly, the correct items will be rendered on the meal preferences page.

User Story #2

As a user, I would like to be able to view a visual representation of nutritional information regarding all of my logged meals in the past week so that I can have a clear understanding of the health impacts of the food I am eating.

#	Description	Estimated Time	Owner
1	Create an UI panel to view a user's eating history, indicated through trends in macro consumption.	6 hrs	Aniket
2	Implement a corresponding API route for processing and sending meal-specific nutritional information.	4 hrs	Vaastav
3	Implement clearing of meal nutritional information weekly in the database.	3 hrs	Aniket
4	Manually test visual representation of nutritional information.	1 hr	Aniket
5	**Total hours**	14 hrs	

Acceptance Criteria:

- Given that the successive UI panels are implemented correctly, a user should be able to view the trends in macro consumption in their eating history.
- Given that the API route is implemented correctly, the information sent to the frontend should be timely and accurate based on the eating history of a particular user.
- Given that the visual representation functionality is implemented correctly, the data that is passed to the frontend should update over time as a user's logged eating history expands.

User Story #3

As a user, I would like to be able to view the specific nutritional information for menu items.

#	Description	Estimated Time	Owner
1	Create a UI panel to view all of a menu item's nutritional information.	5 hrs	Aniket
2	Create a visual representation for caloric intake and macros of the menu item.	5 hrs	Aniket
3	Implement a corresponding API route for processing and sending meal-specific nutritional information.	4 hrs	Vaastav
4	Manually test meal nutrition functionality.	1 hrs	Aniket
5	**Total hours**	15 hrs	

Acceptance Criteria:

- Given that the UI panel is implemented correctly, a user will be able to review all pertinent nutritional information of a menu item, including macros, calories, and serving size.
- Given that the visual representation is implemented correctly, a user will also be able to view a graphical representation of the macros and calories of a meal item.
- Given that the API route is implemented correctly, the information sent to the frontend should be timely and accurate based on what the Purdue Menus API dictates is currently being served.

User Story #4

As a user, I would like to be able to search for items on the menu page so that I can easily check for a specific item on the menu for the given dining court.

#	Description	Estimated Time	Owner
1	Create a search bar component for the user to look menu items up.	2 hrs	Anisha
2	Implement a function to render only the items that match what the user searches for	4 hrs	Anisha
3	Pre-process the menu items for the day into categories based on what allergen categories they fall under	3 hrs	Anisha
4	Render items correctly based on their allergen categories	2 hrs	Anisha
5	Manually test correct UI rendering and API Routes	1 hr	Anisha
6	**Total hours**	12 hrs	

Acceptance Criteria:

- Given that the preprocessing function works correctly, each item should be tagged correctly based on their allergens and be added to the respective data set.
- Given that the allergen tags for each menu item is correct, only items that contain the allergen will be displayed when that filter is applied.
- Given that the search bar is implemented correctly, only the item searched up will be rendered on the screen.

User Story #5

As a user, I would like to upload a profile picture to my profile so that others can see me when finding a lunch buddy through the app.

#	Description	Estimated Time	Owner
1	Create a UI component to prompt the user for profile picture input.	1 hr	Anisha
2	Write a function to upload image files from the user's device to the application.	3 hrs	Anisha
3	Display uploaded picture as user's profile picture on profile page.	3 hrs	Anisha
4	Create API route to forward user profile picture to server.	2 hrs	Vaastav
5	Create a server function to save uploaded pictures to the database.	2 hrs	Anisha
6	Manually test uploading and storing images.	1 hrs	Anisha
7	**Total hours**	12 hrs	

Acceptance Criteria:

- Given that the UI is properly laid out, the UI should display a component that the user can click to change their profile picture.
- Given that the application can connect to the user's device gallery, when the user chooses to update their profile picture they can choose pictures that they have saved to their device.
- Given that the application can display images, the picture uploaded by the user should be displayed as the user's profile picture.
- Given that the application and database can properly connect to each other, the user's profile picture should be saved to the database and associated with their account.

User Story #6

As a user, I would like the ability to make dining dollar transactions, as well as view these changes on my profile page.

