Sprint 1 Materials

5B\_Bulbasaur

A logo with a number in a circle

Description automatically generated

Aidan Thomas

Brianna Bach

Brady O’Connor

Daniel Lam He

Zach Assad

# Retrospective Summary

In our sprint retrospective, the team recognized several aspects of the workflow that went well. Our task allocation and organization, primarily managed through Jira, significantly contributed to a smooth and efficient process. Tasks were clearly outlined in the product backlog, which provided each team member with a defined role, enhancing both individual focus and accountability. This organization represented an improvement from our previous projects, where roles and responsibilities may have been less clear. Additionally, the flexibility in adjusting task assignments as needed allowed us to adapt to challenges and achieve high-quality results.

However, we identified some areas for improvement. Although the product backlog was organized well, having a clearer, overarching vision for each sprint would help align our efforts towards a common goal. On Jira, we couldn’t see the assignees for each task which made it hard for us to know what tasks we were assigned. Additionally, while we appreciated the structure provided by regular meetings, some team members felt that the frequency could be reduced without impacting on our progress. Two of our backlog items were not completed because of their dependency on the backend. Since we were all inexperienced with Node.JS, our backend development took longer than expected.

For the next sprint, we’ll consider refining task assignments to make it easier to see which tasks are assigned to whom, potentially by creating a spreadsheet. We’ll also aim to provide more detailed descriptions for tasks as we get deeper into the project. This added clarity will support both our individual work and our collaborative efforts as we tackle increasingly complex requirements. The product backlog will be updated to reflect these adjustments and ensure tasks align with our long-term objectives.

