**HungerMate Feature 2 Technical Document**

**By:**

**Christopher Mendoza**

**Haricharan Mukthineni**

**Gabriel Braun**

**For:**

**SWEN 5239.32 Agile Software Development**

**Dr. Soma Datta**

**April 22nd, 2024**

**Contents**

Project Features…………………………………………………………………………………………3

Meetings……………………………………………………………………………………………….... 4

Working Agreement……………………………………………………………………………………..6

Sprints…………………………………………………………………………………………………….7

Sprint 1…………………………………………………………………………………………...7

Sprint 2……………………………………………………………………………………………7

Burnup/BurndownCharts ……………………………………………………………………………….7

Backlogs………………………………………………………………………………………………….9

User Stories ……………………………………………………………………………………………11

Code………………………………………………………………………………………………………53

Main……………………………………………………………………………………………...53

Resources………………………………………………………………………………………………..64

## **Project Features**

The purpose of this application is to allow a user to place an order for food from multiple restaurants.

**UI:**

* Have a main menu which gives the user the option of placing an order or exiting the application
* Have a submenu that allows users to filter the type of food they want and view restaurants according to selection
* Have a submenu that allows the user to select any restaurant and view the restaurant’s menu
* Have a submenu that allows the user to select food items, input customizations
* Have a submenu that allows users to view their current cart.
* Have a checkout submenu that allows users to input delivery methods, enter payment information, and receive a confirmation page.

**Back-end**

* Create a database with a list of restaurants.
* Create a database with the menus of restaurants.
* Create a database that stores users' orders.

## Meetings

| **Meeting Date** | **What have you done since last time?** | **What will you do now?** | **What have you completed?** |
| --- | --- | --- | --- |
| 03/31/2024 | n/a | HungerMate Proposal | n/a |
| 04/04/2024 | HungerMate Proposal | User Stories | HungerMate Proposal |
| 04/06/2024 | User Stories | Create JIRA board  Create GitHub working environment | User Stories |
| 04/09/2024 | Created JIRA board  Created GitHub working environment | Create the MySQL databases | Created JIRA board  Created GitHub working environment |
| 04/11/2024 | Begun work on main function for program | Continue work on main function | n/a |
| 04/13/2024 | Created the database and set up the tables for restaurant, menus, and orders | Continue work on main function | Created the database and set up the tables for restaurant, menus, and orders |
| 04/14/2024 | Created a main function.  Created the filter function.  Created the view restaurant function | Work on Project Feature 1 documentation  Plan for next sprint | Project feature 1 documentation |
| 04/17/2024 | Project feature 1 documentation | Begin working on tasks for sprint 2 | n/a |
| 04/20/2024 | PO6: Customization of order  PO4: Food Selection Function  Update restaurant table  Update menu table | Update restaurant table  Update menu table | PO6: Customization of order  PO4: Food Selection Function |
| 04/21/2024 | Update restaurant table  Update menu table  PO2: Restaurant choice function  PO3: Menu choice or selection | PO2: Restaurant choice function  PO3: Menu choice or selection | Update restaurant table  Update menu table |
| 04/22/2024 | PO2: Restaurant choice function  PO3: Menu choice or selection | Project feature 2 documentation | Project feature 2 documentation |

## **Working Agreement**

1. We will meet on Thursdays and Saturdays at 6 p.m., with additional meetings being scheduled if needed.
2. Daily 15 minute scrum meetings are held in Discord.
3. Members that miss meetings will be informed of the meeting's discussion and results.
4. A Discord server has been set up to communicate easily with each other.
5. We have a Google Drive where we will store our artifacts (documents and PowerPoints).
6. Code will be shared via a GitHub repository.
7. We are planning to use C++ as our programming language and will export our database tables as CSV files.
8. Any issues encountered will be reported to the rest of the team.
9. All team members' opinions will be considered, and everybody should agree before any major changes are made.