#	Description	Estimated Time	Owner
1	Create UI to prompt user for dining dollar transaction input	3 hrs	Eric
2	Implement input validation on transaction amount	1 hr	Eric
3	Implement route to update dining dollar usage.	2 hrs	Eric
4	Implement UI so that dining dollar amount updates on profile page locally	2 hrs	Eric
5	Create unit test for UI and API route	2 hrs	Eric
6	**Total hours**	10 hrs	

Acceptance Criteria:

- Given that the UI is properly laid out, the UI should display a button that the user can click to make dining dollar transactions.
- Given the UI is implemented correctly, there should be a text field that allows the user to enter the amount of dining dollars they have used.
- Given the routes are implemented correctly, upon entering dining dollars into the text field the values should update in the database.
- Given the routes are implemented correctly, the values should update on the profile screen after the user updates dining dollars.
- Given that the route is implemented correctly, if an invalid value is entered changes should not be registered locally or in the database.

User Story #7

As a user, I would like the ability to make meal swipe transactions, as well as view these changes on my profile page.

#	Description	Estimated Time	Owner
1	Create UI to prompt user for meal swipe input	2 hrs	Eric
2	Implement input validation on transaction amount	1 hr	Eric
3	Implement swipe count so that it automatically resets at the end of each week	2 hrs	Eric
4	Implement route to update meal swipe usage.	2 hrs	Eric
5	Implement UI so that meal swipe amount updates on profile page locally	2 hrs	Eric
6	Create unit test for UI and API route	2 hrs	Eric
7	**Total hours**	11 hrs	

Acceptance Criteria:

- Given that the UI is properly laid out, the UI should display a button that the user can click to make meal swipe transactions.
- Given the routes are implemented correctly, upon subtracting a meal swipe the values should update in the database.
- Given the routes are implemented correctly, the values should update on the profile screen after the user updates meal swipes.
- Given that the route is implemented correctly, the swipe count should reset every week automatically.

User Story #8

As a user, I would like to have a meal history page so that I can view my past meal history.

#	Description	Estimated Time	Owner
1	Create a button on the profile page that allows users to navigate to the meal history page.	1 hr	Eric
2	Create UI page to display meal history information	3 hrs	Eric
3	Implement scrollable interface that allows user to view all past meal history	2 hrs	Eric
4	Implement routes to fetch meal information from database	3 hrs	Eric
5	Create unit test for UI and API route	1 hrs	Eric
6	**Total hours**	10 hrs	

Acceptance Criteria:

- Given that the UI is implemented correctly the user should be able to access the meal history page through a button on the profile page.
- Given the routes are implemented correctly the app should fetch all of the users' past meal information.
- Given that the UI is implemented correctly it will display the users' past meal information sorted from most recent to least recent.
- Given the UI is implemented correctly the user should be able to scroll through the meal history if the number of meals exceeds the amount that can fit on one page.

User Story #9

As a user, I would like to be able to mark meals as favorites so that I won't miss out on them when they are offered.

#	Description	Estimated Time	Owner
1	Create button on Profile page to redirect to user's favorite meals	1 hr	Sean
2	Create a UI panel for users to favorite menu items and view their current selection of favorite menu items	4 hrs	Sean
3	Have star icons show up next to menu items that are favorited.	3 hrs	Sean
4	Create an API route for the user's favorite meals (get, post, delete).	2 hrs	Vaastav
5	Connect backend and frontend of Favorite Meal together for GET and POST requests	3 hrs	Sean
6	Create manual tests to evaluate the functionality of favorite meals gets, posts, and updates successfully.	1 hr	Sean
7	**Total hours**	14 hrs	

Acceptance Criteria:

- Given that the button is implemented correctly, a user will be able to successfully be redirected to the favorited meals page when the button is clicked.
- Given that the UI panel is implemented correctly, the user should be able to favorite, update, and see their favorited menu items.
- Given that the star icons have been implemented correctly, a star will show up next to menu items that have been favorited.

User Story #10

As a user, I would like to customize my favorited meals to receive specific notifications.

#	Description	Estimated Time	Owner
1	Create notification UI with the user's current favorite meals list	5 hrs	Sean
2	Select specific items to turn notifications off or on	1 hr	Sean
3	Fetch the user's favorited meals and update the state of favorite meal route	4 hrs	Sean
4	Create manual tests for notification system	1 hr	Sean
5	**Total hours**	11 hrs	

Acceptance Criteria:

- Given that the fetch is implemented correctly, the user should be able to view their current list of favorited meals.
- Given that the selection is implemented correctly, the user should be able to toggle the notifications on or off .
- Given that the frontend and backend is merged correctly, the database should update depending on the selection.

User Story #11

As a user, I would like to be able to remove favorite meals when my preference changes.

