Sprint 2 Materials

Team 11

Samuel Bush
Dong Ha Cho
Blake Dejohn
Nicholas Petersilge
Shantanu Raghavan

Retrospective Summary

This sprint stood out as the most feature-intensive to date. As an overall theme, we focused on implementing most of the 3rd party API's that would be required by the project and making sure we plugged the gaps that were left behind from the last sprint (in terms of adding functionality to our application). Despite the complexity of the tasks tackled, we remain satisfied with our procedures. Almost all of our sprint goals were achieved, which demonstrates (we think) the robustness of our procedures. However, one issue that was still not solved by the end of this sprint concerned the failure of our inventory to update upon order placement. We think this was because of the task's low priority, leading us to defer the task to the next sprint. The value of peer evaluations became particularly evident during this sprint, providing the exact ways we could improve upon our team dynamic. In summary, while our recent sprint showcased remarkable progress and achievement, it also highlighted the importance of meticulous task prioritization and the need for continuous refinement in our processes. Looking ahead, we believe that we are right on task to completing the project and provide this report to prove so.

GitHub Release Link

https://github.com/csce-315-331-2024a/project-3-full-stack-agile-web-project-3-902-11/releases/tag/Sprint2

Product Backlog

Task Description	Priority	Dependencies	Estimated Time	Status	User Story Points
Design orders table	High	None	1-2 Days	Complete	2
Design food table	High	None	1-2 Days	Complete	2
Design ingredients table	High	None	1 Day	Complete	1
Join tables where necessary	High	Design of all 3 tables	1 Day	Complete	1
Populate the database	Medium	Database schema completed	2-5 Days	Complete	5
Design an API to interact with the database from frontend for cashier/customer	High	Database schema	2-3 Days	Complete	3
GET endpoint to fetch entire available menu	High	API design, database populated	1-2 Days	Complete	2
GET endpoint to fetch menu items by category	High	API design, database populated	1 Day	Complete	1
POST endpoint to submit a new order	High	API design, database schema	2-3 Days	Complete	3
PATCH endpoint to update inventory stock values when orders placed	High	API design, database schema, order endpoint	2 Days	In progress	2
Design an API for manager use	High	Database schema	2-3 Days	Complete	3

Allow user to manually update inventory stock values	Medium	API design for manager, database schema	1-2 Days	Complete	2
Fetch excess ingredient usage data given certain dates	Medium	Database schema, API design for manager	2-3 Days	Complete	3
Fetch ordering trend data from database	Medium	Database schema, API design for manager	2-3 Days	Complete	3
Fetch commonly ordered pairs data given certain dates	Medium	Database schema, API design for manager	2-3 Days	Complete	3
Update menu items	Medium	API design for manager, database schema, frontend	1-2 Days	Complete	2
Create new menu item with inputted data	Medium	API design for manager, database schema, frontend	1-2 Days	Complete	2
Delete menu item	Medium	API design for manager, database schema, frontend	1-2 Days	Complete	2
Fetch past orders data within certain dates	Medium	API design for manager, database schema	1-2 Days	Complete	2
Option to log into cashier and manager views	High	Authentication system	2-3 Days	Complete	3
Create button to log into cashier view	Medium	Authentication system, UI design	1 Day	Complete	1
Create button to log in to manager view	Medium	Authentication system, UI design	1 Day	Complete	1
Create button to display menu items in a separate tab	Medium	Frontend structure, menu data fetching	1-2 Days	Complete	2

