Sprint 3 Materials

Team 11

Samuel Bush
Dong Ha Cho
Blake Dejohn
Nicholas Petersilge
Shantanu Raghavan

Retrospective Summary

This sprint was characterized as the testing, fixing, and documenting phase of our project. Because all our major functionality was completed in the previous sprint, we focused on making our project as robust as possible before the project deadline for this sprint. This was done through extensive testing, both with our frontend components and our backend logic (which includes API routes, database updates, etc.). Additionally, some new problems emerged, and we had to add tasks to our backlog, but they were not anything that could not be accomplished within the timeframe and our manpower. Some of these included new requirements, such as a kitchen page, but there were also new problems that appeared, which required all hands on deck to ensure they were solved by the deadline. To document our project, we had to determine a way to generate documentation for our code. After much deliberation, we landed on an option that worked well with Next.js, Nextra, and we were quick to document our code as much as necessary. Overall, we would say that this sprint was successful. We did not have as much time as other sprints because of the proximity to finals, however, we still managed to complete all the tasks that we wanted for this sprint. As a result, if we were to do this sprint over again, the only thing that we would change is the amount of time dedicated to completing the project.

GitHub Release Link

https://github.com/csce-315-331-2024a/project-3-full-stack-agile-web-project-3-902-11/releases/tag/Sprint3

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	Complete	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	Complete	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	Complete	5
Allow order viewing component to edit and delete orders	High	Order viewer	2-3 Days	Complete	3
Allow manager to add/modify inventory items	Medium	Inventory API	2-3 Days	Complete	3
Create a small log of kitchen orders	Medium	Order modification API	2-4 Days	Complete	3
Add images to menu on customer page	Low	Customer page	1 Day	Complete	1
Make high contrast text/buttons	Medium	None	1 Day	Complete	1

Add alt-text and ARIA attributes	Low	Images	1-2 Day	Complete	2
Add a background pattern	Low	None	1 Day	Complete	1
Test manager page functions	High	Manager page	4-5 Days	Complete	5
Test customer page functions	High	Customer page	4-5 Days	Complete	5
Test getUsers API for Admin Page	High	Admin page	4-5 Days	Complete	5
Test deleteUser API for Admin Page	High	Admin page	4-5 Days	Complete	5
Test updateUser API for Admin Page	High	Admin page	4-5 Days	Complete	5
Document Admin Page	Medium	Admin page	3 Days	Complete	3
Document Main Page	Medium	Main page	2 Days	Complete	2
Document Customer Page	Medium	Customer Page	3 Days	Complete	3
Document Cashier Page	Medium	Cashier Page	3 Days	Complete	3
Document Manager Page	Medium	Manager Page	3 Days	Complete	3
Clean up CSS on Main Page	Low	Main Page	½ Day	Complete	1

Speed up loading of the Manager Page	Medium	Manager Page	1 Day	Complete	1

What you did	Priority	Dependencies	Estimated Time	Status	User Story Points

Sprint Backlog

Task Description	**User Story Points**	**Assigned to**	**Actual Time Spent**	**Status**
Create a small log of kitchen orders	1	Sam Bush	1	Completed
Track status of an order in kitchen view	2	Sam Bush	2	Completed
Design and implement full kitchen view	3	Sam Bush	3	Completed
Update status of an order in kitchen view	2	Sam Bush	2	Completed

Add images to menu on customer page	2	Dong Ha Cho	2	Completed
Make high contrast text/buttons	3	Dong Ha Cho	2	Completed
Add alt-text and ARIA attributes	3	Nicholas Petersilge	3	Completed
Add a background pattern	2	Nicholas Petersilge	2	Completed
Test manager page functions	2	Shantanu Raghavan	2	Completed
Test customer page functions	2	Shantanu Raghavan	2	Completed
Test cashier page functions	2	Shantanu Raghavan	2	Completed
Test admin page functions	2	Blake Dejohn	2	Completed