## Sprints

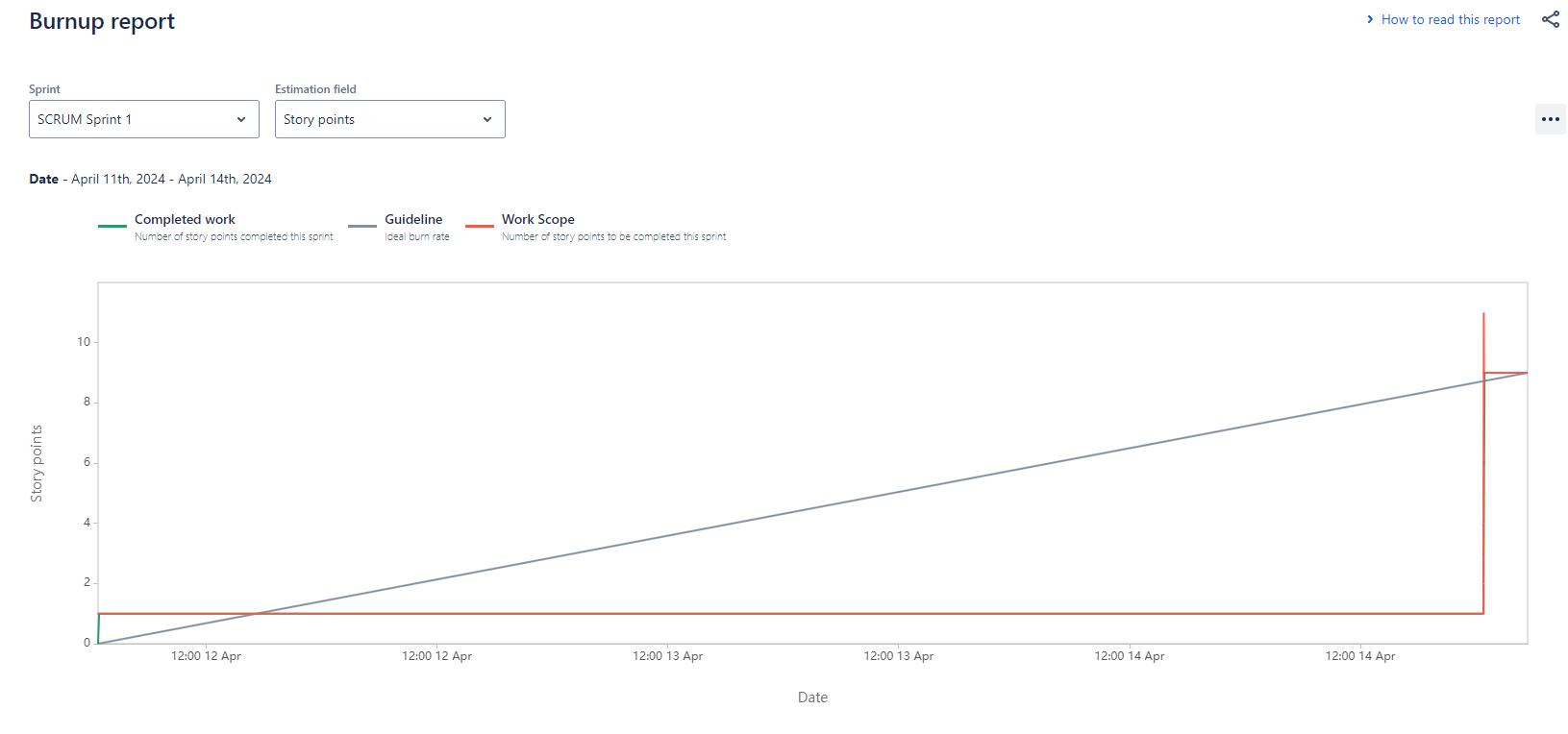
Sprint 1

In the first sprint, the group got together to create user stories for the application. All user stories were then placed in a JIRA Scrum board so tasks could be distributed. In this sprint, databases tables for the restaurant information and the menus were created. The group also began writing the code which allows the user to view restaurants and filter the restaurants by cuisine type.

Sprint 2

In the second sprint the team updated SQL databases for the available restaurants and menus. In this sprint the team also developed the restaurant selection, menu display, food item selection, and customization features of the application.