#	Description	Estimated Time	Owner
1	Create UI panel to have the current selection of user's favorite meals	3 hrs	Sean
2	Create remove button to remove selected meals in real-time	1 hr	Sean
3	Merge frontend and backend to allow for DELETE and GET requests	2 hrs	Sean
4	Create manual tests to test the functionality of the toggle feature.	1 hr	Sean
5	**Total hours**	7 hrs	

Acceptance Criteria:

- Given the select feature has been implemented correctly, a user should be able to select one or multiple meals.
- Given that the remove button has been implemented correctly, a user should be able to remove selected favorite meals.
- Given that the frontend and backend has been merged correctly, the Favorite Meals UI should successfully complete a DELETE request when requested.

User Story #12

As a user, I would like to be able to input times that I am free to eat so that I am only recommended meals during those times.

#	Description	Estimated Time	Owner
1	Create a UI panel to enter a user's schedule of times that they are free to eat.	6 hrs	Aniket
2	Implement a corresponding API route for editing a user's schedule.	4 hrs	Vaastav
3	Configure database schema to store user account schedules.	2 hr	Aniket
4	Manually test scheduling functionality.	1 hrs	Aniket
5	**Total hours**	13 hrs	

Acceptance Criteria:

- Given that the UI panel is implemented correctly, a user will be able to successfully enter the times, as one hour increments, during which they are free to eat.
- Given that the corresponding API route is implemented correctly, the data forwarded from the UI will be stored for easy access by the API.
- Given that the corresponding API route is implemented correctly, a user's update to their corresponding schedule data will be made in real-time.

User Story #13

As a user, I would like to be able to easily view the present day's meal options across Purdue's various dining facilities so that I may choose what menu items to eat from the options available.

#	Description	Estimated Time	Owner
1	Implement a cron job that loads the current day's menu to the database	2 hrs	Vaastav
2	Create a cron job schedule that causes the cron jobs to run every morning between 5 am - 7 am	2 hrs	Vaastav
3	Implement server function to ensure there are no duplicate menu items present or added to database	2 hr	Vaastav
4	Implement API route that allows the client to fetch the current day's menu for each dining court	4 hrs	Vaastav
6	Create unit tests for API route	2 hrs	Vaastav
5	**Total hours**	12 hrs	

Acceptance Criteria:

- Given the cron job is implemented correctly, it should fetch menu items for the day and load them into the menu items dataset
- Given the cron job is implemented correctly, it should fetch menu items for the day and load them into the correct dining court dataset
- Given the cron schedule is implemented correctly, it should run 3 cron jobs at 5:30 am, 6:30 am and 7:30 am respectively
- Given that the API route corresponding to menu items is implemented correctly, the client should be able to fetch the current day's menu from the database

User Story #14

As a developer, I would like to construct a data preprocessing pipeline for our GNN model

#	Description	Estimated Time	Owner
1	Obtain and clean sample Kaggle data for preprocessing	2 hrs	Vaastav
2	Convert dataset into DGL graph	4 hrs	Vaastav
3	Assign feature set to created DGL graph	2 hrs	Vaastav
4	Implement test and train masks to pipe to GNN model	2 hrs	Vaastav
5	**Total hours**	10 hrs	

Acceptance Criteria:

- Given that the dataset is cleaned, it should be normalized and include no null data points
- Given that the dataset is cleaned, it should be savable and loadable as a .dgl file
- Given that the dataset is cleaned, all relevant feature sets should be assigned to the dgl graph

User Story #15

As a user, I would like to be given meal recommendations that are consistent with my past eating habits so that I can stick to eating foods that I enjoy and have eaten in the past.

#	Description	Estimated Time	Owner
1	Construct data pipeline from user database to GNN module	4 hrs	Vaastav
2	Create a predict function to serve recommendation requests	4 hrs	Vaastav
3	Implement API route to recommend meals	4 hrs	Vaastav
4	Create unit test for API route	2 hrs	Vaastav
5	**Total hours**	14 hrs	

Acceptance Criteria:

- Given that the GNN model is implemented correctly, it should be able to incorporate new data points acquired from the database
- Given the GNN model is implemented correctly, it should be able to perform predictions on certain users
- Given the API route is implemented correctly, predictions from the GNN model should be fetchable in the form of a JSON object

User Story #16

As a developer, I would like to create a dataset with dining menu logs and wait times so that I can use the data in our wait-time prediction modeling dataset.

