# Team Byte Me

## TEAM 07

---

# Deliverable 5

---

**Members:**
Yiyang Zhou
Yifei Yin
Ruixin Zhuang
Xinyi Chen
Min Qi Zhang

August 10, 2020

# Table of Contents

# Product Backlog - User Stories

## Shared User Stories

### Register, Login and Logout

**Note**: We have combined register, login and logout user stories. However, a lot of tasks/old user stories have been completed. To reduce possible confusion on user story/task numbers and story points calculations, we are not creating new issues for these user stories. Our old sprint backlogs, task board, burn-down chart, acceptance criteria and references to old user stories remain unchanged.

As a user, I want to register a new account with my email, username and password. (Estimated Cost: 9; Priority: 1)
- #52 for customer (Estimated Cost: 3)
- #55 for restaurant owner (Estimated Cost: 3)
- #78 for restaurant staff (Estimated Cost: 1)

As a user, I want to be able to login to the system with my email or username and password. (Estimated Cost: 10; Priority: 1)
- #54 for customer (Estimated Cost: 1)
- #31 for restaurant owner (Estimated Cost: 8)
- #77 for restaurant staff (Estimated Cost: 1)

As a logged-in user, I want to be able to logout of the application. (Estimated Cost: 4; Priority: 1)
- #53 for customer (Estimated Cost: 1)
- #56 for restaurant owner (Estimated Cost: 2)
- #79 for restaurant staff (Estimated Cost: 0.5)

## Customers

### View Restaurants and Promotions

**#3** As Jerry (a customer), I want to see a list of all the non-expired promotions of all the restaurants (including the promotion's title, the promotion's expired date and the restaurant that issues this promotion), so that I can participate and get benefits. (Estimated Cost: 3; Priority: 1)

**#4** As Jerry (a customer), I want to search restaurants by their names, so that I can find restaurants that I know. (Estimated Cost: 3; Priority: 3)

**#5** As Jerry (a customer), I want to check restaurants near my location, given a specific range. (Estimated Cost: 4; Priority: 3)

**#7** As Jerry (a customer), I want to sort promotions by the expiration dates, so that I will not miss any coupons. (Estimated Cost: 3; Priority: 2)

**#8** As Jerry (a customer), I want to see promotion details (i.e. the restaurant's name, the title, the description, the images, the expired date, the subtasks and the coupon of a promotion), so that I can know my progress about a promotion. (Estimated Cost: 4; Priority: 1)

### Participate in Promotions

**#10** As Jerry (a customer), I want to send a request to the restaurant about the subtask I am completing by pressing a button on the promotion detail page, so that the restaurant can validate my progress on the achievement. (Estimated Cost: 4; Priority: 1)

### View Coupons and Achievements

**#13** As Jerry (a customer), I want to browse all coupons available for me. (Estimated Cost: 5; Priority: 1)

**#14** As Jerry (a customer), I want to sort my coupons according to their expiration dates. (Estimated Cost: 3; Priority: 3)

**#16** As Jerry (a customer), I want to filter my coupons according to the restaurant types. (Estimated Cost: 3; Priority: 3)

**#18** As Jerry (a customer), I want to see an achievement map which shows flags on restaurants (with different colours) depending on how many promotions I have completed there. (Estimated Cost: 3; Priority: 3)

**#19** As Jerry (a customer), I want to see my current available points at a specific restaurant and items that I can redeem with my points. (Estimated Cost: 5; Priority: 3)

### Redeem

**#21** As Jerry (a customer), I want to send a request to the restaurant about the coupon I want to use by pressing a button on the coupon page, so that the restaurant can validate my coupon. (Estimated Cost: 3; Priority: 1)

**#22** As Jerry (a customer), I want to redeem coupons and items with my points. (Estimated Cost: 3; Priority: 3)

## Restaurant Staffs

### Validate Coupons and Promotions

**#82** As Adam (a restaurant staff), I want to see a list of customers that want to have their progress of achievements validated. (Estimated Cost: 3; Priority: 1)

**#83** As Adam (a restaurant staff), I want to see a list of customers that want to use their coupons. (Estimated Cost: 3; Priority: 1)

**#29** As Adam (a restaurant staff), I want to verify if a customer has completed a task by confirming the request message sent by the customer. (Estimated Cost: 3; Priority: 1)

**#28** As Adam (a restaurant staff), I want to validate customers' coupons by verifying the request message sent by the customer. (Estimated Cost: 3; Priority: 1)

## Restaurant Owners

### View Promotions

**#27** As Alice (a restaurant owner), I want to filter promotions according to their status (upcoming, active or expired). (Estimated Cost: 3; Priority: 3)

### Create Promotions

**#39** As Alice (a restaurant owner), I want to be able to see a list of promotions that I have posted. (Estimated Cost: 3; Priority: 1)

**#26** As Alice (a restaurant owner), I want to be able to create promotions with customizable titles, images, descriptions and other relevant information, so that promotion looks attractive, and rules are clear and transparent. (Estimated Cost: 7.5; Priority: 1)

**#23** As Alice (a restaurant owner), I want to customize rules for a promotion. For example, order one item from each of three categories within ten days. (Estimated Cost: 5; Priority: 1)

**#24** As Alice (a restaurant owner), I want to set rules for promotion by picking from a list of presets. For example, visit the restaurant ten times within a month. (Estimated Cost: 5; Priority: 1)

### Rewards and Points

**#20** As Alice (a restaurant owner), I want to be able to customize rewards for an achievement: give customers 100 points (more details specified below) or give them coupons (such as 20% off on any order, 5 dollars off on ordering this item) which they can use at our restaurant (Estimated Cost: 6; Priority: 1)

**#17** As Alice (a restaurant owner), I want to set what customers can redeem with their points. For example, they can redeem a free ice cream with 300 points or get a 50% off coupon with 200 points. (Estimated Cost: 6; Priority: 3)

**#15** (Epic) As Alice (a restaurant owner), I want to start a promotion collaboratively with other restaurant owners, and customers can get interchangeable coupons across both of the restaurants. (Estimated Cost: 6; Priority: 3)

### Analytics

**#11** As Alice (a restaurant owner), I want to see how many people participate in each of the promotions, so that I know which ones are more attractive. (Estimated Cost: 2; Priority: 3)

**#9** As Alice (a restaurant owner), I want to see how many people have obtained each of the coupons, so that I know which ones are more attractive. (Estimated Cost: 4; Priority: 3)

# Release Plan

There are no changes to our release plan.

# Sprint #5

## Sprint Backlog

### Stand-Alone Tasks

**#62** Fix bugs of the image uploading in the "create promotion" page. (Estimated Cost: 1.5; Priority: 1)

### User Stories & Tasks

#### Restaurant Owner

**#39** As Alice (a restaurant owner), I want to be able to see a list of promotions that my restaurant has. (Estimated Cost: 2; Priority: 1)

Tasks
- Get promotions of logged in restaurant owner (Estimated Cost: 1)
- Format and display promotion information to the view promotion page (Estimated Cost: 1)

Acceptance Criteria:
- Given that Alice (as restaurant owner) has logged into the system and there are promotions created for this restaurant, when they are on the promotions page, then they should see a list of all promotions with all the information displayed.
- Given that Alice (as restaurant owner) has logged into the system and there are no promotions created for this restaurant, when they are on the promotions page, then they should see a text saying there are no promotions.