# GitHub Release Link

# <https://github.com/CSCE331-Fall2024/project-3-team-5b/releases/tag/final-sprint-1>

# Product Backlog

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Status | Priority | Sprint Points |
| Executive Summary | Completed | High | 1 |
| System Diagram | Completed | High | 2 |
| Personas and User Stories | Completed | High | 3 |
| Interface Diagram | Completed | High | 4 |
| Team Roles and Qualifications | Completed | High | 0.5 |
| Development Methodology | Completed | High | 1 |
| Planned Scope | Completed | High | 1 |
| Task Breakdown and Scheduling | Completed | High | 4 |
| References | Completed | High | 0.2 |
| Product Backlog | Completed | High | 5 |
| Initial Product Burn-down chart | Completed | High | 0.5 |
| Set up the starter front end code and Github repository | Completed | High | 1 |
| Set up the starter back end code and Github repository | Completed | High | 1 |
| Design the Landing Page for Self-Service Order | Completed | High | 2 |
| Create the menu page | Completed | Medium | 5 |
| Create a Reusable Menu Item React Component | Completed | Medium | 2 |
| Fetch Menu Item Information from the Backend | Not Started | Medium | 2 |
| Add a Navigation Bar for Customers | Completed | Medium | 2 |
| Backend Should Process Menu Item Requests | Completed | Medium | 2 |
| Create a Reusable Button Component for the Manager Interface | Completed | Medium | 2 |
| Add Role-Specific Navigation Bar for Managers | Not Started | Medium | 2 |
| Create the manager landing page | Completed | Medium | 2 |
| Reuse the Customer Menu Item Component | Completed | Medium | 2 |
| Create a Cashier-Specific Layout | Completed | Medium | 2 |
| Add Role-Specific Navigation Bar for Cashiers | Completed | Medium | 2 |
| Implement Checkout Button (Non-Functional) | Completed | Medium | 2 |
| Deploy the website | Completed | Medium | 1 |
| Deploy the backend API | Completed | High | 1 |
| Implement Google OAuth for Authentication (Backend) | Not Started | High | 8 |
| Add Role-Based Redirect After Login (Backend) | Not Started | High | 2 |
| Add Role-Based Navigation Bar Visibility (Frontend) | Not Started | High | 2 |
| Implement Logout Functionality (Frontend) | Not Started | High | 2 |
| Implement Login Page (Frontend) | Not Started | High | 2 |
| Implement fetch request for the login (Frontend) | Not Started | High | 2 |
| Add Allergen and Nutritional Information to Menu Items | Not Started | Low | 2 |
| Display Allergen Alerts When Selecting Items | Not Started | Low | 2 |
| Create an API endpoint that returns the menu item components with its ingredients | Not Started | High | 3 |
| Implement Cart Functionality | Not Started | High | 5 |
| Complete ordering functionality | Not Started | Low | 2 |
| Create a Sales Reports Page | Not Started | High | 4 |
| Create Inventory Usage Report | Not Started | High | 4 |
| Integrate Weather API for Manager Landing Page | Not Started | High | 2 |
| Create Get Inventory API Endpoint | Not Started | High | 4 |
| Create an inventory page that displays all the information of the ingredients | Not Started | Medium | 2 |
| Create Add Inventory API Endpoint | Not Started | High | 2 |
| Create Edit Inventory API Endpoint | Not Started | High | 2 |
| Create Frontend Form for Adding/Editing ingredients | Not Started | Medium | 2 |
| Validation and error handling for Add/Edit Inventory | Not Started | High | 8 |
| Create Delete Inventory API Endpoint | Not Started | High | 3 |
| Create Frontend Button for Deleting Ingredients | Not Started | Low | 2 |
| Error Handling for Ingredient Delete Operation | Not Started | Low | 2 |
| Create Admin Page Layout | Not Started | Low | 2 |
| Add Navigation Link to Admin Page in Manager Landing Page | Not Started | High | 1 |
| Create Add Employee API Endpoint | Not Started | High | 4 |
| Create Edit Employee API Endpoint | Not Started | High | 4 |
| Add Buttons to Trigger Add or Edit Mode | Not Started | Low | 2 |
| Design Unified Edit/Add Employee Form | Not Started | Low | 2 |
| Handle Validation and Error Messaging in the Unified Form | Not Started | Low | 2 |
| Create a landing page for the menu page | Not Started | Low | 2 |
| Design the Menu Items Page Layout | Not Started | Low | 2 |
| Create Add Menu Item API Endpoint | Not Started | High | 4 |
| Create Edit Menu Item API Endpoint | Not Started | High | 4 |
| Create Delete Menu Item API Endpoint | Not Started | High | 4 |
| Implement Menu Item Add/Edit Form with Conditional Logic | Not Started | Low | 2 |
| Add Buttons to Navigate to Menu Item Components | Not Started | Low | 2 |
| Design the Menu Item Components Page Layout | Not Started | Low | 2 |
| Create Add Item Component API Endpoint | Not Started | High | 4 |
| Create Edit Item Component API Endpoint | Not Started | High | 4 |
| Create Delete Item Component API Endpoint | Not Started | High | 4 |
| Implement Item Component Add/Edit Form with Conditional Logic | Not Started | Low | 2 |
| Manage Ingredients within Components | Not Started | Low | 2 |
| Create a Reviews Section in the Menu Item Component | Not Started | Low | 2 |
| Set Up Backend API to Retrieve Reviews | Not Started | High | 5 |
| Create Reviews database table | Not Started | Low | 2 |
| Display Average Rating and Number of Reviews | Not Started | Low | 2 |
| Add a 'Leave a Review' Button to the Menu Item Component | Not Started | Low | 2 |
| Build a Review Submission Form | Not Started | Low | 2 |
| Implement Backend API for Review Submission | Not Started | High | 3 |
| Validate and Display Customer Reviews in Real Time | Not Started | Low | 2 |
| Create a Review Summary Report for Managers | Not Started | High | 5 |
| Display the Review summary report under Reports Page | Not Started | High | 4 |
| Implement Frontend Form Validation | Not Started | Low | 2 |
| Set Up Backend Error Handling | Not Started | Low | 2 |
| Integrate Frontend Error Boundaries | Not Started | Low | 2 |
| Automated Testing Suite | Not Started | Low | 2 |
| Create Order History Page | Not Started | High | 8 |
| Develop Backend API to Fetch Order History | Not Started | High | 4 |
| Design Order Summary Cards or Table | Not Started | High | 4 |
| Implement Search and Filter Functionality | Not Started | Low | 2 |
| Error Handling and Loading States | Not Started | Low | 2 |
| Integrate Google Translate API into the kiosk interface | Not Started | High | 4 |
| Create a dropdown menu on the main screen for language selection | Not Started | Low | 2 |
| Ensure all menu text, buttons, and labels dynamically update when a language is selected | Not Started | Low | 2 |
| Design icons and buttons with a larger-than-standard size | Not Started | Low | 2 |
| Use universally recognized symbols for actions | Not Started | Low | 2 |
| Develop a toggle switch on the main screen to enable high-contrast mode | Not Started | Low | 2 |
| Create a separate high-contrast color scheme | Not Started | Low | 2 |
| Ensure that all images, text, and icons adjust to the high-contrast setting | Not Started | Low | 2 |
| Add a text resizing feature on the main screen | Not Started | Low | 2 |
| Ensure that resized text adapts fluidly across all layouts | Not Started | Low | 2 |
| Test functionality with screen magnifiers | Not Started | Low | 2 |
| Provide a 'Text Size' button directly on each page | Not Started | Medium | 5 |
| Implement large, readable text settings | Not Started | Low | 2 |
| Add descriptive alt text to each image and icon | Not Started | Low | 2 |
| Verify screen reader compatibility | Not Started | Low | 2 |
| Integrate a text-to-speech (TTS) option | Not Started | Low | 2 |
| Add a 'Read Aloud' button next to each main section | Not Started | Low | 2 |
| Ensure universal icons and symbols are easily understood | Not Started | Low | 2 |
| Design a streamlined layout with minimal elements per page | Not Started | Low | 2 |
| Group related items for clarity | Not Started | Low | 2 |
| Design Display-Only Interface Layout for Multi-Display | Not Started | Low | 2 |
| Develop a Responsive Layout to Adapt to Large Screens | Not Started | Low | 2 |
| Implement Auto-Scroll or Slide Features | Not Started | Low | 2 |
| Configure Backend for Real-Time Updates | Not Started | Low | 2 |
| Ensure Consistency of Displayed Menu and Promotional Information | Not Started | Low | 2 |