#	Description	Estimated Time	Owner
1	Create function to merge historical wait-time dataset with historical menu item dataset	2 hrs	Mark
2	Create server function to fetch historical menu items from Purdue API	4 hrs	Mark
5	Conduct manual unit tests to verify proper implementation of data merge function	2 hrs	Mark
6	**Total hours**	8 hrs	

Acceptance Criteria:

- Given historical datasets are manipulated correctly, the wait-time and menu item datasets should be inner-joined by time and location.
- Given historical datasets are merged correctly, the dataset should include respective location, date, and menu items as attributes in the dataset, with no missing values.
- Given the server function is implemented correctly, a request to Purdue API should return the dates and menu items for every dining court of a given time frame.

User Story #17

As a developer, as part of the XGBoost modeling process, I would like to generate an importance matrix to identify the most important influencing factors affecting dining court wait-times so that I may use these factors for wait-time prediction.

#	Description	Estimated Time	Owner
1	Perform feature engineering to prep the dataset for modeling	4 hrs	Mark
2	Split dataset into train and test datasets and initialize hyperparameters and ML model	3 hr	Mark
3	Generate importance matrix, implement features into predictor algorithm	2 hrs	Mark
4	**Total hours**	9 hrs	

Acceptance Criteria:

- Given that the dataset has been manipulated and cleaned properly, the dataset should be able to be processed by the model without errors.
- Given the data has been split into test and train datasets, the datasets should have sufficient information and enough features for the model for learning and validation.
- Given the XGBoost model has been implemented successfully, an importance bar-plot should show the important features and their weight

User Story #18

As a developer, I would like to fine-tune the XGBoost model to accurately predict wait-times per dining court.

#	Description	Estimated Time	Owner
1	Create XGBRegressor model with sample dataset and actual dataset	3 hrs	Mark
2	Calibrate hyperparameters to increase prediction accuracy	6 hrs	Mark
3	Visualize prediction results	2 hrs	Mark
4	Test and evaluate model accuracy	3 hrs	Mark
5	**Total hours**	14 hrs	

Acceptance Criteria:

- Given the data is available and regressor model has been implemented successfully, the model should return a time-series estimation of the wait-times for per dining court.
- Given that the model has been calibrated properly, the accuracy of the model should be visible and of a sufficient score.
- Given that the visualization of the predicted wait-times is displayed and the data for the date is available, a developer can input a time and dining court through a function that would return a time-series plot of the wait-times for the specified dining court at the specified time

Remaining Backlog

Functional

1. **Account Registration, Login, and Management**
    1. ~~As a user, I would like to be able to register for a PurdueEats account so that all of my information is associated with my account.~~
    2. ~~As a user, I would like to be able to login to my PurdueEats account so that I can track my meals and view wait times.~~
    3. As a user, I would be able to login to my PurdueEats account via two-factor authentication so that my account is more secure.
    4. ~~As a user, I would like to be able to change my account name or password so that I have control over my credentials.~~
    5. ~~As a user, I would like to be able to enter my meal preferences so that I can have better meal recommendations based on my current likings.~~
    6. ~~As a user, I would like my password to be reset when required so that I can still access my account if I forget my password.~~
    7. ~~As a user, I would like to be able to delete my user account so that my user-generated data is removed from the application's database.~~
    8. ~~As a user, I would like to have a profile page so that I can edit my credentials, name, and meal plan as well as access my past meal history.~~
    9. ~~As a user, I would like to upload a profile picture to my profile so that others can see me when finding a lunch buddy through the app.~~
    10. As a user, I would like to have my login persist on exiting and re-entering the application so that I do not need to spend time logging in every time I would like to open the app.
2. **Application Navigation**
    1. ~~As a user, I would like to have a navigation bar so that I can access all of the features of the application from this screen.~~
3. **Dining Facilities and Menus**
    1. ~~As a user, I would like to be able to easily view meal options across Purdue's various dining facilities so that I may choose where to eat from the options I'm given.~~
    2. ~~As a user, I would like to be able to filter meals based on if they are vegetarian or contain specific allergens so that I can be sure that what I am eating fits my dietary needs.~~
    3. As a user, I would like to be able to view the wait-times at each of Purdue's dining facilities so that I can decide where it would be quickest for me to eat.
    4. ~~As a user, I would like to be able to view the distance from my location to each of Purdue's dining facilities so that I may account for distance to each facility as a factor in my dining decision-making.~~
    5. ~~As a user, I would like to be able to view a map of my current location and each dining court so that I can receive directions to any dining court of choice.~~
    6. ~~As a user, I would like to be able to mark meals as favorites so that I won't miss out on them when they are offered.~~
    7. ~~As a user, I would like to be able to remove favorite meals when my preference changes.~~
    8. ~~As a user, I would like to customize my favorited meals to receive specific notifications.~~