**#20** As Alice (a restaurant owner), I want to be able to customize rewards for an achievement: give customers 100 points (more details specified below) or give them coupons (such as 20% off on any order, 5 dollars off on ordering this item) which they can use at our restaurant (Estimated Cost: 6; Priority: 1)

Tasks
- Popup for a description of coupon (Estimated Cost: 2)
- Popup for the number of points (Estimated Cost: 2)
- Store reward information to the database (Estimated Cost: 2)

Acceptance Criteria:
- Given that Alice (as restaurant owner) is on the promotion page, when she chooses points as a reward, then a number input box will show up for Alice to enter how many points she would like to send to customers for this reward.

- Given that Alice (as restaurant owner) is on the create promotion page, when she chooses a coupon as a reward, then "create a coupon" and "choose from the existing coupons" buttons will show up.
- Given that Alice (as restaurant owner) chooses a coupon as a reward, when Alice clicks the "create a coupon" button, then the coupon creating screen will show up.
- Given that the coupon creating screen pops up , when Alice enters valid information for all the required fields (coupon title, coupon description, coupon expired date) and clicks the "Save" button, then the coupon is created, the coupon creating screen is closed and Alice should see the newly created coupon on the promotion page.
- Given that the coupon creating screen pops up, when Alice enters invalid information for any of the required fields and clicks the "Save" button, then an error message will show up indicating the problem.
- Given that the coupon creating screen pops up but Alice no longer wants to create coupons for this promotion, when Alice clicks on the x button on the coupon creating screen, then the coupon creating screen is closed and no coupon is created.
- Given that Alice (as restaurant owner) chooses a coupon as a reward, when Alice clicks the "choose from the existing coupons" button, then the coupon selecting screen will show up and it will show a list of all the existing coupons of her restaurant.
- Given that the coupon selecting screen pops up, when Alice chooses one coupon for this promotion and clicks on the "Save" button, then the coupon is saved as the reward of the promotion, the coupon selecting screen is closed and Alice should see the coupon she chose on the promotion page.
- Given that the coupon selecting screen pops up but Alice no longer wants to select coupons for this promotion, when Alice clicks on the x button on the coupon selecting screen, then the coupon selecting screen is closed and no coupons are selected.

## Customer

**#52** As Jerry (a customer), I want to register a new account with my email, username and password. (Estimated Cost: 3; Priority: 1)

Acceptance Criteria:
- Given that Jerry (as a customer) is at the registration page, when he clicks "Already have an account? Login", then he should be redirected to the login page.
- Given that Jerry (as a customer) is at the registration page, when he enters valid email, username and password and presses on the "Register" button, then he should have a new account, be logged into the system and redirected to the home page.
- Given that Jerry (as a customer) is at the registration page, when he enters invalid email, username and password, then he should be warned of incorrect inputs.

**#53** As Jerry (a customer), I want to be able to logout of the application. (Estimated Cost: 1; Priority: 1)

Acceptance Criteria:
- Given that Jerry (as a customer) wants to log out, when he clicks the "Logout" button, then he should be logged out of the system and redirected to the login page.

**#8** As Jerry (a customer), I want to see promotion details (i.e. the restaurant's name, the title, the description, the images, the expired date, the subtasks and the coupon of a promotion), so that I can know my progress about a promotion. (Estimated Cost: 4; Priority: 1)

Tasks
- Create UI for displaying promotion details (Estimated Cost: 2.5)
- Get promotion details from the database (Estimated Cost: 1.5)

Acceptance Criteria:
- Given that Jerry (as a customer) has clicked into the promotion details, when he is on the detail page, he should see the restaurant name, promotion images, the promotion title, the promotion description, the promotion time period and his current progress.

**#3** As Jerry (a customer), I want to see a list of all the non-expired promotions of all the restaurants (including the promotion's title, the promotion's expired date and the restaurant that issues this promotion), so that I can participate and get benefits. (Estimated Cost: 3; Priority: 1)

Tasks
- Create UI for displaying promotions (Estimated Cost: 2)
- Get the required information of a promotion from the database (Estimated Cost: 1)
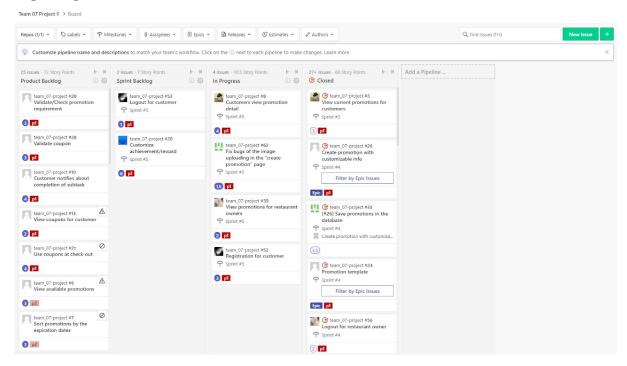
Acceptance Criteria:
- Given that Jerry (as a customer) has logged into the system, when he is on the home page, then he should see a list of all the non expired promotions.
- Given that Jerry (as a customer) has logged into the system, when he clicks the "view details" button of one of the promotions, then he should be redirected to another page to see the promotion details.
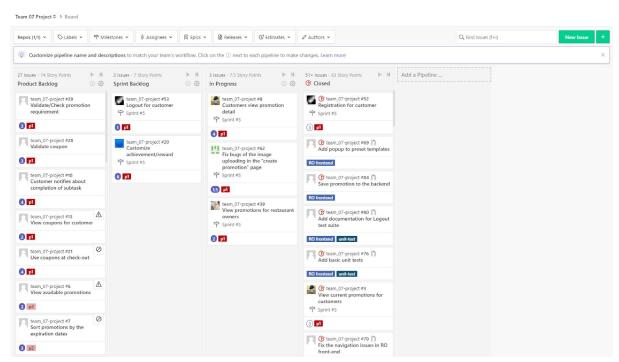
## Division of Labour

| Task Number | Cost | Priority | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Assignment |
|---|---|---|---|---|---|---|---|---|---|---|
| #62 | 1.5 | 1 | 0 | 1 | 0.5 | 0 | 0 | 0 | 0 | Xinyi Chen |
| #39 | 2 (1.5 left) | 1 | 0 | 1 | 0.5 | 0 | 0 | 0 | 0 | Min Qi Zhang |
| #52 | 3 (0.5 left) | 1 | 0 | 0.5 | 0 | 0 | 0 | 0 | 0 | Ruixin Zhuang |
| #53 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | Ruixin Zhuang |
| #8 | 4 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | Yiyang Zhou |
| #3 | 3(0.5 left) | 1 | 0.5 | 0 | 0 | 0 | 0 | 0 | 0 | Yiyang Zhou |
| #20 | 6 (4 left) | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | Yifei Yin |
| Total | 20.5 | | 0.5 | 4.5 | 4 | 2 | 2 | 0 | 0 | |
| Work Left | 20.5 | | 17.5 | 14.5 | 10 | 10 | 0 | 0 | 0 | |

# Task Board

## Beginning



## End
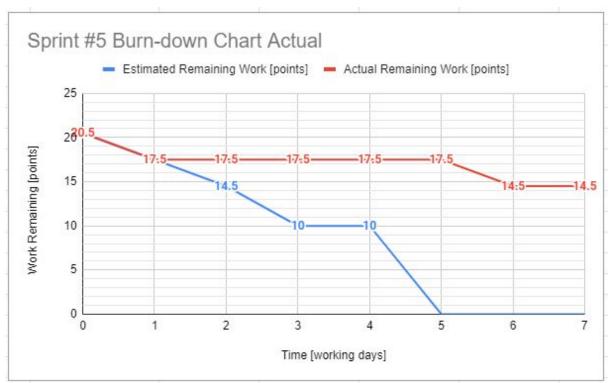
# Burn-down Chart

Beginning



End

# Sprint #6

## Sprint Backlog

### Stand-Alone Tasks

**#85** Fix the bug of missing jwt token when saving promotions to the backend. (Estimated Cost: 2; Priority: 1)

**#62** Fix bugs of the image uploading in the "create promotion" page. (Estimated Cost: 1.5; Priority: 1)

**#87** Merge restaurant staff and customer frontend. (Estimated Cost: 3; Priority: 1)

### User Stories & Tasks

#### Restaurant Owner

**#39** As Alice (a restaurant owner), I want to be able to see a list of promotions that my restaurant has. (Estimated Cost: 3; Priority: 1)

Tasks
- Get promotions of logged in restaurant owner (Estimated Cost: 1)
- Format and display promotion information to the view promotion page (Estimated Cost: 2)

Acceptance Criteria:
- Given that Alice (as restaurant owner) has logged into the system and there are promotions created for this restaurant, when they are on the promotions page, then they should see a list of all promotions with all the information displayed.
- Given that Alice (as restaurant owner) has logged into the system and there are no promotions created for this restaurant, when they are on the promotions page, then they should see a text saying there are no promotions.

