Foodie Connector

Sprint 3 Retrospective

Team 25: Zequan Wu, Shuqi Ma, Shiwen Xu, Xiaonan Shen

What went well?

User Story #26

As an order creator, I would like other users set themselves as ready before I could confirm the order so that we can make sure everyone has selected the items they want.

Sub-task	Estimated Time	Responsible Person
Create an API for other users to set themselves as ready	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface to for other users to set themselves as ready	5h	Shiwen Xu
Create a user interface to show the status of each group member to the group creator	6h	Shuqi Ma

Completed:

With this user story implemented, group members can set them as ready when they finish checking out. The creator can see everyone’s status in the order detail page.

User Story #27

As a user, I would like to see all nearby public orders that I can join.

Sub-task	Estimated Time	Responsible Person
Implement an API that returns all nearby public orders	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface to view all nearby public orders	5h	Zequan Wu

Completed:

With this user story implemented, users can select an address from saved addresses or current address to view all nearby public orders and join a public order.

User Story #28

As a user, I would like to split the total cost so that I only pay the price and delivery fee proportional to the food chosen by me.

Sub-task	Estimated Time	Responsible Person
Implement an API for showing the price	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface to show the price that the user need to pay	5h	Shiwen Xu

Completed:

With this user story implement, users can see the amount of money he/she needs to pay when they are checking out. The total price includes the price of the items from their carts, tax in current area, delivery fee and the tips.

User Story #29

As a user, I would like to follow friends.

Sub-task	Estimated Time	Responsible Person
Implement an API to following friends	1h	Xiaonan Shen
Implement an API to show a list of all friends	1h	Xiaonan Shen
Feature test for APIs	1h	Xiaonan Shen
Create a user interface to follow friends	5h	Shuqi Ma
Create a user interface to show all friends that have been followed in the friend list	6h	Shuqi Ma

Completed:

With the user story implement, users can follow friends by adding email. After the friend is successfully added to the list, the friend will appear on the friend list.

User Story #30

As a user, I would like to invite friends from the friend list to join the order.

Sub-task	Estimated Time	Responsible Person
Implement an API to send invitations to friends	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create user interface to invite friends from the friend list to join the order	5h	Shuqi Ma

Completed:

With this user story implement, users can invite friends from the friend list to join the order. The users who are invited will receive an email to accept the invitation. Once the users accept the invitation, they will be added into the order member list successfully.

User Story #31

As a user, I would like to track my order status.

Sub-task	Estimated Time	Responsible Person
Implement an API that gives the order status	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface to track the order status such as preparing, on the way, etc.	5h	Shiwen Xu

Completed:

With this user story implement, users can track their order status at order detail page.

User Story #32

As a user, I would like to track my driver's real-time location.

Sub-task	Estimated Time	Responsible Person
Setup websocket	1h	Xiaonan Shen
Implement an API through websocket that gives the real-time location of the driver	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface for users to track driver's real-time location from a map	5h	Shiwen Xu

Completed:

With this user story implemented, users can see an update of driver's position on google map every ten seconds at order detail page.

User Story #33

As a user, I would like to rate the restaurants from which I've ordered.

Sub-task	Estimated Time	Responsible Person
Implement an API to rate restaurants	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface to rate restaurants from which the user has ordered	5h	Zequan Wu

Completed:

With this user story implemented, users can rate the restaurants at order history page and change previous rates at anytime. When backend receives the rates, the latest average rates will be calculated.

User Story #34

As a user, I would like to see my order history so I could choose the same restaurant again easily.

Sub-task	Estimated Time	Responsible Person
Implement an API for showing order history	1h	Xiaonan Shen
Feature test for API	1h	Xiaonan Shen
Create a user interface to view order history	5h	Zequan Wu
Create a user interface to cancel unfinished orders created by the user	5h	Zequan Wu

Completed:

With user story implemented, users can view the detail of their order history and all information about past orders. The creator of the order can cancel unfinished orders at order history page.

User Story #39

As an administrator, I would like to manage restaurants' menus.

Sub-task	Estimated Time	Responsible Person
Setup menu management in the dashboard	3h	Xiaonan Shen

Completed:

With user story implemented, the administrator can edit the restaurants’ menus at administrator dashboard page.

User Story #40

As a user, I would like to receive notifications when my order status updated.

Sub-task	Estimated Time	Responsible Person
Setup Pusher	1h	Xiaonan Shen
Implement backend events to trigger notifications	2h	Xiaonan Shen
Create user interface to view notifications	5h	Zequan Wu

Completed:

With this user story implemented, users are able to receive updates of order status at header bar without refreshing the webpage whenever the backend triggers a notification.

User Story #25

As an order creator, I would like to be the only user who confirms the order.

Create a user interface only for the order creator to check out the order	3h	Shiwen Xu

Completed:

With the user story implemented, the creator of the order has the ability to check out the order and the status of each member will be shown.

User Story #18

As a user, I would like to create group orders.

Create a user interface to cancel the group order and the order becomes invalid	3h	Shuqi Ma

Completed:

With this user story implemented, the creator of the order is able to cancel the group order at order history page or restaurant detail page.

What did not go well?

We still have some minor problems and corner cases that are not taken considered due to the compact time.

How should we improve?

Out of all the user stories, we have finished most of the tasks of Sprint 3 so far, even though our project is not perfect and there still exist small bugs that need improvement. So far, we all have learned a lot from this course. Next, I will demonstrate what exactly we learned and how we are going to improve ourselves in the future.

First of all, in this project, each group member all learned new stuff by overcoming all kinds of tough problems. We separate our project to backend and frontend. We were using PHP for our backend development because it contains powerful built-in libraries. Additionally, we used MySQL as our database to store all information such as username, password, restaurant information and etc. In frontend, we take advantage of React because it contains lots of good sources and decent user interface. We used react.js not only for the better user interface, instead of piling up a simple HTML website.

Other than the technical skills, we also learned a lot of soft skill from this semester-long project. First of all, efficient communications save a lot of time. During the sprint 1, there were no much communications between people working on frontend and people working on backend about the APIs. That is the major cause of rewriting code due to the difference between mock data and real APIs. So, starting from sprint 2, we came to an agreement on those APIs before we actually started writing code, which saves a lot of time.
Additionally, collaboration. A good project is not built by one person. It requires the efforts of each team member. Everyone has something that is not good at, and something they are good at. Knowing how to distribute the work reasonably can maximize the use of each group member. Last but not least, negotiate is also essential to the group project. Everyone has different opinions on different things. When our opinions conflict, we learn how to resolve problems in a broader context.

Overall, each of us learned technical knowledge as well as important soft skills that are very useful for future collaboration. It is a very precious experience to work in a team like this. We will still keep improving ourselves and trying to do better in the future.