Sprint Backlog

As the Panda Express Owner, I want a Complete Project Design, so that I can view your proposed work for the POS System Improvement

1. Executive Summary
   1. Assignees: Brady O’Connor
   2. Time estimate: 0.5 h
   3. Time Spent: 0.5 h
   4. Status: Complete
2. System Diagram
   1. Assignees: Zach Assad
   2. Time estimate: 1.5 h
   3. Time Spent: 1.5 h
   4. Status: Complete
3. Personas and User Stories
   1. Assignees: Daniel Lam He
   2. Time estimate: 1 h
   3. Time Spent: 2 h
   4. Status: Complete
4. Interface Diagram
   1. Assignees: Brianna, Daniel, and Zach
   2. Time estimate: 2 h
   3. Time Spent: 5 h
   4. Status: Complete
5. Team Roles and Qualifications
   1. Assignees: Brady O’Connor
   2. Time estimate:0.5 h
   3. Time Spent: 0.5 h
   4. Status: Complete
6. Development Methodology
   1. Assignees: Aidan Thomas
   2. Time estimate: 0.75 h
   3. Time Spent: 0.5 h
   4. Status: Complete
7. Planned Scope
   1. Assignees: Brianna Bach (Daniel Lam He took over the task)
   2. Time estimate: 0.5 h
   3. Time Spent: 1 h
   4. Status: Complete
8. Task Breakdown and Scheduling
   1. Assignees: Daniel Lam He
   2. Time estimate: 2 h
   3. Time Spent: 2 h
   4. Status: Complete
9. References
   1. Assignees: Whole team
   2. Time estimate: 0.2 h
   3. Time Spent: 0.2 h
   4. Status: Complete
10. Product Backlog
    1. Assignees: Daniel Lam He
    2. Time estimate: 5h
    3. Time Spent: 7h
    4. Status: Complete
11. Initial Product Burn-down chart
    1. Assignees: Daniel Lam He
    2. Time estimate: 1h
    3. Time Spent: 1h
    4. Status: Complete

As the developer, I want to start the code in a Github repository, so I can make updates to the code.