**#20** As Alice (a restaurant owner), I want to be able to customize rewards for an achievement: give customers 100 points (more details specified below) or give them coupons (such as 20% off on any order, 5 dollars off on ordering this item) which they can use at our restaurant (Estimated Cost: 6; Priority: 1)

Tasks
- Popup for a description of coupon (Estimated Cost: 2)
- Popup for the number of points (Estimated Cost: 2)
- Store reward information to the database (Estimated Cost: 2)

Acceptance Criteria:

- Given that Alice (as restaurant owner) is on the promotion page, when she chooses points as a reward, then a number input box will show up for Alice to enter how many points she would like to send to customers for this reward.
- Given that Alice (as restaurant owner) is on the create promotion page, when she chooses a coupon as a reward, then "create a coupon" and "choose from the existing coupons" buttons will show up.
- Given that Alice (as restaurant owner) chooses a coupon as a reward, when Alice clicks the "create a coupon" button, then the coupon creating screen will show up.
- Given that the coupon creating screen pops up , when Alice enters valid information for all the required fields (coupon title, coupon description, coupon expired date) and clicks the "Save" button, then the coupon is created, the coupon creating screen is closed and Alice should see the newly created coupon on the promotion page.
- Given that the coupon creating screen pops up, when Alice enters invalid information for any of the required fields and clicks the "Save" button, then an error message will show up indicating the problem.
- Given that the coupon creating screen pops up but Alice no longer wants to create coupons for this promotion, when Alice clicks on the x button on the coupon creating screen, then the coupon creating screen is closed and no coupon is created.
- Given that Alice (as restaurant owner) chooses a coupon as a reward, when Alice clicks the "choose from the existing coupons" button, then the coupon selecting screen will show up and it will show a list of all the existing coupons of her restaurant.
- Given that the coupon selecting screen pops up, when Alice chooses one coupon for this promotion and clicks on the "Save" button, then the coupon is saved as the reward of the promotion, the coupon selecting screen is closed and Alice should see the coupon she chose on the promotion page.
- Given that the coupon selecting screen pops up but Alice no longer wants to select coupons for this promotion, when Alice clicks on the x button on the coupon selecting screen, then the coupon selecting screen is closed and no coupons are selected.

## Restaurant Staff

**#82** As Adam (a restaurant staff), I want to see a list of customers who want to have their progress of achievements validated and the content of their request message (including customer's user name and the description of subtask). (Estimated Cost: 3; Priority: 1)

Tasks:
- Get a list of validation requests and their content from the backend (Estimated Cost: 1)
- Create UI for viewing a list of requests (Estimated Cost: 2)

Acceptance Criteria:

- Given that Adam (a restaurant staff), is on the validation page, when there are pending requests (neither verified or rejected) to the restaurant that Adam is working at, then he should see all of them and each of them shows the customer's username and the description of subtask.
- Given that Adam (a restaurant staff), is on the validation page, when there is no pending request to the restaurant that Adam is working at, then he should see a prompt indicating that there is no request at the moment.

**#29** As Adam (a restaurant staff), I want to verify if a customer has completed a task by confirming the request message sent by the customer. (Estimated Cost: 3; Priority: 1)

Tasks:
- Update the status of the request to the backend (Estimated Cost: 1)
- Create UI for the reject and confirm button and the confirmation popup (Estimated Cost: 2)

Acceptance Criteria:
- Given that Adam (as restaurant owner) is on the validation page, when he presses the confirm or reject button, then a popup will show up to verify his action.
- Given that Adam (as restaurant owner) is prompt with the confirmation popup of the rejection, when he presses the yes button on the popup, then the popup will close and the status of this request will turn into rejected.
- Given that Adam (as restaurant owner) is prompt with the confirmation popup of the verification, when he presses the yes button on the popup, then the popup will close and the status of this request will turn into verified.
- Given that Adam (as restaurant owner) is prompt with the confirmation popup of the verification, when he presses the cross button on the top right of the popup, then the popup will close and the status of this request will not change.

**#77** As Adam (a restaurant staff), I want to be able to login to the system with my email and password. (Estimated Cost: 1; Priority: 1)

Acceptance Criteria:
- Given that Adam (as a restaurant staff) wants to use our product, when he opens the login page, then there should be two input fields on the page for email or username and password.
- Given that Adam (as a restaurant staff) has filled in the email/username and password correctly, when he clicks on the "Login" button, then he should be logged in and redirected to the home page.
- Given that Adam (as a restaurant staff) has filled in the email/username and password incorrectly, when he clicks on the "Login" button, then an error message shows up.

**#78** As Adam (a restaurant staff), I want to register a new account with my email and password. (Estimated Cost: 1; Priority: 1)

Acceptance Criteria:
- Given that Adam (as a restaurant staff) is at the registration page, when he clicks "Already have an account? Login", then he should be redirected to the login page.
- Given that Adam (as a restaurant staff) is at the registration page, when he enters valid email, username and password and presses on the "Register" button, then he should have a new account, be logged into the system and redirected to the home page.
- Given that Adam (as a restaurant staff) is at the registration page, when he enters invalid email, username and password, then he should be warned of incorrect inputs.

**#79** As Adam (a restaurant staff), I want to be able to logout of the application. (Estimated Cost: 0.5; Priority: 1)

Acceptance Criteria:
- Given that Adam (as a restaurant staff) wants to log out, when he clicks the "Logout" button, then he should be logged out of the system and redirected to the login page.

## Customer

**#53** As Jerry (a customer), I want to be able to logout of the application. (Estimated Cost: 1; Priority: 1)

Acceptance Criteria:
- Given that Jerry (as a customer) wants to log out, when he clicks the "Logout" button, then he should be logged out of the system and redirected to the login page.

**#8** As Jerry (a customer), I want to see promotion details (i.e. the restaurant's name, the title, the description, the images, the expired date, the subtasks and the coupon of a promotion), so that I can know my progress about a promotion. (Estimated Cost: 4; Priority: 1)

Tasks
- Create UI for displaying promotion details (Estimated Cost: 2.5)
- Get promotion details from the database (Estimated Cost: 1.5)

Acceptance Criteria:
- Given that Jerry (as a customer) has clicked into the promotion details, when he is on the detail page, he should see the restaurant name, promotion images, the promotion title, the promotion description, the promotion time period and his current progress.