Fetch menu items with sufficient inventory	Medium	API endpoint , database design	1-2 Days	Complete	2
Display menu items on the newly generated tab and do not create any interactivity	Medium	Tab creation, data fetching	1-2 Days	Complete	2
Display menu items in a static manner based on menu data fetched from the database	Medium	Data fetching	1 Day	Complete	1
Integrate translation API for accessibility	Medium	API selection, frontend structure	2-4 Days	Complete	4
Accommodate for WCAG 2.1 guidelines	High	Throughout frontend development	1-3 days per view	Not started	3
Create frontend design to allow users to choose which category to view	Medium	Menu data structure	1-2 Days	Complete	2
Fetch menu items grouped per category	Medium	API endpoint	1-2 Days	Complete	2
Update GUI based on category selection	Medium	Frontend logic, data fetching	1 Day	Complete	1
Create frontend design to allow user to add items to their cart	High	Frontend structure, cart logic	2-3 Days	Complete	3
Create frontend design to allow users to view their cart and make modifications	High	Frontend structure, cart logic	2-3 Days	Complete	3
Create frontend design to allow user to complete their order and check out their cart	High	Frontend structure, cart logic, API calls	2-3 Days	Complete	3
Call API to update the database with the new order	Medium	Frontend logic, API endpoint	1 Day	Complete	1

Call API to update ingredient stock	Medium	Frontend logic, API endpoint	1 Day	Complete	1
Display weather information from third-party API	Low	API selection	1-2 Days	Complete	2
Allow cashier to quickly add items to the current order	High	Menu item data, cart logic (frontend)	2-3 Days	Complete	3
Fetch menu items grouped per category	Medium	API endpoint for category-based fetching	1-2 Days	Complete	2
Display menu items in any given category	Medium	Frontend structure, data fetching	1-2 Days	Complete	2
Allow items to be added to an order with the option of choosing a quantity of a given item	High	Frontend structure, cart logic	2-3 Days	Complete	3
Create frontend design to allow cashier to check out the current order	High	Frontend structure, cart logic, API calls	2-3 Days	Complete	3
Call API to update database with the new order	Medium	Frontend logic, API endpoint	1 Day	Complete	1
Call API to update database's appropriate ingredient stock data	Medium	Frontend logic, API endpoint	1 Day	Complete	1
Create frontend interface for manager to modify menu items	Medium	Frontend form design, API for updating	2-4 Days	Complete	4
Create API calls from frontend to update menu item	Medium	API design for updating items	1-2 Days	Complete	2
Create frontend interface for manager to create a new menu item	Medium	Frontend form design, API for creation	2-4 Days	Complete	4
Create API calls from frontend to update database with inputted data	Medium	API design for item creation	1-2 Days	Complete	2

Create API calls from frontend to fetch inventory stock data	Medium	API design for inventory	1-2 Days	Complete	2
Use fetched data from API to update GUI to stock data	Medium	Frontend UI for stock display	1-2 Days	Complete	2
Create API calls from frontend to update inventory stock data	Medium	API design for inventory update	1-2 Days	Complete	2
Create frontend design for manager to view a visualized representation of excess ingredient usage data	Medium	Data visualization library, decision on chart types	2-3 Days	Complete	3
Create API calls from the frontend to fetch excess ingredient usage data	High	API design for excess ingredient data, backend logic to calculate	2-3 Days	Complete	3
Use fetched data from API to create a visualization of excess ingredient usage data	Medium	Data visualization library, frontend logic	2-3 Days	Complete	3
Create frontend design for manager to view a visualization of commonly ordered item pairs	Medium	Data visualization library, decision on chart types	2-3 Days	Complete	3
Create API calls from frontend to fetch commonly ordered pairs data	High	API design, backend logic to calculate pairs	2-3 Days	Complete	3
Use fetched data from API to create a visualization of commonly ordered item pairs data	Medium	Data visualization library, frontend logic	2-3 Days	Complete	3
Create frontend design for manager to view a visualization of past order data	Medium	Data visualization library, backend API for past orders	2-3 Days	Complete	3
Create API calls from frontend to fetch past order data	Medium	API design for past orders	1-2 Days	Complete	2
Use fetched data from API to create a visualization of past order data	Medium	Data visualization library, frontend logic	2-3 Days	Complete	3
Create frontend interface for manager to view all menu items	Medium	Frontend UI structure, backend API for menu items	2-3 Days	Complete	3