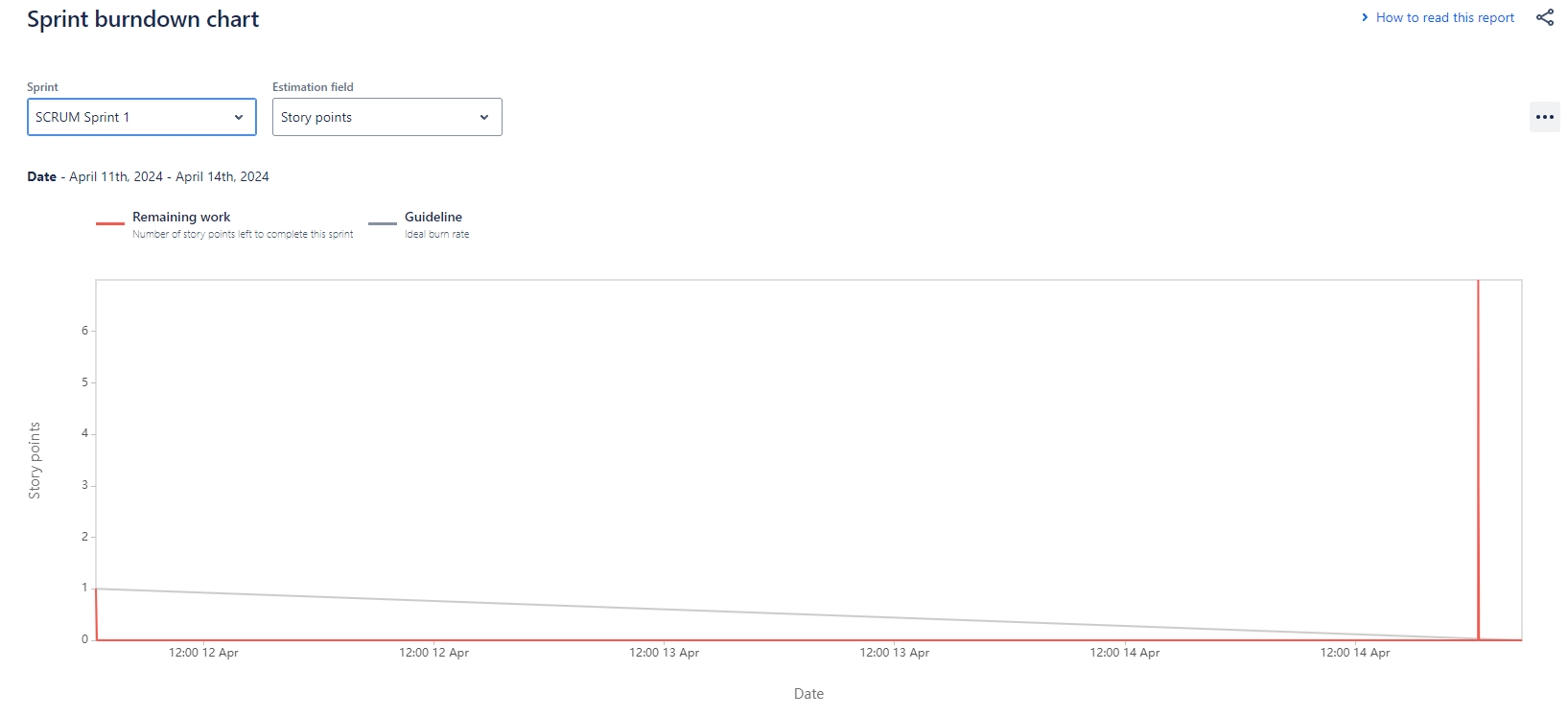
Sprint 1 Burnup Chart



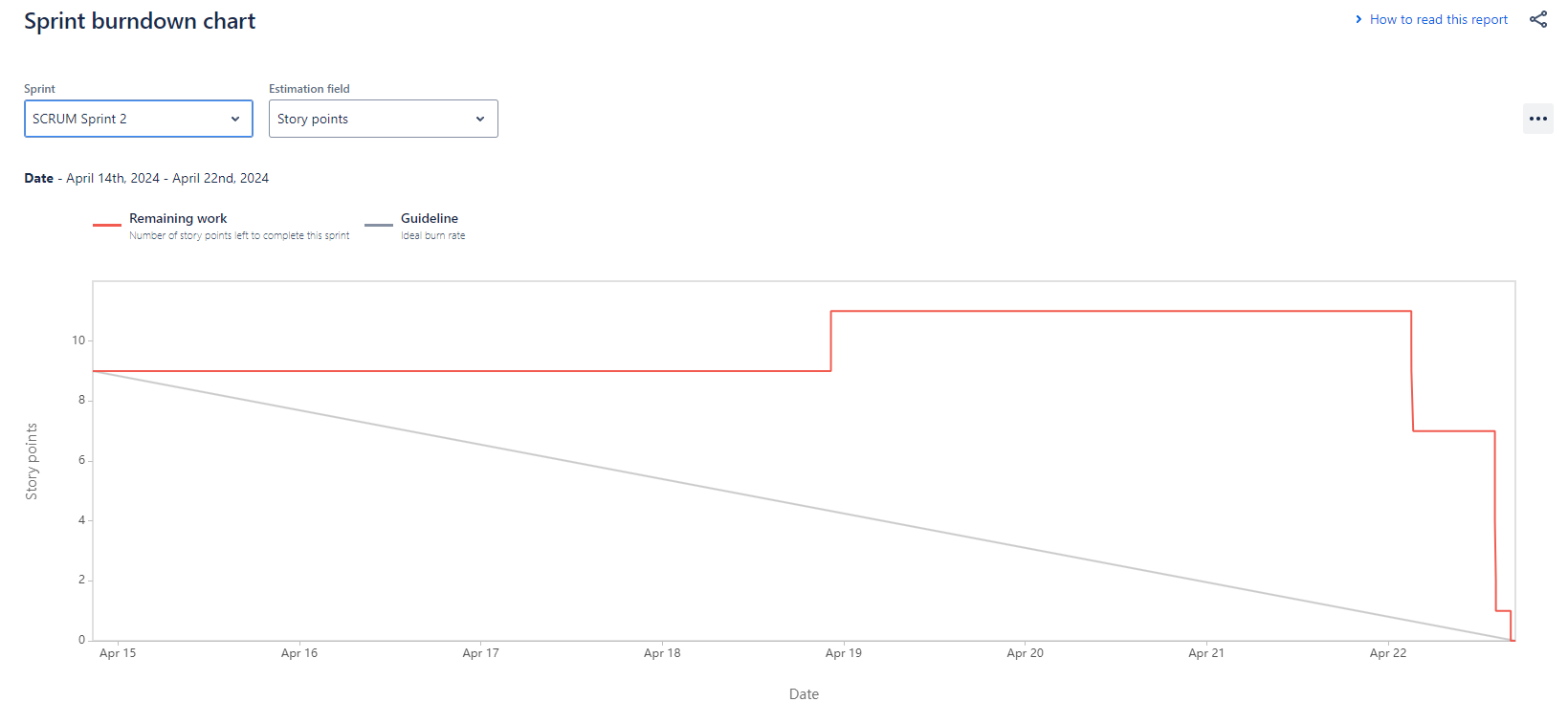
Sprint 2 Burnup Chart



Sprint 1 Burndown Chart

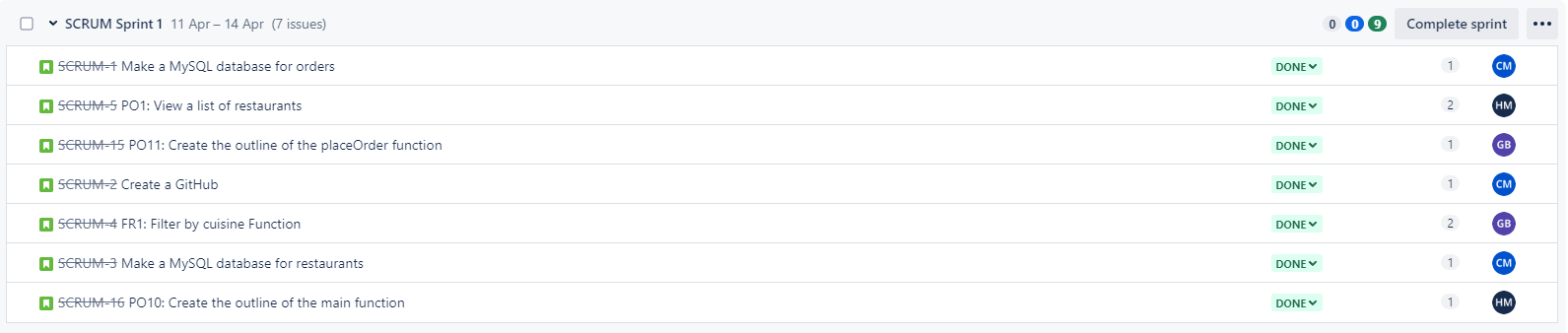


Sprint 2 Burndown Chart



**We believe the charts for Sprint 1 might not be perfect due to the team still learning how to use Jira.**

Backlogs







## **User Stories**

| **ID** | **Story Summary** | **Acceptance Criteria** | **Priority** | **Estimate** |
| --- | --- | --- | --- | --- |
| FR1 | As a user, I want to be able to filter between different restaurants according to cuisine so that I may select the type of food that I want. | user is able to select the culture of the food and a menu will be retrieved from database and displayed | High | 2 |
| PO1 | As a user, I want to be able to see a list of restaurants so that I may choose where I am ordering my food from. | user is able to view the array of available restaurants | High | 1 |
| PO2 | As a user, I want to be able to choose a restaurant from the available list | user is able to select a restaurant from the available list of restaurants | High | 1 |
| PO3 | As a user, I want to be able to see a menu so that I may make my food selections. | user is able to see a menu | High | 1 |
| PO4 | As a user, I want to be able to select one or more items from the menu and add them to my cart | user is able to select one or more menu items, and then the item(s) appear in their cart | High | 1 |
| PO5 | As a user, I want to be able to order food from multiple different restaurants in a single order | user is able to order items from more than one restaurant in a single order | High | 2 |
| PO6 | As a very picky eater, I want to be able to customize the food items I select. | Users will be allowed to input customization instructions. | High | 1 |
| PO7 | As a user, I want to be able to see my order and the total cost prior to payment. | user is able to see the cost breakdown of their order prior to being prompted for payment | High | 1 |
| PO8 | As a user, I want to be able to enter my payment information so that I can pay for my order. | user is allowed to enter card information | High | 1 |
| PO9 | As a user, I want to be able to see a confirmation of my payment. | user is able to see a confirmation message that is printed on the screen | High | 1 |
| DP1 | As a user who is very busy, I want to have the option to select delivery instead of pickup. | user is allowed to select a delivery option and it will ask for their address | High | 1 |
| CL1 | As somebody who is not fluent in English, I want to change the language so that I can read the app | user is able to select a language and the app will be display according to the selection | Low | 3 |
| RR1 | As somebody who only likes to eat at the best restaurants, I want to be able to see reviews. | user can see the average rating of the restaurant | Medium | 1 |
| RR2 | As a user, I want to be able to leave a 1–5 star review. | user can leave a review for restaurants | Medium | 1 |