**#13** As Jerry (a customer), I want to browse all coupons available for me. (Estimated Cost: 2; Priority: 1)

Tasks

- Create UI for displaying all coupons of current logged in customer (Estimated Cost: 1)
- Get information of coupons from the database (Estimated Cost: 1)

Acceptance Criteria:
- Given that Jerry (as a customer) is at the navigation menu, when he clicks the "coupons" tab, then he should be redirected to the coupon page.
- Given that Jerry (as a customer) is at the coupon page, when he looks at this page, then he should see all the coupons he can use or he has used with information about coupon description and restaurant name.
- Given that Jerry (as a customer) is at the coupon page, when he checks each coupon, then he should see a text saying "use it" for coupons that he can use and "used" for coupons that he has used.

**#10** As Jerry (a customer), I want to send a request to the restaurant about the subtask I am completing by pressing a button on the promotion detail page, so that the restaurant can validate my progress on the achievement. (Estimated Cost: 4; Priority: 1)

Tasks
- Add a "Participate" button at the promotion detail page (customer frontend). When this button is clicked, create progress and all subtasks in the database. (Estimated Cost: 1)
- Send a post request to the database for clicking a "Validate" button. (Estimated Cost: 1)
- Display subtask completion status. (Estimated Cost: 2)

Acceptance Criteria
- Given that Jerry (as a customer) is on the promotion detail page, when he clicks on the "Participate" button, then he should be able to see a list of subtasks, each followed by a "Validate" button.
- Given that Jerry (as a customer) has clicked the "Participate" button and saw a list of subtasks with the "Validate" button, when he clicks on the "Validate" button, then a validation request should be sent to the restaurant staff side.
- Given that Jerry (as a customer) has participated in a promotion and validated some subtasks, when he lands on this promotion detail page, then the validated subtasks are followed by the word "Completed."
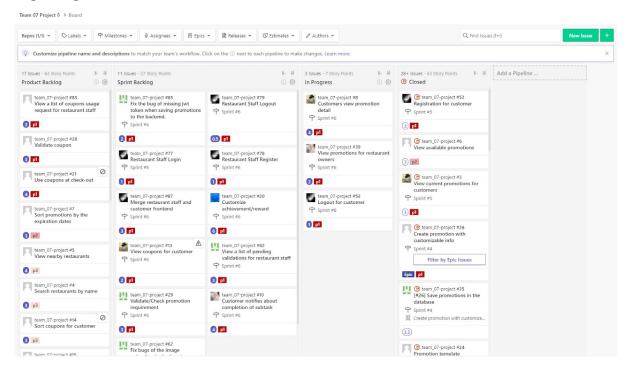
## Division of Labour

| Task Number | Cost | Priority | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Assignment |
|---|---|---|---|---|---|---|---|---|---|---|
| #85 | 2 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | Xinyi Chen |
| #62 | 1.5 | 1 | 0 | 0 | 1 | 0.5 | 0 | 0 | 0 | Xinyi Chen |
| #39 | 2 (1.5 left) | 1 | 0 | 0 | 0.5 | 1 | 0 | 0 | 0 | Min Qi Zhang |

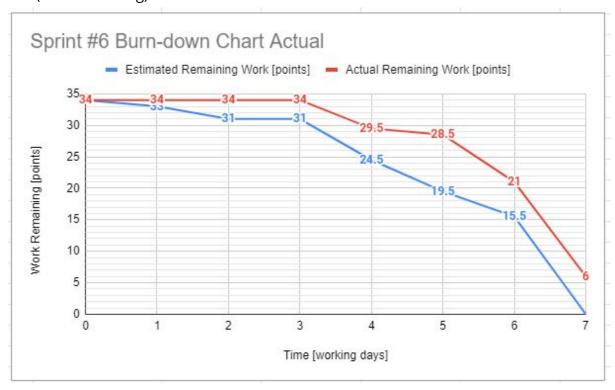| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| #53 | 1 (0.5 left) | 1 | 0.5 | 0 | 0 | 0 | 0 | 0 | 0 | Ruixin Zhuang |
| #8 | 4 | 1 | 0 | 0 | 1 | 2 | 1 | 0 | 0 | Yiyang Zhou |
| #20 | 6 (4 left) | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | Yifei Yin |
| #82 | 3 | 1 | 0 | 0 | 0 | 0 | 2 | 1 | 0 | Xinyi Chen |
| #87 | 3 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | Ruixin Zhuang |
| #13 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Yiyang Zhou |
| #29 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 2 | Xinyi Chen |
| #77 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Ruixin Zhuang |
| #78 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Ruixin Zhuang |
| #79 | 0.5 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | Ruixin Zhuang |
| #10 | 4 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 2 | Min Qi Zhang |
| Total | 34 | | 2.5 | 2 | 4.5 | 4.5 | 5 | 6 | 6.5 | |
| Work Left | 34 | | 33 | 31 | 31 | 24.5 | 19.5 | 15.5 | 0 | |

# Task Board

## Beginning



## End (and Interesting)



We have completed most of the planned tasks with two tasks left in progress.

# Burn-down Chart

Beginning



End (and Interesting)



We have burnt down 22.5 points in the last two days of the sprint.

# Sprint #7

## Sprint Backlog

### Stand-Alone Tasks

**#94** Check if promotion is complete (backend). (Estimated Cost: 5; Priority: 1)

### User Stories & Tasks

#### Restaurant Staff

**#83** As Adam (a restaurant staff), I want to see a list of customers that want to use their coupons and the content of their request message (including customer's user name and the description of coupon). (Estimated Cost: 3; Priority: 1)

Tasks:
- Get a list of validation requests and their content from the backend (Estimated Cost: 1)
- Create UI for viewing a list of requests (Estimated Cost: 2)

Acceptance Criteria:
- Given that Adam (a restaurant staff), is on the validation page, when there are pending requests (neither verified or rejected) to the restaurant that Adam is working at, then he should see all of them and each of them shows the customer's username and the description of the coupon.
- Given that Adam (a restaurant staff), is on the validation page, when there is no pending request to the restaurant that Adam is working at, then he should see a prompt indicating that there is no request at the moment.

**#28** As Adam (a restaurant staff), I want to validate customers' coupons by verifying the request message sent by the customer. (Estimated Cost: 3; Priority: 1)

Tasks:
- Update the status of the request to the backend (Estimated Cost: 1)
- Create UI for the reject and confirm button and the confirmation popup (Estimated Cost: 2)

Acceptance Criteria:
- Given that Adam (as restaurant owner) is on the validation page, when he presses the confirm or reject button, then a popup will show up to verify his action.
- Given that Adam (as restaurant owner) is prompt with the confirmation popup of the rejection, when he presses the yes button on the popup, then the popup will close and the status of this request will turn into rejected.

- Given that Adam (as restaurant owner) is prompt with the confirmation popup of the verification, when he presses the yes button on the popup, then the popup will close and the status of this request will turn into verified.
- Given that Adam (as restaurant owner) is prompt with the confirmation popup of the verification, when he presses the cross button on the top right of the popup, then the popup will close and the status of this request will not change.

## Customer

**#13** As Jerry (a customer), I want to browse all coupons available for me. (Estimated Cost: 5; Priority: 1)

Tasks
- Create UI for displaying all coupons of current logged in customer (Estimated Cost: 4)
- Get information of coupons from the database (Estimated Cost: 1)

Acceptance Criteria:
- Given that Jerry  (as a customer) is at the navigation menu, when he clicks the "coupons" tab, then he should be redirected to the coupon page.
- Given that Jerry  (as a customer) is at the coupon page, when he looks at this page, then he should see all the coupons he can use or he has used with information about coupon description and restaurant name.
- Given that Jerry  (as a customer) is at the coupon page, when he checks each coupon, then he should see a text saying "use it" for coupons that he can use and "used" for coupons that he has used.