Create API calls from frontend to fetch menu items	Medium	API design for menu items	1-2 Days	Complete	2
Use fetched data from API to update GUI to show menu items	Medium	Frontend logic	1-2 Days	Complete	2
Create frontend interface for manager to modify menu items	Medium	Frontend form design, backend API for updating items	2-4 Days	Complete	4
Create API calls from frontend to update menu item	Medium	API design for item updates	1-2 Days	Complete	2
Create frontend interface for manager to create a new menu item	Medium	Frontend form design, backend API for item creation	2-4 Days	Complete	4
Create API calls from frontend to update database with inputted data	Medium	API design for item creation	1-2 Days	Complete	2
Create API calls from frontend to fetch inventory stock data	Medium	API design for inventory	1-2 Days	Complete	2
Use fetched data from API to update GUI to stock data	Medium	Frontend UI for stock display	1-2 Days	Complete	2
Create API calls from frontend to update inventory stock data	Medium	API design for inventory update	1-2 Days	Complete	2
Allow manager to switch to the menu page	Low	Frontend navigation	1 Day	Complete	1
Obtain OAuth2 credentials from the Google Developer Console	High	None	1-2 Days	Complete	2
Implement the backend authorization endpoint	High	OAuth2 library	2-3 Days	Complete	3
Implement token exchange and secure storage logic	High	OAuth2 library, authorization endpoint	2-3 Days	Complete	3

Implement user profile retrieval using Google's API	Medium	OAuth2 library, token exchange	1-2 Days	Complete	2
Design and implement the "Login with Google" UI element	Medium	UI design guidelines	1-2 Days	Complete	2
Implement frontend logic to initiate and handle the OAuth2 flow	Medium	OAuth2 library, "Login with Google" UI	2-3 Days	Complete	3
Implement token transmission to the backend	Medium	Frontend OAuth2 logic	1 Day	Complete	1
Create admin page	High	Login page	4-5 Days	Not Complete	5
Add order viewing component to employee view	High	Order viewing component, employee view	1-3 Day	Not Complete	2
Allow order viewing component to edit and delete orders	High	Order viewer	2-3 Days	Not Complete	3
Allow user to add/modify inventory items	Medium	Inventory API	2-3 Days	Not Complete	3
Implement functionality to view menu items in order of decreasing popularity	Medium	Menu API	1-2 Days	Not Complete	2
Create a log of order operations(deletion, adding)	Medium	Order modification API	2-4 Days	Not Complete	3
Add images to menu on customer page	Low	Customer page	1 Day	Not Complete	1
Make high contrast text/buttons	Medium	None	1 Day	Not Complete	1
Add alt-text and ARIA attributes	Low	Images	1-2 Day	Not Complete	2

Add a background pattern	Low	None	1 Day	Not Complete	1
Add magnifier API	Medium	Customer page	1 Day	Not Complete	2
Test manager page functions	High	Manager page	4-5 Days	Not Complete	5
Test customer page functions	High	Customer page	4-5 Days	Not Complete	5
Test cashier page functions	High	Cashier page	4-5 Days	Not Complete	5
Test admin page functions	High	Admin page	4-5 Days	Not Complete	5

Sprint Backlog

Task Description	**User Story Points**	**Assigned to**	**Actual Time Spent**	**Status**
Design frontend interfaces for item display	3	Dong Ha	2	Complete
Design frontend interfaces for adding items	3	Dong Ha	1	Complete