| **[SCRUM-1]** [**Make a MySQL database for orders**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-1)Created: 10/Apr/24 Updated: 11/Apr/24 Resolved: 11/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00013: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Columns:   * Int order number * String customer name * String restaurants (concatenation of all restaurants that were ordered from) * String order (concatenation of all items) * String customizations * String delivery choice (pickup or delivery) * String customer address * Int customer zip code * String customer city * String customer state * String payment method (PayPal, VISA, etc.) * Int total price |
| --- |

| **[SCRUM-2]** [**Create a GitHub**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-2)Created: 10/Apr/24 Updated: 15/Apr/24 Resolved: 11/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00015: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| DoD: GitHub is created and accessible to all team members |
| --- |

| **[SCRUM-3]** [**Make a MySQL database for restaurants**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-3)Created: 10/Apr/24 Updated: 14/Apr/24 Resolved: 13/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00017: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| The database should have the following columns:   * Int Restaurant ID * String Name * String cuisine (Chinese, Mexican, etc.) * String address * Int zip code * String city * String state * String Menu Item #1 Name * Int Menu Item #1 Price * Repeat for 5 total items… * Float Rating (1-5)   Make sure to have at least 3 restaurants in the database |
| --- |

| **[SCRUM-4]** [**FR1: Filter by cuisine Function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-4)Created: 10/Apr/24 Updated: 15/Apr/24 Resolved: 14/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00016: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to filter between different restaurants according to cuisine so that I may select the type of food that I want.  DoD: User is able to filter different restaurants according to the cuisine |
| --- |

| **[SCRUM-5]** [**PO1: View a list of restaurants**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-5)Created: 10/Apr/24 Updated: 15/Apr/24 Resolved: 14/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Haricharan Mukthineni](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3Af2bdf885-0aed-4838-9a77-52ccfdfe2bd4) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00014: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to see a list of restaurants so that I may choose where I am ordering my food from.  DoD: Ticket is complete when a list of restaurants is printed to the screen |
| --- |

| **[SCRUM-6]** [**PO2: Restaurant choice function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-6)Created: 10/Apr/24 Updated: 22/Apr/24 Resolved: 22/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0009b: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 2 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to choose a restaurant from the available list  DoD: function successfully allows user to pick a restaurant and retrieve the menu |
| --- |

| **[SCRUM-7]** [**PO3: Menu choice or selection**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-7)Created: 10/Apr/24 Updated: 22/Apr/24 Resolved: 22/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0009j: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 2 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to see a menu so that I may make my food selections. |
| --- |

| **[SCRUM-8]** [**PO4: Food Selection Function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-8)Created: 10/Apr/24 Updated: 22/Apr/24 Resolved: 22/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Haricharan Mukthineni](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3Af2bdf885-0aed-4838-9a77-52ccfdfe2bd4) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0009f: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 2 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to select one or more items from the menu and add them to my cart  DoD: Ticket is complete once user is able to make one or more food selections. |
| --- |

| **[SCRUM-9]** [**PO5: Multiple Restaurant Selection**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-9)Created: 10/Apr/24 Updated: 22/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00058: |
| --- | --- |
| **Sprint:** |  |
| **Story point estimate:** | 0 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to order food from multiple different restaurants in a single order  DoD: Considered complete once user is able to go through multiple restaurant view menus, and add items. |
| --- |

| **[SCRUM-10]** [**PO6: Customization of order**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-10)Created: 10/Apr/24 Updated: 22/Apr/24 Resolved: 22/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0005d: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 2 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As a very picky eater, I want to be able to customize the food items I select.  DoD: Ticket is done when the user is allowed to customize food in their order. After the user has finished adding items to their cart, prompt the user to type out their customizations and hit enter. Save the customization to the order |
| --- |

| **[SCRUM-11]** [**Store an order to orders database**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-11)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0008f: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Create a function that stores an order to the orders database. This should not erase previous orders from the database.  DoD: An order can be successfully stored to the database |
| --- |

| **[SCRUM-12]** [**PO7: Display entire order**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-12)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0008b: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to see my order and the total cost prior to payment.  DoD: If the user is shown their entire, itemized order along with the prices and the total price prior to being asked for their payment then this ticket is complete. |
| --- |

| **[SCRUM-13]** [**PO8: Payment Information function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-13)Created: 11/Apr/24 Updated: 22/Apr/24 | |
| --- | --- |
| **Status:** | In Progress |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Haricharan Mukthineni](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3Af2bdf885-0aed-4838-9a77-52ccfdfe2bd4) |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0005e: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to enter my payment information so that I can pay for my order.  DoD: User is allowed to select payment method and add any payment information required. |
| --- |

| **[SCRUM-14]** [**PO9: User receives payment information**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-14)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0008n: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to see a confirmation of my payment.  DoD: This ticket is complete when a confirmation payment is displayed to the console after the user has input their payment info. |
| --- |