**#10** As Jerry (a customer), I want to send a request to the restaurant about the subtask I am completing by pressing a button on the promotion detail page, so that the restaurant can validate my progress on the achievement. (Estimated Cost: 4; Priority: 1)

Tasks
- Add a "Participate" button at the promotion detail page (customer frontend). When this button is clicked, create progress and all subtasks in the database. (Estimated Cost: 1)
- Send a post request to the database for clicking a "Validate" button. (Estimated Cost: 1)
- Display subtask completion status. (Estimated Cost: 2)

Acceptance Criteria
- Given that Jerry (as a customer) is on the promotion detail page, when he clicks on the "Participate" button, then he should be able to see a list of subtasks, each followed by a "Validate" button.
- Given that Jerry (as a customer) has clicked the "Participate" button and saw a list of subtasks with the "Validate" button, when he clicks on the "Validate" button, then a validation request should be sent to the restaurant staff side.

- Given that Jerry (as a customer) has participated in a promotion and validated some subtasks, when he lands on this promotion detail page, then the validated subtasks are followed by the word "Completed."

**#21** As Jerry (a customer), I want to send a request to the restaurant about the coupon I want to use by pressing a button on the coupon page, so that the restaurant can validate my coupon. (Estimated Cost: 3; Priority: 1)

Tasks
- Add a button at the Coupon List page for each coupon (customer frontend). (Estimated Cost: 1)
- Create a coupon verification request in the database when the above button is clicked. (Estimated Cost: 2)

Acceptance Criteria:
- Given that Jerry (as a customer) is at the navigation menu, when he clicks the "coupons" tab, then he should be redirected to the coupon page.
- Given that Jerry (as a customer) is at the coupon page, when he looks at this page, then he should see all the coupons he can use or he has used with information about coupon description and restaurant name.
- Given that Jerry (as a customer) is at the coupon page, when he checks each coupon, then he should see a text saying "use it" for coupons that he can use and "used" for coupons that he has used.
- Given that Jerry (as a customer) is at the coupon page, when he clicks the "use it" button, then this sends a request to the restaurant staff for validation.

**#18** As Jerry (a customer), I want to see an achievement map which shows flags on restaurants (with different colours) depending on how many promotions I have completed there. (Estimated Cost: 3; Priority: 3)

Tasks:

- Import Google Map API
- Show a flag at the location of the restaurant if the customer gained some achievement from this restaurant
- Change the color of the flag based on the number of achievements the customer gained from this restaurant and add a popup to show the restaurant's name and the number of achievements that the customer has completed.

Acceptance Criteria:
- Given that Jerry (as a customer) is at the navigation menu, when he clicks the "Explore" tab, then he should be redirected to the achievement map page.
- Given that Jerry (as a customer) is at the achievement map page, when he looks at his page, then he should see the Toronto city map with some flags. The flag shows up at the restaurant's location. The color of it indicates how many achievements Jerry have completed there.
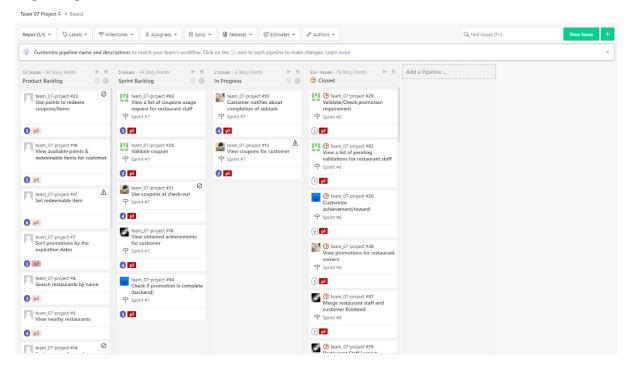
- Given that Jerry (as a customer) is at the achievement map page, when he clicks a flag, then he should see the restaurant name and the number of achievements he have completed there.
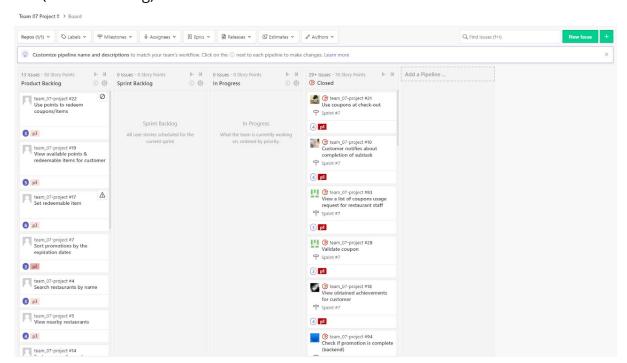
## Division of Labour

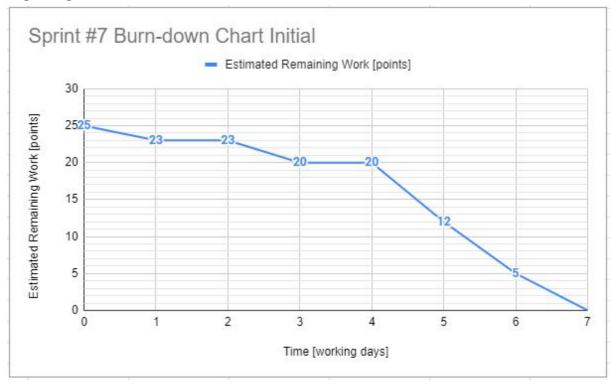| Task Number | Cost | Priority | Day 1 | Day 2 | Day 3 | Day 4 | Day 5 | Day 6 | Day 7 | Assignment |
|---|---|---|---|---|---|---|---|---|---|---|
| #13 | 2 (1 left) | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Yiyang Zhou |
| #10 | 4 (3.5 left) | 1 | 0 | 1 | 1 | 1 | 0.5 | 0 | 0 | Min Qi Zhang |
| #83 | 3 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Xinyi Chen |
| #28 | 3 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | Xinyi Chen |
| #21 | 4 | 1 | 0 | 0 | 1 | 1 | 2 | 0 | 0 | Yiyang Zhou |
| #18 | 4 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | Ruixin Zhuang |
| #94 | 5 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | Yifei Yin |
| Total | 25 | | 2 | 2 | 5 | 5 | 5.5 | 3 | 1 | |
| Work Left | 25 | | 23 | 23 | 20 | 20 | 12 | 5 | 0 | |

# Task Board

## Beginning



## End (and Interesting)
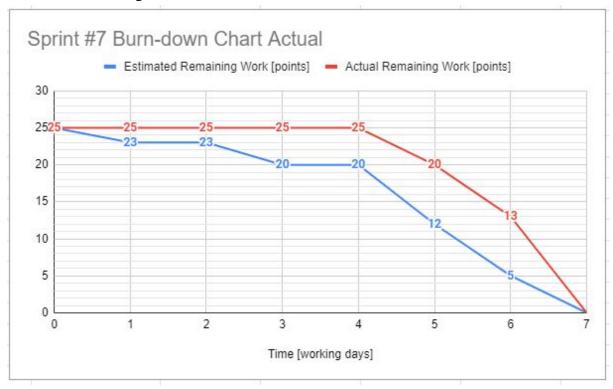


We completed all the tasks from the sprint backlog.
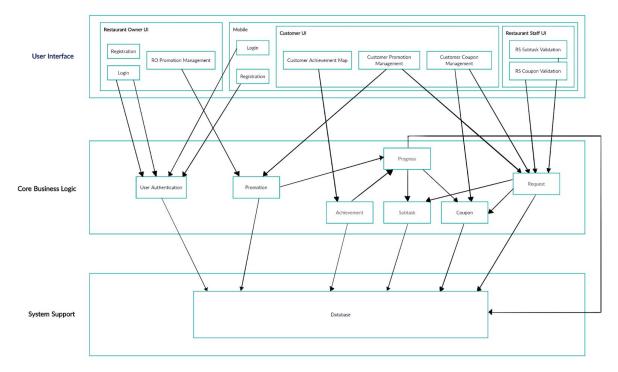
# Burn-down Chart

Beginning



End (and Interesting)



We have burnt 20 points in the last two days of the sprint and completed all the planned tasks. This is our first time of completing all the tasks in a sprint.