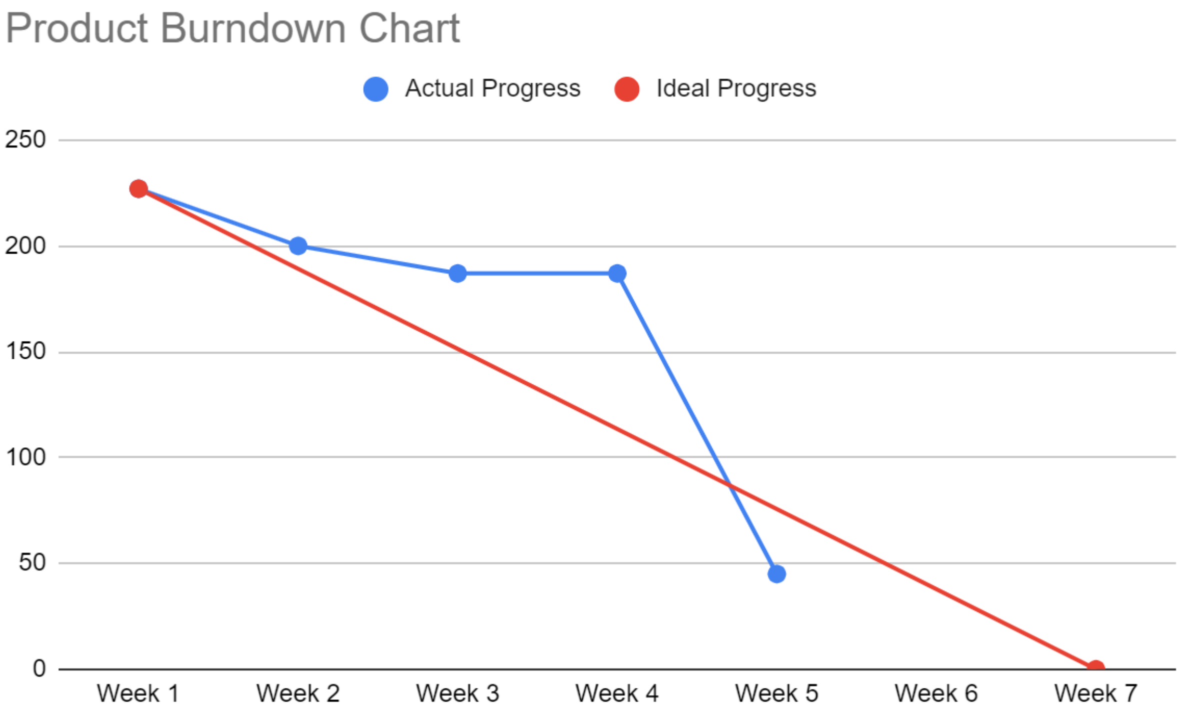
Design frontend interfaces to view and modify current order	3	Dong Ha	1	Complete
Add POST request to update database with new order	3	Nicholas	3	Complete
Obtain OAuth2 credentials from the Google Developer Console	2	Blake	3	Complete
Implement the backend authorization endpoint	3	Blake	2	Complete
Implement token exchange and secure storage logic	2	Blake	1	Complete
Implement user profile retrieval using Google's API	2	Blake	1	Complete
Design and implement the "Login with Google" UI element	3	Blake	2	Complete
Implement frontend logic to initiate and handle the OAuth2 flow	2	Blake	1	Complete

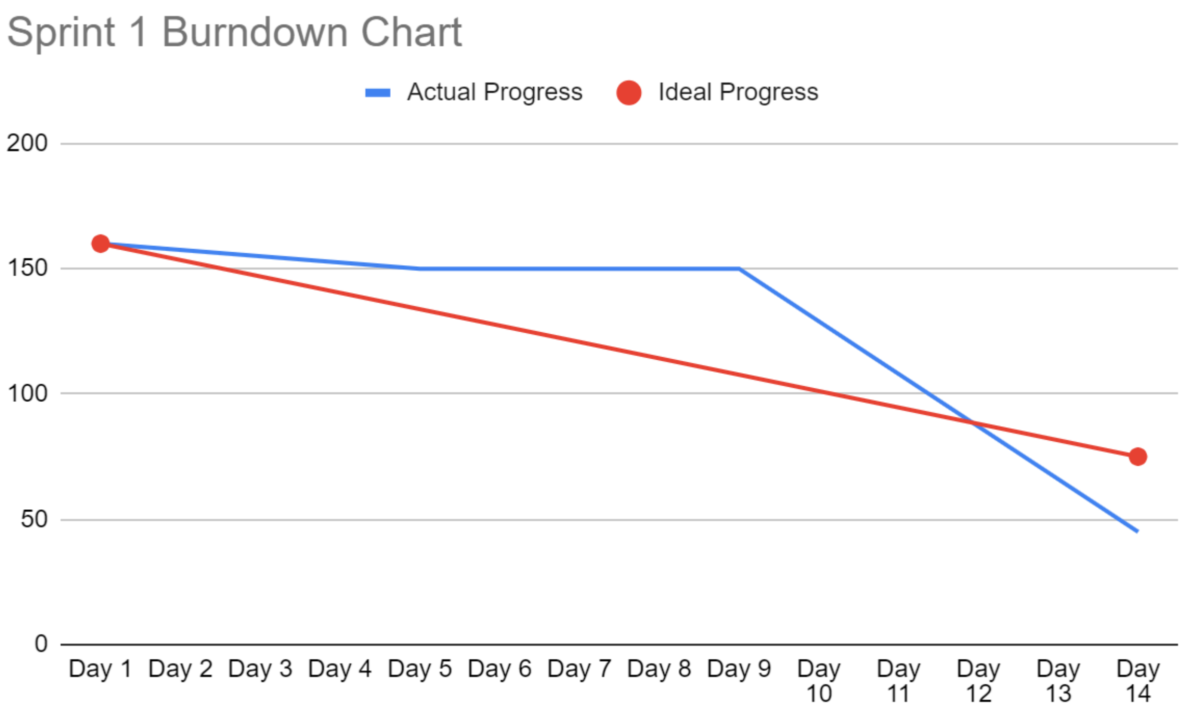
Implement Google OAuth for logging in to different profiles.	2	Blake	1	Complete
Implement weather API to display weather on customer self-checkout page.	2	Nicholas	2	Complete
Implement translation API to translate self-checkout page	2	Nicholas	3	Complete
Create frontend interface for manager to modify menu items	4	Shantanu	2	Complete
Create API calls from frontend to update menu item	2	Shantanu	3	Complete
Create frontend interface for manager to create a new menu item	4	Shantanu	1	Complete
Create API calls from frontend to update database with inputted data	2	Shantanu	1	Complete