| **[SCRUM-15]** [**PO11: Create the outline of the placeOrder function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-15)Created: 11/Apr/24 Updated: 14/Apr/24 Resolved: 14/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) | **Assignee:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00014:i |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Create a barebones outline of the code.  DoD: Users should receive all the right prompts at all the right places regardless of whether the data exists or not. See HungerMate Order of Operations google doc for details. |
| --- |

| **[SCRUM-16]** [**PO10: Create the outline of the main function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-16)Created: 11/Apr/24 Updated: 14/Apr/24 Resolved: 13/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Gabriel Braun](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A74d07f85-55ca-427f-a5e8-55977d7dd781) | **Assignee:** | [Haricharan Mukthineni](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3Af2bdf885-0aed-4838-9a77-52ccfdfe2bd4) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0001a:d |
| --- | --- |
| **Sprint:** | SCRUM Sprint 1 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Create a barebones outline of the code.  DoD: Users should receive all the right prompts at all the right places regardless of whether the data exists or not. See HungerMate Order of Operations google doc for details. |
| --- |

| **[SCRUM-17]** [**DP1: Delivery/Pickup Functionality**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-17)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00089: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As a user who is very busy, I want to have the option to select delivery instead of pickup.  DoD: After the user has finalized their order, but before they pay, prompt them if they want to pickup or have their food delivered. Add a delivery fee if they have their food delivered. Modify the order as well. |
| --- |

| **[SCRUM-18]** [**RR1:See restaurant reviews**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-18)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00087: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 1 |

| **Description** |  |
| --- | --- |

| Story: As somebody who only likes to eat at the best restaurants, I want to be able to see reviews.  DoD: User is able to see reviews of different restaurants |
| --- |

| **[SCRUM-19]** [**CL1: Language change function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-19)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0007r: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Story: As somebody who is not fluent in English, I want to change the language so that I can read the app  DoD: This ticket is considered complete if the user is able to change the language that is written to the console |
| --- |

| **[SCRUM-20]** [**RR2: Restaurant rating function**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-20)Created: 11/Apr/24 Updated: 15/Apr/24 | |
| --- | --- |
| **Status:** | To Do |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | Unassigned |
| **Resolution:** | Unresolved | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i0007z: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 3 |
| **Story point estimate:** | 2 |

| **Description** |  |
| --- | --- |

| Story: As a user, I want to be able to leave a 1–5 star review.  DoD: This ticket is complete, if a user is able to leave a 1-5 star rating on 1-all of the restaurant(s) that they ordered from. |
| --- |

| **[SCRUM-21]** [**Update menu database**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-21)Created: 18/Apr/24 Updated: 22/Apr/24 Resolved: 22/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00097: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 2 |
| **Story point estimate:** | 1 |

| **[SCRUM-22]** [**Update restaurant table**](https://uhcl-team-dyhfcbnb.atlassian.net/browse/SCRUM-22)Created: 19/Apr/24 Updated: 22/Apr/24 Resolved: 22/Apr/24 | |
| --- | --- |
| **Status:** | Done |
| **Project:** | [Scrum](https://uhcl-team-dyhfcbnb.atlassian.net/secure/BrowseProject.jspa?id=10001) |
| **Components:** | None |
| **Affects versions:** | None |
| **Fix versions:** | None |

| **Type:** | Story | **Priority:** | Medium |
| --- | --- | --- | --- |
| **Reporter:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) | **Assignee:** | [Christopher Mendoza](https://uhcl-team-dyhfcbnb.atlassian.net/secure/ViewProfile.jspa?accountId=712020%3A18e1a5d3-52b1-4f2d-a95d-0ddb31e5c39c) |
| **Resolution:** | Done | **Votes:** | 0 |
| **Labels:** | None | | |
| **Remaining Estimate:** | Not Specified | | |
| **Time Spent:** | Not Specified | | |
| **Original estimate:** | Not Specified | | |

| **Rank:** | 0|i00093: |
| --- | --- |
| **Sprint:** | SCRUM Sprint 2 |
| **Story point estimate:** | 1 |

Generated at Mon Apr 22 21:24:11 UTC 2024 by Christopher Mendoza using Jira 1001.0.0-SNAPSHOT#100251-rev:54097c057d0bda11b22396420f3b76c8fd28eb13.