9. ~~As a user, I would like to be able to rate meals so that I get better meal recommendations based on my preferences.~~
10. ~~As a user, I would like to use a search bar to search through the dining court menus on any given day.~~
11. ~~As a user, I would like to be able to view the specific nutritional information for menu items.~~
12. ~~As a user, I would like to be able to see the correct menu items so that I can view what is offered at dining courts, record the right meal preferences, and record my meals.~~

4. **Meal Recommendation**
    1. ~~As a user, I would like to be able to input times that I am free to eat so that I am only recommended meals during those times.~~
    2. ~~As a user, I would like to be given meal recommendations that are consistent with my past eating habits so that I can stick to eating foods that I enjoy and have eaten in the past.~~
5. **Buddy System**
    1. As a user, I would like to be able to match with other users to dine at locations that meet our mutual dining preferences so that I can meet new people while still abiding by COVID-19 regulations.
    2. As a user, I would like to be able to share my current location with my matched user so that I can meet up with them and dine together.
    3. As a user, I would like to have a "I'm here" button so that I can let my matched user know that I have arrived at the agreed location.
6. **User Eating Habits**
    1. ~~As a user, I would like to be able to view a visual representation of nutritional information regarding my meals so that I can have a clear understanding of the health impacts of the food I am eating.~~
    2. ~~As a user, I would like to be able to view my past eating history so that I can keep track of my diet.~~
7. **Meal Plan Manager**
    1. ~~As a user, I would like to be able to view the number of meal swipes that I have remaining for the week so that I can manage my meal swipe count better.~~
    2. ~~As a user, I would like to be able to view my dining dollars balance so that I can track my transactions.~~
    3. ~~As a user, I would like to be able to update and edit my dining dollars balance so that I can make necessary changes if my meal plan changes.~~
8. **Meal Reviews and Ratings**
    1. As a user, I would like to be able to write reviews about dining courts so that I can share my experiences with others.
    2. As a user, I would like to be able to upvote or downvote reviews from others so that I can promote relevant reviews and demote irrelevant ones.
    3. As a user, I would like to be able to report reviews that are spam or inappropriate so that they can be taken down.
    4. As a user, I would like to be able to read reviews about dining courts so that I can learn more about a dining court from my fellow students before making a decision on where to eat.

9. **Accessibility**
    1. As a user, I would like to be able to adjust the app's text size so that I am able to clearly read information from the app.
    2. As a user, I would like to be able to choose between a dark mode and a light mode UI so that I can have a color scheme that suits my preferences.
    3. As a user, I would like to be able to access a settings page so that I can manage my notifications, light/dark mode UI, text size, and fun fact feature.
    4. As a user, I would like to be able to view answers to frequently asked questions so that if I run into issues with the app I can find potential solutions.
    5. As a user, I would like to be able to send feedback to the creators of the application within the application so that if there are any issues the developers can fix it.
    6. As a user, I would like to have a daily Purdue fun fact displayed each day that I log in so that I have an enjoyable user experience.
    7. As a user, I would like to disable the display of a Purdue fun fact so that I have a more minimalistic app experience.
10. **Summaries and Trends**
    1. As a user, I would like to view cumulative weekly summaries of what all users are eating through the app so that I can find popular eating trends.
    2. As a user, I would like to be able to see a wrap-up of the food that I have eaten over a semester so that I can look back on my eating habits for the semester.

Non-functional

1. ~~As a developer, I would like to create a dataset with dining menu logs and wait times so that I can use the data in our wait-time prediction modeling dataset.~~
2. ~~As a developer, as part of the XGBoost modeling process, I would like to generate an importance matrix to identify the most important influencing factors affecting dining court wait-times so that I may use these factors for wait-time prediction.~~
3. ~~As a developer, I would like to fine-tune the XGBoost model to accurately predict wait-times per dining court.~~
4. ~~As a developer, I would like to be able to visualize the model training process so that I can improve model performance based on new insights.~~
5. ~~As a developer, I would like to have custom GNN layer and model assets so that I can assemble a high performance GNN model engine~~
6. ~~As a developer, I would like to construct a data preprocessing pipeline for our GNN model~~
7. As a developer, I would like to be able to view database analytics so that I can track and scale the database.
8. As a developer, I would like to be able to view user feedback submitted through the application so that I can incrementally remove bugs and improve the application.
9. As a developer, I would like to view user data analytics so that I can improve user experience over time.
10. As a developer, I would like to view API interface documentation so that front-end and back-end developers can work independently.

11. As a developer, I would like the application to be containerized so that I have more control of the permissions and resources used by the application.