Create API calls from frontend to fetch inventory stock data	2	Shantanu	2	Complete
Use fetched data from API to update GUI to stock data	2	Shantanu	2	Complete
Create API calls from frontend to update inventory stock data	2	Shantanu	2	Complete
Create frontend design for manager to view a visualized representation of excess ingredient usage data	3	Sam	2	Complete
Create API calls from the frontend to fetch excess ingredient usage data	3	Sam	2	Complete
Use fetched data from API to create a visualization of excess ingredient usage data	3	Sam	1	Complete
Create frontend design for manager to view a visualized representation of item sales report	2	Sam	1	Complete

Create API calls from the frontend to fetch sales report data	2	Sam	2	Complete
Use fetched data from API to create a visualization of sales report data	2	Sam	1	Complete
Create frontend design for manager to view a visualization of commonly ordered item pairs	1	Sam	1	Complete
Create API calls from frontend to fetch commonly ordered pairs data	2	Sam	3	Complete
Use fetched data from API to create a visualization of commonly ordered item pairs data	2	Sam	2	Complete
Develop the PATCH endpoint to update inventory stock after order placement.	1	Nicholas	1	In Progress
Develop the POST endpoint for submitting new orders.	2	Nicholas	3	Complete

Burn-down Charts





Next Sprint’s SCRUM Meeting Schedule

For this final sprint, we are hoping to meet every three days, with the first meeting on Wednesday 4/17. From then on, we will start programming early, making progress as soon as possible to ensure we have all features implemented for the final product.

Appendix 1: SCRUM Meeting Agendas and Minutes

SCRUM Meeting 1 for Revs POS System
Prepared by: Shantanu Raghavan
Meeting Date: 4/01/2024

Meeting Attendees

1. Samuel Bush
2. Dong Ha Cho
3. Blake Dejohn
4. Nicholas Petersilge
5. Shantanu Raghavan

Meeting Agenda Items

- New Sprint Expectations
- Allocation of Responsibilities

Status Update Since Last Meeting

Accomplishments:

- Sprint 1 complete
- Manager view, cashier view, customer view all exist now

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Adjust inventory stock	Shantanu	Yes
Display inventory stock	Shantanu	Yes

Before The Next Meeting

Plans:

- Make progress on tasks
- Communicate dependencies

Task Assignments:

Task Description	Assigned to
Design frontend interfaces for item display	Dong Ha
Design frontend interfaces for adding items	Dong Ha
Design frontend interfaces to view and modify current order	Dong Ha
Add POST request to update database with new order	Nicholas
Obtain OAuth2 credentials from the Google Developer Console	Blake
Implement the backend authorization endpoint	Blake
Implement token exchange and secure storage logic	Blake
Implement user profile retrieval using Google's API	Blake
Design and implement the "Login with Google" UI element	Blake
Implement frontend logic to initiate and handle the OAuth2 flow	Blake
Implement Google OAuth for logging in to different profiles.	Blake
Implement weather API to display weather on customer self-checkout page.	Nicholas
Implement translation API to translate self-checkout page	Nicholas
Create frontend interface for manager to modify menu items	Shantanu
Create API calls from frontend to update menu item	Shantanu
Create frontend interface for manager to create a new menu item	Shantanu
Create API calls from frontend to update database with inputted data	Shantanu
Create API calls from frontend to fetch inventory stock data	Shantanu
Use fetched data from API to update GUI to stock data	Shantanu
Create API calls from frontend to update inventory stock data	Shantanu
Create frontend design for manager to view a visualized representation of excess ingredient usage data	Sam
Create API calls from the frontend to fetch excess ingredient usage data	Sam
Use fetched data from API to create a visualization of excess ingredient	Sam

usage data	
Create frontend design for manager to view a visualized representation of item sales report	Sam
Create API calls from the frontend to fetch sales report data	Sam
Use fetched data from API to create a visualization of sales report data	Sam
Create frontend design for manager to view a visualization of commonly ordered item pairs	Sam
Create API calls from frontend to fetch commonly ordered pairs data	Sam
Use fetched data from API to create a visualization of commonly ordered item pairs data	Sam
Develop the PATCH endpoint to update inventory stock after order placement.	Nicholas
Develop the POST endpoint for submitting new orders.	Nicholas

Minutes from Previous Meeting

Since we had just finished sprint 1, we were discussing how we completed a considerable amount of work. We were realizing that we are a lot more efficient than we had anticipated, so we will assign more tasks this sprint. Some members will not be able to work for the first week due to other commitments, so we prepared to see most of our progress in the later portion of the sprint.

SCRUM Meeting 2 for Rev's Grill POS System
Prepared by:Shantanu Raghavan
Meeting Date:04/08/2024

Meeting Attendees

6. Samuel Bush
7. Dong Ha Cho
8. Blake Dejohn
9. Nicholas Petersilge
10. Shantanu Raghavan

Meeting Agenda Items

- Forecast for sprint completion
- Clarify division of labor

Status Update Since Last Meeting

Accomplishments:

- Some manager functionalities added
- Some login functionality added

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Create frontend interface for manager to modify menu items	Shantanu	yes
Create API calls from frontend to update menu item	Shantanu	yes
Create frontend interface for manager to create a new menu item	Shantanu	yes
Create API calls from frontend to update database with inputted data	Shantanu	yes
Obtain OAuth2 credentials from the Google Developer Console	Blake	yes

Before The Next Meeting

Plans:

- Make considerable progress, try and finish before next meeting
- Communicate roadblocks

Task Assignments:

Task Description	**Assigned to**
Design frontend interfaces for item display	Dong Ha
Design frontend interfaces for adding items	Dong Ha
Design frontend interfaces to view and modify current order	Dong Ha
Add POST request to update database with new order	Nicholas
Obtain OAuth2 credentials from the Google Developer Console	Blake
Implement the backend authorization endpoint	Blake
Implement token exchange and secure storage logic	Blake
Implement user profile retrieval using Google's API	Blake
Design and implement the "Login with Google" UI element	Blake
Implement frontend logic to initiate and handle the OAuth2 flow	Blake
Implement Google OAuth for logging in to different profiles.	Blake
Implement weather API to display weather on customer self-checkout page.	Nicholas
Implement translation API to translate self-checkout page	Nicholas
Create frontend interface for manager to modify menu items	Shantanu
Create API calls from frontend to update menu item	Shantanu
Create frontend interface for manager to create a new menu item	Shantanu
Create API calls from frontend to update database with inputted data	Shantanu
Create API calls from frontend to fetch inventory stock data	Shantanu