1. Set up the starter front end code and Github repository
   1. Assignees: Aidan Thomas
   2. Time estimate: 1 hour
   3. Time Spent: 20 minutes
   4. Status: Complete
2. Set up the starter back end code and Github repository
   1. Assignees: Aidan Thomas
   2. Time estimate: 1 hour
   3. Time Spent: 20 minutes
   4. Status: Complete
3. Deploy Backend
   1. Assignee: Daniel Lam He
   2. Time estimate: 1 hour
   3. Time spent: 20 minutes
   4. Status: Complete
4. Deploy Frontend
   1. Assignee: Daniel Lam He
   2. Time estimate: 1 hour
   3. Time spent: 20 minutes
   4. Status: Complete

As a customer, I would like to see the menu items, so I can place an order

1. Design the Landing Page for Self-Service Order
   1. Assignees: Daniel Lam He
   2. Time estimate: 1h
   3. Time Spent: 2h
   4. Status: Complete
2. Create the menu page
   1. Assignees: Zach, Brady
   2. Time estimate: 1.5 h
   3. Time Spent: 6.15 h
   4. Status: Complete
3. Create a Reusable Menu Item React Component
   1. Assignees: Brady
   2. Time estimate: 1 h
   3. Time Spent: 3h
   4. Status: Complete
4. Fetch Menu Item Information from the Backend
   1. Assignees: Aidan -> Daniel Lam
   2. Time estimate: 2 h
   3. Time Spent:
   4. Status: To Do
5. Add a Navigation Bar for Customers
   1. Assignees: Brianna
   2. Time estimate: 1h
   3. Time Spent: 2h
   4. Status: Complete
6. Backend Should Process Menu Item Requests
   1. Assignees: Aidan, Brianna
   2. Time estimate: 3h
   3. Time Spent: 6h
   4. Status: Complete

As a manager, I would like to see the landing page, so I can perform managerial tasks.

1. Create a Reusable Button Component for the Manager Interface
   1. Assignees: Brady
   2. Time estimate: 1h
   3. Time Spent: 3h
   4. Status: Completed
2. Add Role-Specific Navigation Bar for Managers
   1. Assignees: Brianna, Daniel -> Aidan Thomas
   2. Time estimate: 2 h
   3. Time Spent:
   4. Status: To Do
3. Create the manager landing page
   1. Assignees: Brady, Zach -> Aidan
   2. Time estimate: 1.5 h
   3. Time Spent: 2 h
   4. Status: Completed

As a cashier, I would like to see the take order page, so I can take customer orders.

1. Create a Cashier-Specific Layout
   1. Assignees: Zach, Brady, Aidan
   2. Time estimate: 5h
   3. Time Spent: 4 h
   4. Status: Complete
2. Add Role-Specific Navigation Bar for Cashiers
   1. Assignees: Brianna
   2. Time estimate: 1 h
   3. Time Spent: 1 h
   4. Status: Completed
3. Implement Checkout Button (Non-Functional)
   1. Assignees: Zach -> Brady
   2. Time estimate: 1.5 h
   3. Time Spent: 30 min
   4. Status: Completed

**Total number of completed user story points for each team member**

|  |  |
| --- | --- |
| Team Member | Points |
| Aidan Thomas | 12.45 |
| Brady O’Connor | 11.2 |
| Brianna Bach | 7.2 |
| Daniel Lam He | 14.7 |
| Zach Assad | 10.2 |

# Burn-down Charts

# Next Sprint’s SCRUM Meeting Schedule

First Scum Meeting: Friday, November 8th

Second Scrum Meeting: Monday, November 11th

Third Scrum Meeting: Friday, November 15th

Appendix 1: SCRUM Meeting Agendas and Minutes

SCRUM Meeting 1 for Project 3

Prepared by: Daniel Lam He

Meeting Date: 10/27/2024

## Meeting Attendees

1. Aidan Thomas
2. Brianna Bach
3. Brady O’Connor
4. Daniel Lam He
5. Zach Assad

## Meeting Agenda Items

* Updates to Project 3 Design and Management
* Missing items to complete for Project 3 Design and Management

## Status Update Since Last Meeting

Accomplishments:

* Several team members started their sprint tasks and some of them got completed

Tasks Completed:

|  |  |  |
| --- | --- | --- |
| **Task Description** | **Assigned to** | **Completed? (yes/no)** |
| System Diagram | Zach Assad | yes |
| Personas and User Stories | Daniel Lam He | yes |
| Product Backlog | Daniel Lam He | yes |
| Set up the starter front end code and Github repository | Aidan Thomas | yes |
| Set up the starter back end code and Github repository | Aidan Thomas | yes |

## Before The Next Meeting

Plans:

* Have completed some tasks for the MVP

Task Assignments:

|  |  |
| --- | --- |
| **Task Description** | **Assigned to** |
| Design the Landing Page for Self-Service Order | Daniel Lam He |
| Create the menu page | Zach, Brady |
| Create a Reusable Menu Item React Component | Brady |
| Fetch Menu Item Information from the Backend | Aidan |
| Add a Navigation Bar for Customers | Brianna |
| Backend Should Process Menu Item Requests | Aidan, Brianna |

## Minutes from Previous Meeting

N/A since this is the first scrum meeting.