# Snapshots

We committed the snapshots every 48 hours, please check out this link to see all snapshots of task board and burn-down chart:

https://github.com/CSCC01/team_07-project/blob/master/Burn-Down%20Chart%20%26%20Task%20Board

# High-level Architecture



Higher resolution version:
https://user-images.githubusercontent.com/29013266/89834565-17f47a80-db31-11ea-8050-f15d6b7e50f9.png

# Changes Made to the Diagram

We have changed the components in the core business logic layer to match our actual code. We also changed the grouping in the User Interface layer. We removed the message sender component because it is just a request function in our actual code, integrated into Customer Promotion Management and Customer Coupon Management.

# Description of Each Component

For each dependency (except for the **Database**), we will describe the logic and responsibility on each side.

## User Interface

### Login

Note: Identical for all 2 login components

This component provides users with an interface for login. It depends on **User Authentication** to log in.

Dependencies

1.  This component depends on **User Authentication** to login. The request made to **User Authentication** includes:
    - identifier
    - password
    it will alert the user if the login credential does not match. It will redirect to the homepage if the login credential matches.

## Register

Note: Identical for all 2 register components

This component contains UI and functions for registration. It depends on **User Authentication** to register.

Dependencies

1.  This component depends on **User Authentication** to register. The request made to **User Authentication** includes:
    - username
    - email
    - password
    It would alert the user if the registration failed. It will redirect to the login page if the registration is successful.

## RO Promotion Management

This component provides restaurant owners with an interface for creating and viewing promotion. It depends on **Promotion** to create a new promotion and get promotion details.

Dependencies

1.  This component depends on **Promotion** to gather all the promotions that were created by the current restaurant. The request made to **Promotion** includes:
    - Restaurant id

The response will contain all promotions that were created by the current restaurant along with the following information:
- Promotion images
- Description
- Starting date
- Ending date
- Subtasks

Then **RO Promotion Management** will render these data on the page.

2. This component depends on **Promotion** to create a new promotion. The request made to **Promotion** includes:
   - Restaurant id
   - Promotion images (up to 5 images)
   - Description
   - Starting date
   - Ending date
   - Subtasks
   - Coupon description

   The response will tell this component whether creation is successful or not. It will display an alert about the result. If the creation is successful, it will redirect to the home page.

## Customer Promotion Management

This component provides customers with an interface for viewing all promotion information and sending requests for subtask validation. It depends on **Promotion** to get promotion information, depends on **Request** to send subtask validation requests and depends on **Progress** to get user-specific progress information of each promotion.

### Dependencies

1. This component depends on all the promotions that are currently available to customers (meaning promotions that are not expired). The requests made to **Promotion** do not include any data since the response should contain all currently available promotions.
   Then **Customer Promotion Management** will render the response on the front-end. **Promotion** will also provide customers' progress information for **Customer Promotion Management** to render. This is mentioned in the **Promotion** section below.

2. This component depends on the **Request** to send subtask validation requests. The request made to **Request** includes:
   - Restaurant id
   - Subtask id
   - User id
   - Type, which is "subtask"
   - Status, which is "pending"

   This request will create a new request object in the database with the above information. Restaurant staff can see those request objects and operate on them.

## Customer Coupon Management

This component provides customers with an interface for viewing coupons that customers can use or have used and sent requests for coupon validation.

### Dependencies

1. This component depends on **Coupon** to get all the coupons that current users can use or have used, and then render them in the front-end.
2. This component depends on **Request** to create a coupon request object in the back-end which can then be seen by restaurant staff. The information send to **Request** includes:
   - Restaurant id
   - User id
   - Coupon id
   - Type, which is "coupon"
   - Status, which is "pending"

## Customer Achievement Map

This component provides customers with an interface to see all the achievements they have made on a Toronto map. Each restaurant on the map has a "flag" which keeps track of the number of promotions the user has completed (i.e. achievement)

### Dependencies

1. This component only depends on **Achievement**. This component needs all the achievements of the current customer from **Achievement**. Request made to **Achievement** includes User id. The response from **Achievement** provides the following data for each achievement:
   - Restaurant id

- Restaurant location (latitude & longitude)
- Completion number

With these three pieces of information, **Customer Achievement Map** can draw each achievement and its location on the map.

## RS Subtask Validation

This component provides restaurant staff with an interface to confirm/reject a subtask validation request from customers.

### Dependencies

1. This component only depends on **Request**. It needs all the subtask request objects that belong to the restaurant that the current staff is working at, so that This component can render all the subtask requests that the staff should know on the front-end.
2. This component also depends on **Request** because it needs to solve a request by confirming/rejecting it. This will change the status of a subtask request object in the database from "pending" to "confirmed" or "rejected."

## RS Coupon Validation

This component provides restaurant staff with an interface to confirm/reject a coupon validation request from customers.

### Dependencies

1. This component only depends on **Request**. It needs all the coupon request objects that belong to the restaurant that the current staff is working at, so that This component can render all the coupon requests that the staff should know on the front-end.
2. This component also depends on **Request** because it needs to solve a request by confirming/rejecting it. This will change the status of a coupon request object in the database from "pending" to "confirmed" or "rejected."

## User Authentication

This component controls user authentication actions, including login and register. It depends on the Database for checking login credentials and creating new users.

### Dependencies

1. This component depends on the **Database** to check login credentials. It queries the database to see if login credentials match or not. Then it sends the result back to **Login**, as mentioned in the **Login** component description.
2. This component depends on the **Database** to create a new user. It queries the database to see if the username already exists. If not, it will create a new user in the database with the data provided. Then it sends the result back to **Registration**, as mentioned in the **Registration** component description.

# Core Business Logic

## Promotion

This component provides the information of all the promotions that have been created by all restaurant owners. It provides the information to **RO Promotion Management**, **Customer Promotion Management** and **Progress**.

### Dependencies

1. This component depends on the **Database** to obtain the promotion details (i.e. promotion title, description, expired date, starting date, etc.).

## Progress

This component manages the status of a promotion participated by a user. This component is the most important component since it is responsible for doing the following:

- Provide customer-specific progress information to **Promotion**, which can then be used for **Customer Promotion Management**
- Notify the **Achievement** when a progress is completed (more details in **Achievement** section)

### Dependencies

1. This component depends on the **Database** to get/modify all progress objects.
2. This component depends on the **Coupon**. It needs **Coupon** to create a coupon in the database when a progress is completed.
3. This component depends on the **Subtask**. Each progress stores a promotion_id. With promotion_id, It gets/creates all the subtasks in the promotion of that promotion_id from **Subtask**. This is useful because when a progress is created, it also creates all the subtasks. Also when a subtask is completed, the progress needs to get all the subtasks' status. If all subtasks are completed, the progress should change its status from "ongoing" to "completed." Thus, **Progress** relies heavily on the get and create methods of the **Subtask**.

## Subtask

This component manages all the subtasks. It does the following things:

- provides detail information (i.e. description, index and status) of a subtask for **Progress**, see details in the **Progress** section
- provides detail information (i.e. description, index and status) of a subtask for **Request**, see details in the **Request** section

### Dependencies

- This component depends on the **Database** to get/modify all subtask objects.