## **Code :**

#include <cppconn/driver.h>

#include <cppconn/exception.h>

#include <cppconn/statement.h>

#include <iostream>

#include <mysql\_connection.h>

#include <mysql\_driver.h>

#include <string>

#include <vector>

#include <format>

using namespace std;

// Show List of Restaurants

void queryRestaurants(sql::Connection \*con, const string&cuisine = "") {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

if (cuisine.empty()) {

res = stmt->executeQuery("SELECT idrestaurant,RestaurantName,RestaurantAddress FROM restaurant");

}

else {

res = stmt->executeQuery("SELECT idrestaurant,RestaurantName,RestaurantAddress FROM restaurant WHERE RestaurantCuisine = '" + cuisine + "'");

}

cout << "Restaurants:" << endl;

while (res->next()) {

cout << res->getInt("idrestaurant")<<":" << res->getString("RestaurantName") << "\n" << "Address: " << res->getString("RestaurantAddress") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

//Mcdonalds Menu Display

void DisplayMcDonaldsMenu(sql::Connection\* con) {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

res = stmt->executeQuery("SELECT idmcdonaldsmenu,item\_name,item\_price FROM mcdonaldsmenu");

cout << "Menu:" << endl;

while (res->next()) {

cout << res->getInt("idmcdonaldsmenu") << ":" << res->getString("item\_name") << "\n" << "Price: " << res->getDouble("item\_price") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

//Raising Canes Menu Display

void DisplayRaisingCanesMenu(sql::Connection\* con) {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

res = stmt->executeQuery("SELECT idraisingcanesmenu,item\_name,item\_price FROM raisingcanesmenu");

cout << "Menu:" << endl;

while (res->next()) {

cout << res->getInt("idraisingcanesmenu") << ":" << res->getString("item\_name") << "\n" << "Price: " << res->getDouble("item\_price") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

//Taco Bell Menu Display

void DisplayTacoBellMenu(sql::Connection\* con) {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

res = stmt->executeQuery("SELECT idtacobellmenu,item\_name,item\_price FROM tacobellmenu");

cout << "Menu:" << endl;

while (res->next()) {

cout << res->getInt("idtacobellmenu") << ":" << res->getString("item\_name") << "\n" << "Price: " << res->getDouble("item\_price") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

//Taco Cabana Menu Display

void DisplayTacoCabanaMenu(sql::Connection\* con) {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

res = stmt->executeQuery("SELECT idtacocabanamenu,item\_name,item\_price FROM tacocabanamenu");

cout << "Menu:" << endl;

while (res->next()) {

cout << res->getInt("idtacocabanamenu") << ":" << res->getString("item\_name") << "\n" << "Price: " << res->getDouble("item\_price") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

//panda express menu display

void DisplayPandaExpressMenu(sql::Connection\* con) {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

res = stmt->executeQuery("SELECT idpandaexpressmenu,item\_name,item\_price FROM pandaexpressmenu");

cout << "Menu:" << endl;

while (res->next()) {

cout << res->getInt("idpandaexpressmenu") << ":" << res->getString("item\_name") << "\n" << "Price: " << res->getDouble("item\_price") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

//Pei Wei menu display

void DisplayPeiWeiMenu(sql::Connection\* con) {

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

res = stmt->executeQuery("SELECT idpeiweimenu,item\_name,item\_price FROM peiweimenu");

cout << "Menu:" << endl;

while (res->next()) {

cout << res->getInt("idpeiweimenu") << ":" << res->getString("item\_name") << "\n" << "Price: " << res->getDouble("item\_price") << "\n" << endl;

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

}

void MainMenu() {

cout << "Welcome to Hunger Mate" << "\n";

cout << "1: Order" << "\n";

cout << "2: Exit" << "\n";

cout << "Enter Choice: ";

}

//filter resturants by type

void FilterRestaurantMenu()

{

cout << "Restaurant Types" << "\n";

cout << "1:American" << "\n";

cout << "2:Mexican" << "\n";

cout << "3:Chinese" << "\n";

cout << "4:View all" << "\n";

cout << "5:Go Back" << "\n";

cout << "Enter Choice: ";

}

//customize items function

std::vector<string> customizeOrder(std::vector<string> cart) {

std::vector<std::string> customizations;

std::string input;

for (const std::string& str : cart) {

cout << "Enter any Customizations for " << str << ":" << "\n";

std::getline(std::cin, input);

customizations.push\_back(input);

}

return customizations;

}

//adding item to cart function

void addItemToCart(sql::Connection\* con,std::vector<string>& cart, std::vector<string>foodItems, const std::string& tableName) {

foodItems.clear();

int choice;

std::cout << "Enter the number of the item you want to add to your cart: ";

std::cin >> choice;

std::cin.ignore(std::numeric\_limits<std::streamsize>::max(), '\n'); // Consume newline character

try {

sql::Statement\* stmt = con->createStatement();

sql::ResultSet\* res;

std::string query = "SELECT item\_name FROM " + tableName;

res = stmt->executeQuery(query);

while (res->next()) {

foodItems.push\_back(res->getString("item\_name"));

}

delete res;

delete stmt;

}

catch (sql::SQLException& e) {

cout << "SQL Exception: " << e.what() << endl;

}

// Check for valid input

if (choice > 0 && choice <= foodItems.size()) {

cart.push\_back(foodItems[choice - 1]);

std::cout << foodItems[choice - 1] << " added to your cart.\n";

}

else {

std::cout << "Invalid item number. Please try again.\n";

}

}

int main()