Use fetched data from API to update GUI to stock data	Shantanu
Create API calls from frontend to update inventory stock data	Shantanu
Create frontend design for manager to view a visualized representation of excess ingredient usage data	Sam
Create API calls from the frontend to fetch excess ingredient usage data	Sam
Use fetched data from API to create a visualization of excess ingredient usage data	Sam
Create frontend design for manager to view a visualized representation of item sales report	Sam
Create API calls from the frontend to fetch sales report data	Sam
Use fetched data from API to create a visualization of sales report data	Sam
Create frontend design for manager to view a visualization of commonly ordered item pairs	Sam
Create API calls from frontend to fetch commonly ordered pairs data	Sam
Use fetched data from API to create a visualization of commonly ordered item pairs data	Sam
Develop the PATCH endpoint to update inventory stock after order placement.	Nicholas
Develop the POST endpoint for submitting new orders.	Nicholas

Minutes from Previous Meeting

Many members of the group were busy and couldn't work until now, so it is understandable that we do not have much progress. We clarified task assignments, discussed feasibility, and concluded with a discussion about the final demo. Each member stated a different time that they could get their tasks done, but it seemed like everything would be done by Saturday for certain.

SCRUM Meeting 3 for Rev's Grill POS System
Prepared by:Shantanu Raghavan
Meeting Date:04/13/2024

Meeting Attendees

11. Sam Bush
12. Dong Ha Cho
13. Blake Dejohn
14. Nicholas Petersilge
15. Shantanu Raghavan

Meeting Agenda Items

- Discuss Accomplishments
- Discuss remaining tasks

Status Update Since Last Meeting

Accomplishments:

- cashier mostly done
- OAuth completed
- Manger functions almost done
- APIS done

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Design frontend interfaces for item display	Dong Ha	yes
Design frontend interfaces for adding items	Dong Ha	yes
Design frontend interfaces to view and modify current order	Dong Ha	yes
Add POST request to update database with new order	Dong Ha	yes
Implement the backend authorization endpoint	Blake	yes

Implement token exchange and secure storage logic	Blake	yes
Implement user profile retrieval using Google's API	Blake	yes
Design and implement the "Login with Google" UI element	Blake	yes
Implement frontend logic to initiate and handle the OAuth2 flow	Blake	yes
Implement Google OAuth for logging in to different profiles.	Blake	yes
Implement weather API to display weather on customer self-checkout page.	Nicholas	yes
Implement translation API to translate self-checkout page	Nicholas	yes
Create API calls from frontend to update database with inputted data	Shantanu	yes
Create API calls from frontend to fetch inventory stock data	Shantanu	yes
Use fetched data from API to update GUI to stock data	Shantanu	yes
Create API calls from frontend to update inventory stock data	Shantanu	yes
Create frontend design for manager to view a visualized representation of excess ingredient usage data	Sam	yes
Create API calls from the frontend to fetch excess ingredient usage data	Sam	yes
Use fetched data from API to create a visualization of excess ingredient usage data	Sam	yes
Create frontend design for manager to view a visualized representation of item sales report	Sam	yes
Create API calls from the frontend to fetch sales report data	Sam	yes
Use fetched data from API to create a	Sam	yes

visualization of sales report data		
Create frontend design for manager to view a visualization of commonly ordered item pairs	Sam	yes
Create API calls from frontend to fetch commonly ordered pairs data	Sam	yes
Use fetched data from API to create a visualization of commonly ordered item pairs data	Sam	yes
Develop the POST endpoint for submitting new orders.	Nicholas	yes

Before The Next Meeting

Plans:

- Finish up remaining tasks
- Merge everything well, be careful with pull requests

Task Assignments:

Task Description	**Assigned to**
PATCH endpoint to update inventory	Nicholas

Minutes from Previous Meeting

Since we were able to work a lot more this week, everything is mostly done. We were discussing how merge conflicts were arising frequently, and how to solve them. Pull requests were a good addition, and having someone view your work creates a higher standard of work quality. We did not have to discuss much else since we were mostly done.