## Achievement

This component provides achievement information **Customer Achievement Map** to fetch.

### Dependencies

1. This component depends on the **Database** to get/modify all achievement objects.
2. This component depends on the **Progress** know when to modify its completion number (usually, increment the number by 1). The logic is that when a progress is completed, it will let **Achievement** know, and achievement will update its completion number. If this progress is from a new restaurant, **Achievement** will create a new achievement in the database with completion number initialized to 1.

## Coupon

This component managed all the coupons. It does the following things:

- provides detail information of a coupon for **Request**, see details in the **Request** section
- provides detail information of a coupon for **Progress**, see details in the **Progress** section
- provides user-specific coupon information to **Customer Coupon Management**

### Dependencies

1. This component depends on the **Database** to get/modify all coupon objects.

## Request

This is an important component that:

- allows **Customer Promotion Management** to create a subtask request object to the database

- allows **Customer Coupon Management** to create a coupon request object to the database
- allows **RS Subtask Validation** to change the status of a subtask request object
- allows **RS Coupon Validation** to change the status of a coupon request object

Dependencies

1. This component depends on the **Database** to get/modify all request objects.
2. This component depends on the **Coupon**. It needs to get a coupon by coupon_id from **Coupon** to provide coupon information to the **RS Coupon Validation**.
3. This component depends on the **Subtask**. It needs to get/modify a subtask by subtask_id from **Subtask** to provide subtask information to **RS Subtask Validation**.


# System Support

## Database

This is where all the data are stored.

# Verification and Validation

## Testing Strategy

Important components are tested. Each component is tested by a test suite which contains many test cases. All test suites are located right beside the component it tests, as recommended by [this official React documentation](). Each test suite has an overview of the tests it performs at the beginning of the file using JSDoc format. In addition to appropriate names given to test cases, a short description is also written before each test case. In general, each test case contains tests regarding the component's rendering results, functionalities and helper functions it uses.

Use [this GitHub search link]() to find all of our test cases.

# Meeting Log

## 2020-07-31

**Demo**

Since this was the first time Cho Yin looked at our product, we demoed all the features our app currently has. The following is a list of features we showed:

- restaurant staff and customer login (combined)
- restaurant staff and customer register (combined)
- restaurant owner login
- restaurant create promotion
- customer view promotion list

We also showed the features that are in progress with draft images:

- validation of subtasks and coupons
- message system

**Questions and suggestions**

Q: Can you give us any feedback or suggestions?
A: Good work so far. If we can see WebSocket notifications, that would be great.

Q: Is there any main feature or functionality you dislike?
A: Focus more on the gamification part.
Q: The point system is the core of our gamify part. We also have a level system. Do you think this is gamified?
A: You need to consider the WebSocket first. After that, I think levelling up is a good idea.

**How and when to address these feedback**

- We decided to try to use WebSocket during sprint #7; we will apply it to the message system so users can receive instant messages without refreshing the page.
- We will continue to come up with a solution for the gamified requirement during the following sprints.

**Links to the recordings**

- https://github.com/CSCC01/team_07-project/blob/master/Deliverables/Validation/2020-07-31%20PickEasy%20Demo%20with%20Cho%20Yin%20piece%201.mp4
- https://github.com/CSCC01/team_07-project/blob/master/Deliverables/Validation/2020-07-31%20PickEasy%20Demo%20with%20Cho%20Yin%20piece%202.mp4

## 2020-08-05



**Summary of question**

We have proposed a new gamify solution: when a user completes a promotion at a restaurant, a flag will show up on the map at the location of the restaurant; when the user completes a certain number of promotions, the colour of the flag will change to indicate upgrade. Does this feature satisfy the gamified requirement?
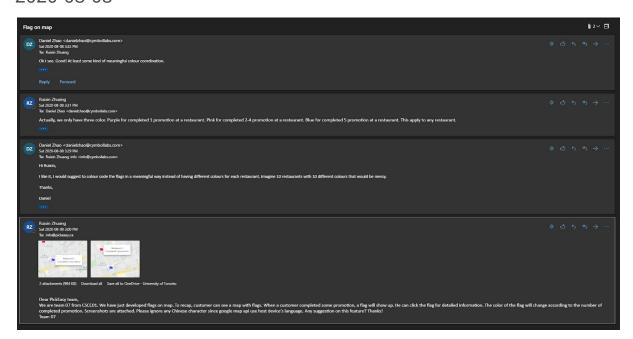
**When to address**
- We will address this new gamified solution during sprint #7.
- Update: we implemented this new feature in sprint #7.

**Link to the screenshot**

https://github.com/CSCC01/team_07-project/blob/master/Deliverables/Validation/Gamified.png

## 2020-08-08

**Question and suggestion**

Q: We have implemented flags on the map. (Recap of this feature). Is there any suggestion on this feature?

A: I would suggest the colour code the flags in a meaningful way instead of having different colours for each restaurant.

Q: We only have three colours. Purple for completing 1 promotion. Pink for completing 2-4 promotions. Blue for completing 5 promotions. This applies to any restaurant.

A: Good!

**When to address**

- We already implemented this new feature in sprint #7.

**Link to the screenshot**

https://github.com/CSCC01/team_07-project/blob/master/Deliverables/Validation/Flag%20on%20map.png

# Retrospection

Throughout this term, our project has gone through 7 sprints, and our project velocity is shown in the chart below.

| Sprint # | Estimated Velocity | Actual Velocity |
|---|---|---|
| 1(deliverable 3) | 26 | 16 |
| 2 (deliverable 4) | 17 | 4 |
| 3 (deliverable 4) | 32 | 26 |
| 4 (deliverable 4) | 29 | 11.5 |
| 5 (deliverable 5) | 20.5 | 6 |
| 6 (deliverable 5) | 34 | 28 |
| 7 (deliverable 5) | 25 | 25 |

## What was our estimated project velocity?

### Sprint #1

The estimated velocity for sprint #1 was 26. Since this is our first sprint, we have no idea how many points we can complete within a sprint.

### Sprint #2

The estimated velocity for sprint #2 was 17. During this sprint, we had midterms. Therefore, for sprint #2, we planned relatively less work compare to sprint #1.

### Sprint #3

The estimated velocity for sprint #3 was 32. During this sprint, our schedule was less-occupied, so we planned more points in sprint #3. We planned more work than sprint #1 because, by that point of time, we were getting more familiar with the framework, we can complete more tasks.

### Sprint #4

The estimated velocity for sprint #4 was 29. We planned almost the same amount of work as sprint #3.

### Sprint #5

The estimated velocity for sprint #5 was 20.5. During this sprint, we all have to work on other course work. Thus, in this sprint, we did not plan new tasks; we simply planned the leftover user stories from the previous sprint and got 20.5 estimated velocity.

### Sprint #6

The estimated velocity for sprint #6 was 34. During this sprint, we have a less-occupied schedule this week, and this is the second last sprint, so we planned more tasks. We knew that our highest estimated velocity was 32, and that was our maximum capability, so for this sprint, we estimated 34 points, which was slightly more than sprint #3.

### Sprint #7

The estimated velocity for sprint #7 was 25. We added all the priority 1 user stories left from the product backlog into the sprint backlog during this sprint. Luckily, the sum of those user story points did not exceed our maximum capability.

## What was our actual project velocity?

### Sprint #1

We completed 16 points.

### Sprint #2

We completed 4 points. Unfortunately, we were busy during this sprint, so we were not able to complete many tasks. Also, we did not re-plan this sprint.

### Sprint #3

We completed 26 points. We were able to burn down many points because we have a relatively less-occupied schedule in this sprint.