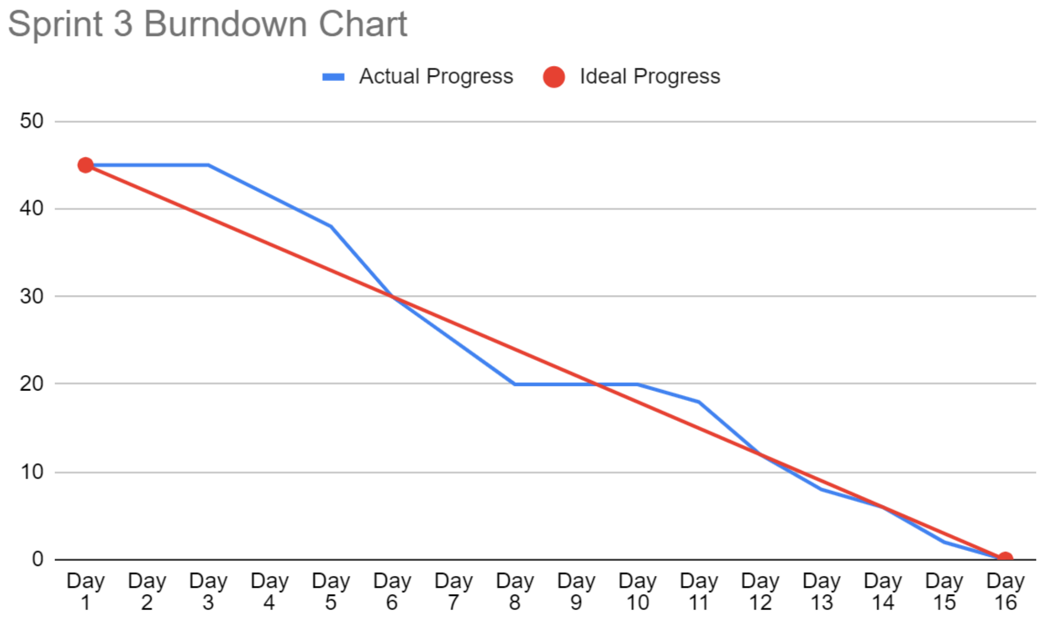
Test getUsers API for Admin Page	2	Blake Dejohn	2	Completed
Test deleteUser API for Admin Page	2	Blake Dejohn	2	Completed
Test updateUser API for Admin Page	2	Blake Dejohn	2	Completed
Document Admin Page	2	Blake Dejohn	2	Completed
Document Main Page	2	Nicholas Petersilge	2	Completed
Document Customer Page	2	Dong Ha Cho	2	Completed
Document Cashier Page	2	Dong Ha Cho	2	Completed
Document Kitchen Page	1	Sam Bush	1	Completed

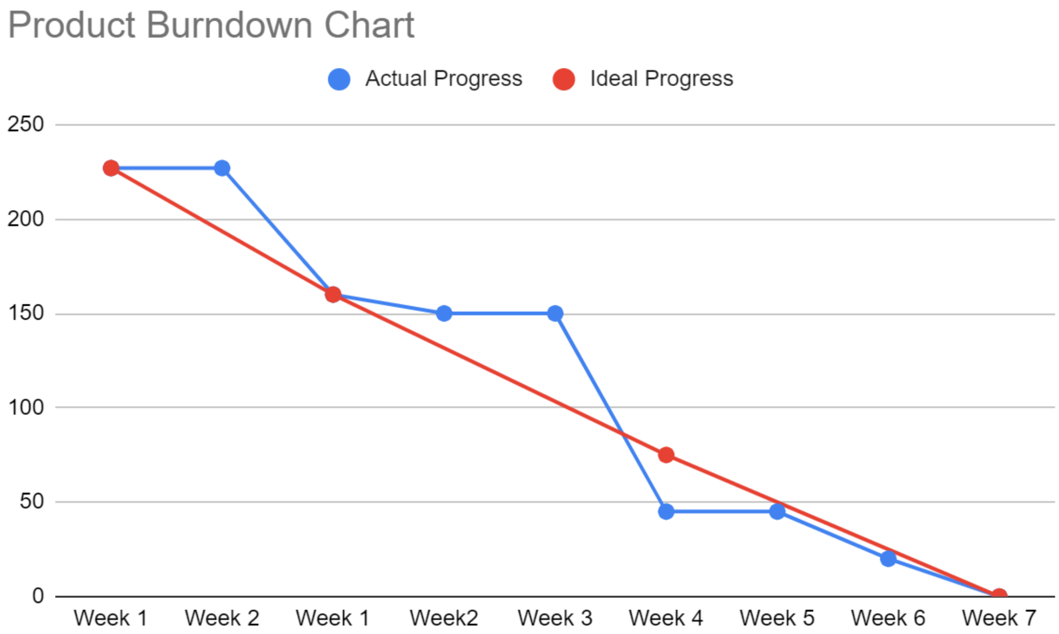
Document Manager Page	2	Shantanu Raghavan	2	Completed
Clean up CSS on Main Page	2	Nicholas Petersilge	2	Completed
Speed up loading of the Manager Page	1	Shantanu Raghavan	1	Completed

Summary Table

Team Member	**Completed User Story Points**	**Sprint #**
Shantanu Raghavan	9	Sprint 3
Sam Bush	10	Sprint 3
Nicholas Petersilge	9	Sprint 3
Blake Dejohn	10	Sprint 3
Dong Ha Cho	9	Sprint 3

Burn-down Charts





Appendix 1: SCRUM Meeting Agendas and Minutes

SCRUM Meeting 1 for Rev's Grill POS
Prepared by: Shantanu Raghavan
Meeting Date: 04/20/2024

Meeting Attendees

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

Meeting Agenda Items

- Discuss sprint expectations
- Task allocation

Status Update Since Last Meeting

Accomplishments:

- Created basic admin page
- Created basic kitchen page

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Created admin page	Blake	yes
Created kitchen page	Sam	yes

Before The Next Meeting

Plans:

- Develop functionality for kitchen and admin page
- Test application

Task Assignments:

Task Description	**Assigned to**
Create a small log of kitchen orders	Sam
Track status of an order in kitchen view	Sam
Design and implement full kitchen view	Sam
Update status of an order in kitchen view	Sam
Add images to menu on customer page	Dong Ha
Make high contrast text/buttons	Dong Ha
Add alt-text and ARIA attributes	Dong Ha
Add a background pattern	Shantanu
Test manager page functions	Shantanu
Test customer page functions	Shantanu
Test cashier page functions	Blake
Test admin page functions	Blake
Test getUsers API for Admin Page	Blake
Test deleteUser API for Admin Page	Blake
Test updateUser API for Admin Page	Blake
Document Admin Page	Blake
Document Main Page	Nicholas
Document Customer Page	Nicholas
Document Cashier Page	Dong Ha
Document Kitchen Page	Sam
Document Manager Page	Shantanu
Clean up CSS on Main Page	Shantanu
Speed up loading of the Manager Page	Shantanu

Minutes from Previous Meeting

We discussed the feasibility of completion for this project, given our schedules and task assignments. We reassigned tasks based on who was free, and based on our strengths. We discussed blockers, such as what testing framework to use, how to use them, and how to implement auto-generated documentation. We also had to modify the story points for certain tasks, just because they did not reflect their significance or time taken.

SCRUM Meeting 2 for Rev's Grill POS
Prepared by:Shantanu Raghavan
Meeting Date: 04/24/2024

Meeting Attendees

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

Meeting Agenda Items

- Discuss progress/ concerns about progress
- Debug testing/documentation

Status Update Since Last Meeting

Accomplishments:

- Started testing admin page
- Some kitchen page functionality

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Create a small log of kitchen orders	Sam	Yes
Track status of an order in kitchen view	Sam	Yes
Test getUsers API for Admin Page	Blake	Test getUsers API for Admin Page
Test deleteUser API for Admin Page	Blake	Test deleteUser API for Admin Page
Add a background pattern	Shantanu	Add a background pattern

Before The Next Meeting

Plans:

- Finish kitchen page
- Begin testing rigorously

Task Assignments:

Task Description	**Assigned to**
Design and implement full kitchen view	Sam
Update status of an order in kitchen view	Sam
Add images to menu on customer page	Dong Ha
Make high contrast text/buttons	Dong Ha
Add alt-text and ARIA attributes	Dong Ha
Test manager page functions	Shantanu
Test customer page functions	Shantanu
Test cashier page functions	Blake
Test admin page functions	Blake
Test updateUser API for Admin Page	Blake
Document Admin Page	Blake
Document Main Page	Nicholas
Document Customer Page	Nicholas
Document Cashier Page	Dong Ha
Document Kitchen Page	Sam
Document Manager Page	Shantanu
Clean up CSS on Main Page	Shantanu
Speed up loading of the Manager Page	Shantanu

We discussed how our progress was going for the sprint. Certain team members felt like they were not contributing, so they took on more tasks. We discussed blockers in testing, since the framework we had initially chosen was not working as intended. We pivoted to using vitest, and everyone was on the same page for testing, but documentation was still an issue.

SCRUM Meeting 3 for Rev's Grill POS
Prepared by: Shantanu Raghavan
Meeting Date: 04/27/2024

Meeting Attendees

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

Meeting Agenda Items

- Discuss final tasks
- Discuss final bugs

Status Update Since Last Meeting

Accomplishments:

- Tested the application
- Documentation written

Tasks Completed:

Task Description	**Assigned to**	**Completed? (yes/no)**
Design and implement full kitchen view	Sam	yes
Update status of an order in kitchen view	Sam	yes
Add images to menu on customer page	Dong Ha	yes
Make high contrast text/buttons	Dong Ha	yes
Add alt-text and ARIA attributes	Dong Ha	yes
Test manager page functions	Shantanu	yes
Test customer page functions	Shantanu	yes
Test cashier page functions	Blake	yes
Test admin page functions	Blake	yes

Test updateUser API for Admin Page	Blake	yes
Document Admin Page	Blake	yes
Document Main Page	Nicholas	yes
Document Customer Page	Nicholas	yes
Document Cashier Page	Dong Ha	yes
Document Kitchen Page	Sam	yes
Document Manager Page	Shantanu	yes
Clean up CSS on Main Page	Shantanu	yes
Speed up loading of the Manager Page	Shantanu	yes

Minutes from Previous Meeting

We discussed some bugs that arose when we were each working on separate tasks. One example was the order viewing component had some functionality impacted by another user's changes, but we did not notice until the scrum meeting. This arose because we were discussing the completeness of our project, and noticed it when doing a final run through. We discussed what tasks were remaining, and all hands were on deck for completing those tasks.