SCRUM Meeting 2 for Project 3

Prepared by: Daniel Lam He

Meeting Date: 10/30/2024

## Meeting Attendees

1. Aidan Thomas
2. Brianna Bach
3. Brady O’Connor
4. Daniel Lam He
5. Zach Assad

## Meeting Agenda Items

* Updates of completed tasks for the github release
* Update actual time taken to complete task
* Work needed to be done before the end of next scrum meeting

## Status Update Since Last Meeting

Accomplishments:

* Skeleton completed for Login Page
* Menu board skeleton page
* Customer halfway done
* Connection to the database and backend base code

Tasks Completed:

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Description** | **Assigned to** | **Completed? (yes/no)** | **TT** |
| Design and Management Document | Team | Yes |  |
| Design the landing Page for Self-Service Order | Daniel | yes | 2 h |
| Manager Landing Page | Aidan | yes | 2 h |
| Create Reusable Menu Item React Component | Brady | yes | 3 h |
| Create Menu Page | Zach | yes | 1 h 45 min |
| Create Checkout button | Brady | yes | 30 min |
| Backend Should Process Menu Item Requests | Brianna | no | 2h |

## Before The Next Meeting

Plans:

* Complete the navigation bar for cashiers, customers, and managers
* Make menu items dynamic
* Complete the Backend endpoint that processes menu item requests
* Create Cashier specific layout
* Reuse Customer menu item component

Task Assignments:

|  |  |
| --- | --- |
| **Task Description** | **Assigned to** |
| Complete the navigation bar for cashiers, customers, and managers | Zach Assad, Brady O’Connor |
| Make menu items dynamic | Aidan, Zach |
| Complete the Backend endpoint that processes menu item requests | Brianna |
| Create Cashier specific layout | Aidan, Zach |
| Reuse Customer menu item component | Brady |
| Make Team Meeting presentation | Daniel Lam He |

## Minutes from Previous Meeting

In our previous meeting, we mentioned tasks completed so far in the sprint to update the sprint board. System Diagram, personas and user stories, product backlog, and creating a GitHub repository were the completed tasks. We also discussed tasks needed to be completed for Project 3 Design and Management and MVP in which the tasks were about the customer user interface.

SCRUM Meeting 3 for project 3

Prepared by: Daniel Lam He

Meeting Date: 11/04/2024

## Meeting Attendees

1. Aidan Thomas
2. Brianna Bach
3. Brady O’Connor
4. Daniel Lam He
5. Zach Assad

## Meeting Agenda Items

* Completed Items after the MVP
* Items to be completed before the end of the sprint

## Status Update Since Last Meeting

Accomplishments:

* Fully functional Menu page for the cashier and customer
* Backend and Frontend deployed

Tasks Completed:

|  |  |  |  |
| --- | --- | --- | --- |
| Task Description | Assigned to | Completed? (yes/no) | TT |
| Backend Should Process Menu Item Requests | Brianna | Yes | 4h |
| Add a Navigation Bar for Customers | Brianna | yes | 2h |
| Create a Cashier-Specific Layout | Aidan | Yes | 4h |
| Cashier layout buttonized (Menu Page) | Zach | Yes | 1h |
| Added pictures to the menu sidebar (menu Page) | Zach | Yes | 30 min |
| Functional menu buttons and cart (Menu Page) | Brady | Yes | 3 h |
| Deployed the backend and frontend | Daniel | Yes | 40 min |

## Before The Next Meeting

Plans:

* Finalize sprint 1 tasks

Task Assignments:

|  |  |
| --- | --- |
| **Task Description** | **Assigned to** |
| Navigation bar working for cashier | Brianna |
| Fetch Menu item Information from backend | Aidan |
| Add Role-Specific Navigation Bar for Managers | Daniel |

## Minutes from Previous Meeting

In the previous Scrum meeting we discussed tasks completed for the MVP and future tasks to be completed before the end of the sprint. We also thought of adding a new column for completed tasks called TT (Time Taken) as this is needed for the burn down chart and only the assignee knows how much time it took them to complete the task.