### Sprint #4

We completed 11.5 points. Since this is the last sprint of deliverable 4, we also need to write the deliverable 4 simultaneously. Writing deverable 4 significantly reduced the burndown velocity. We raised the point of user story #35 at the beginning of the sprint because we realized that storing promotion to the database requires more time to implement.

### Sprint #5

We only completed 6 points. We fell into the same situation as sprint #2, so we could not complete many tasks.

### Sprint #6

We completed 28 points, which was close to the estimated velocity.

We completed all the story points for this sprint.

## How did our project progress throughout the term?

### Sprint #1, #2, #3 and #4

Before deliverable 3, we set up the environment with Strapi as the back-end, React as the front-end and ESLint as the code style checker. After validating with the clients, we decided to prioritize creating and customizing promotions, as they are the most important features. Thus, in addition to setting up the environment, we spent our first sprint to implement a user interface for these features.

In the sprint #2, #3 and #4, we focused on the basic functionalities on the customer side because clients wanted to see a customer front-end. Our second sprint was not very productive because of midterms, and during the third sprint we spent most of our time learning writing unit tests in React. Only in the third sprint, we made significant progress on the customer front-end application. We also took many suggestions from clients and modified our design accordingly. For example, we changed the customer interface to mainly focus on promotions rather than restaurants.

### Sprint #5, #6 and #7

During sprint #5, #6 and #7, we have done a lot of refactoring in our code base. Other than that, we focused on the validation process of coupons and subtasks and the achievement system (a gamified feature to attract customers as required by our clients); as mentioned by our TA and clients, validation and gamification are the two important features that we still lack. Besides client requirements, we also did some decorations to the interface for the final presentation.

## Did we follow our plan(s) exactly? What difficulties have we encountered?

To be honest, we did not follow our plans exactly throughout the project. We did not do well in sprint #1, #2, #4 and #5. In the first sprint, many of us were still learning the frameworks we are using. However, we considered such difficulties in sprint #1 were merely temporary. In sprint #2 and #5, we were busy with other courses and thus we were not very productive. Actually, we had predicted such situations, so we arranged relatively low story points for those two sprints.

In sprint #4, we had some difficulties from clients. They wanted us to use a different customer interface. We used to think that homepage contains a list of restaurants. However, clients wanted a list of promotions. Therefore, we have to replan our user stories and change some code.

Like we have mentioned, we knew that we would not have much time for sprint #5, so we put most of our work in sprint #6 and #7. Based on previous experience, we thought a productive yet doable sprint should have about 30 story points. We knew the last two sprints may be a bit overwhelming, but we were under the pressure to produce a good and finished project. Therefore, we decided to push ourselves a little and do two heavy sprints with velocity 34 and 25.

When we were planning for sprint #7, we realized that the point system is too much for us to implement within a sprint. We might not be able to satisfy the gamify requirement since the point system is the core of gamification in our product. We came up with another gamified solution, which takes less time to implement and satisfy the requirement. We received the confirmation from our client to implement this feature. Thus, our last sprint has an estimated velocity of 25, which is within our capability.

## Was our contingency plan useful at those points or did we have to come up with new solutions?

Fortunately, we have almost never used our contingency plan because no one has dropped the course or called in sick. Occasionally, people told the group that he/she cannot come to the meeting; they did this ahead of the time just like we agreed on a contingency plan. We usually changed our meeting time to fit everyone in such cases.

## What is the most important thing we learned from working on this project?

After those 7 sprints, we have learnt two important lessons:
1. Client requirements including UI elements and interaction logic should be clarified as early as possible. The cost to replan and change code will be much dearer later.
2. Database should be well-organized from the beginning and changed carefully in later development. The basic structure of the project should be made crystal clear to all members. Refactoring can be very time-consuming.

## How does the work done for deliverable 5 compare to the work on all previous deliverables, both in terms of progress and end result?

In deliverable 5, we have completed a significant number of tasks. Especially the last two sprints, we have completed 53 points, which is almost equal to the total of actual work done of sprint #1, #2, #3 and #5. Moreover, we followed closely to our plan in those two sprints. In contrast, we did not perfectly follow the plan during our previous sprints (sprint #1, #2, #4 and #5).

Compared to previous sprints, we followed the client requirements more closely; we verified all our ideas with clients ahead of time. After refactoring the back-end, our team also worked more efficiently. Yet, we did spend some time finding out an interesting

way to gamify the achievement system. But in the end, we think clients love the great idea from Xinyi Chen.

## Changes to Product Backlog

1. We have lowered the cost for #78 and #79 because we already implemented the code for register and logout; all we have to do is to reuse the code.
    - **Before**: the cost of **#78** was 3 and the cost of **#79** was 1
    - **After**: the cost of **#78** is 1 and the cost of **#79** is 0.5

2. We have deleted user story #6. It is a duplication of #3. From the previous deliverable, we revised user story #3 but was not aware that #3 was similar to #6 until deliverable 5; so we decided to delete #6.
    - **#6** As Jerry (a customer), I want to see a list of available promotions, so that I can participate and get benefits.
3. We have modified user story #8 to make it more estimable.
    - **Before: #8** As Jerry (a customer), I want to see promotion details, so that I can know my  progress and other information about a promotion.
    - **After: #8** As Jerry (a customer), I want to see promotion details (i.e. the restaurant's name, the title, the description, the images, the expired date, the subtasks and the coupon of a promotion), so that I can know my progress about a promotion.
4. We have modified user story #3 to make it more estimable.
    - **Before: #3** As Jerry (a customer), I want to see a list of all the current promotions.
    - **After: #3** As Jerry (a customer), I want to see a list of all the non-expired promotions of all the restaurants (including the promotion's title, the promotion's expired date and the restaurant that issues this promotion), so that I can participate and get benefits.

5. We have modified user story #10 to make the requirement more clear.
    - **Before: #10** As Jerry (a customer), I want to notify the restaurant about the subtask I am completing by pressing a button on the app, so the restaurant can validate my progress on the achievement.
    - **After: #10** As Jerry (a customer), I want to send a request to the restaurant about the subtask I am completing by pressing a button on the promotion detail page, so that the restaurant can validate my progress on the achievement.

6. We have raised the cost of #13 from 2 to 5 because after restructuring the database, this task now involves some adjustment to previous codes.

7. We have modified user story #18 to make it more estimable by providing more details.
    - **Before: #18** As Jerry (a customer), I want to see obtained achievements for each restaurant.
    - **After:  #18** As Jerry (a customer), I want to see an achievement map which shows flags on restaurants (with different colours) depending on how many promotions I have completed there.

8. We have modified user story #21
   - **Before: #21** As Jerry (a customer), I want to notify the restaurant about the coupon I am using by pressing a button on the app.
   - **After: #21** As Jerry (a customer), I want to send a request to the restaurant about the coupon I want to use by pressing a button on the coupon page, so that the restaurant can validate my coupon.

9. We have modified user story #82
   - **Before: #82** As Adam (a restaurant staff), I want to see a list of customers that want to have their progress of achievements validated.
   - **After: #82** As Adam (a restaurant staff), I want to see a list of customers who want to have their progress of achievements validated and the content of their request message (including customer's user name and the description of subtask).

10. We have modified user story #83
    - **Before: #83** As Adam (a restaurant staff), I want to see a list of customers that want to use their coupons.
    - **After: #83** As Adam (a restaurant staff), I want to see a list of customers that want to use their coupons and the content of their request message (including customer's user name and the description of coupon).

11. We have raised the cost of user story #39 from 2 to 3 because we spent more time on finding a solution of formatting the images nicely.