{ //MySQL Connector Setup

sql::Driver\* driver = get\_driver\_instance();

sql::Connection\* con = driver->connect("tcp://localhost:3306", "root", "root");

con->setSchema("hungermate");

int choice;

int restaurantChoice;

string cuisine;

//vectors storing the data that will be sent to database

std::vector<std::string> foodItems;

std::vector<string> customizations;

std::vector<string> restaurantChoices;

std::vector<string> cart;

do {

MainMenu();

cin >> choice;

cin.ignore(); // Clear the input buffer

switch (choice) {

case 1:

do {

FilterRestaurantMenu();

cin >> choice;

cin.ignore(); // Clear the input buffer

switch (choice) {

case 1:

cuisine = "American";

queryRestaurants(con, cuisine);

cout << "Enter number of restaurant to select it:";

cin >> restaurantChoice;

cin.ignore();

if (restaurantChoice ==1)

{

char continueShopping = 'y';

DisplayMcDonaldsMenu(con);

while (continueShopping == 'y' || continueShopping == 'Y') {

addItemToCart(con,cart, foodItems,"mcdonaldsmenu");

//customizeOrder(cart);

std::cout << "Do you want to add more items to your cart? (y/n): ";

std::cin >> continueShopping;

}

customizeOrder(cart);

cout << "Items in your cart:" << endl;

for (const string& item : cart) {

cout << item << endl;

}

}

if (restaurantChoice == 2)

{

char continueShopping = 'y';

DisplayRaisingCanesMenu(con);

while (continueShopping == 'y' || continueShopping == 'Y') {

addItemToCart(con, cart, foodItems, "raisingcanesmenu");

std::cout << "Do you want to add more items to your cart? (y/n): ";

std::cin >> continueShopping;

}

customizeOrder(cart);

cout << "Items in your cart:" << endl;

for (const string& item : cart) {

cout << item << endl;

}

}

break;

case 2:

cuisine = "Mexican";

queryRestaurants(con, cuisine);

cout << "Enter number of restaurant to select it:";

cin >> restaurantChoice;

cin.ignore();

if (restaurantChoice == 3)

{

char continueShopping = 'y';

DisplayTacoBellMenu(con);

while (continueShopping == 'y' || continueShopping == 'Y') {

addItemToCart(con, cart, foodItems, "tacobellmenu");

std::cout << "Do you want to add more items to your cart? (y/n): ";

std::cin >> continueShopping;

}

customizeOrder(cart);

cout << "Items in your cart:" << endl;

for (const string& item : cart) {

cout << item << endl;

}

}

if (restaurantChoice == 4)

{

char continueShopping = 'y';

DisplayTacoCabanaMenu(con);

while (continueShopping == 'y' || continueShopping == 'Y') {

addItemToCart(con, cart, foodItems, "tacocabanamenu");

std::cout << "Do you want to add more items to your cart? (y/n): ";

std::cin >> continueShopping;

}

customizeOrder(cart);

cout << "Items in your cart:" << endl;

for (const string& item : cart) {

cout << item << endl;

}

}

break;

case 3:

cuisine = "Chinese";

queryRestaurants(con, cuisine);

cout << "Enter number of restaurant to select it:";

cin >> restaurantChoice;

cin.ignore();

if (restaurantChoice == 5)

{

char continueShopping = 'y';

DisplayPandaExpressMenu(con);

while (continueShopping == 'y' || continueShopping == 'Y') {

addItemToCart(con, cart, foodItems, "pandaexpressmenu");

//customizeOrder(cart);

std::cout << "Do you want to add more items to your cart? (y/n): ";

std::cin >> continueShopping;

}

customizeOrder(cart);

cout << "Items in your cart:" << endl;

for (const string& item : cart) {

cout << item << endl;

}

}

if (restaurantChoice == 6)

{

char continueShopping = 'y';

DisplayPeiWeiMenu(con);

while (continueShopping == 'y' || continueShopping == 'Y') {

addItemToCart(con, cart, foodItems, "peiweimenu");

//customizeOrder(cart);

std::cout << "Do you want to add more items to your cart? (y/n): ";

std::cin >> continueShopping;

}

customizeOrder(cart);

cout << "Items in your cart:" << endl;

for (const string& item : cart) {

cout << item << endl;

}

}

break;

case 4:

cuisine = "";

queryRestaurants(con, cuisine);

break;

default:

cout << "Invalid choice. Please try again." << endl;

break;

}

} while (choice != 5);

break;

case 2:

cout << "Exiting program..." << endl;

break;

default:

cout << "Invalid choice. Please try again." << endl;

break;

}

} while (choice != 2);

delete con;

return 0;

}

## Resources

Creating a free online database:

<https://www.youtube.com/watch?v=TMGHOW8Hzvw&list=LL&index=1&ab_channel=AutomationStepbyStep>

C++ MySQL Connector Setup:

<https://www.youtube.com/watch?v=a_W4zt5sR1M&ab_channel=BoostMyTool>