

EE/CprE/SE 492 STATUS REPORT 04

02/26/25 - 03/11/26

Group number: SDMAY26_44

Project title: Culinary Command

Client &/Advisor: Matayas Durr's Father (Restaurant Owner) / Simanta Mitra

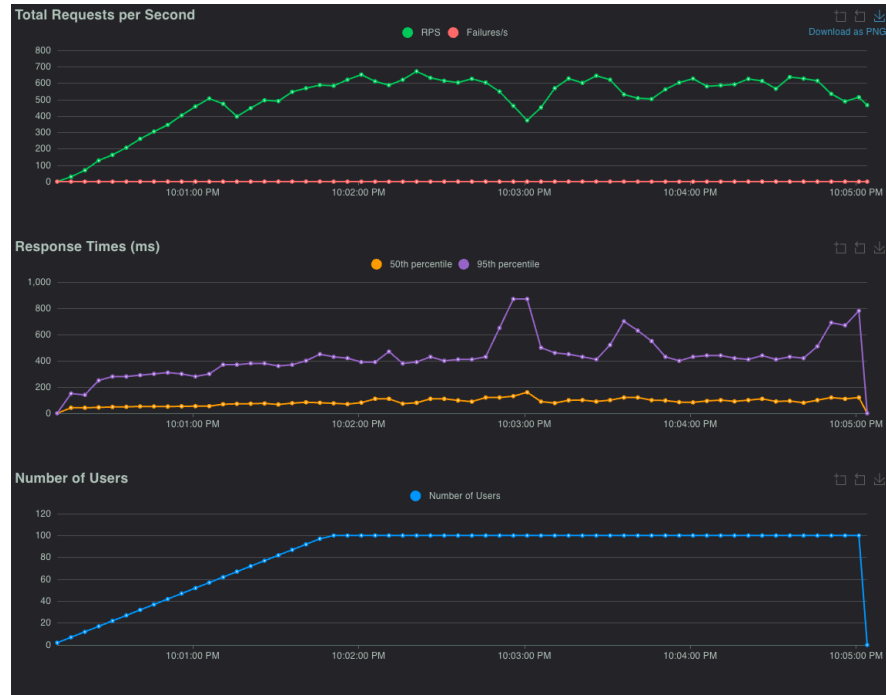
Team Members/Role: Anthony Phan, Wyatt Hunter, Ryan Rockey, Kevin Tran, Matayas Durr

o Weekly Summary

This week, the team focused on improving the functionality, scalability, and usability of the Culinary Command application. Several new features were implemented across both the frontend and backend, including recipe creation logic, task management improvements, and a feedback system for in-person demos. Infrastructure was also upgraded to support scalability by introducing an AWS EC2 Auto-scaling group capable of adjusting instance count based on traffic. UI work continued with finalized Figma designs being integrated into production pages, and automated testing efforts began using Playwright to validate the authentication and application flows. The team made strong progress toward improving system performance, user experience, and development infrastructure.

o Past week accomplishments

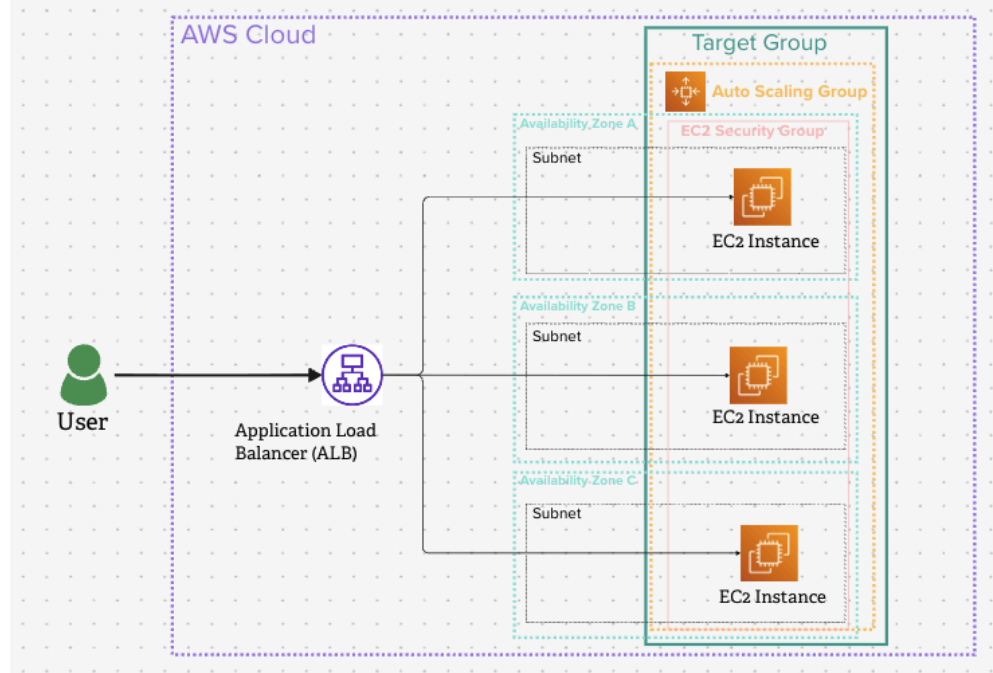
- o Anthony Phan
 - o Completed Figma Designs along with renders into prod
- o Kevin Tran
 - o Implemented recipe creation logic along with migrations.
 - o Performed load testing to determine what infrastructure would be necessary for Culinary Command:



Complete results can be [found here](#).

- Switched infrastructure to be scalable. The team is now using an EC2 Auto Scaling Group (ASG) to detect and distribute the traffic.
 - New EC2 instances will spin up if the request per second (RPS) rate hits a certain threshold.
 - The ASG is configured to keep a *minimum* of 1 EC2 instance up at all times and a *maximum* of 3 instances.
 - If the RPS rate is under the threshold for 15 minutes, then the ASG will *scale in*, spinning down the unnecessary EC2 instances.

Culinary Command Scalability Architecture



- The **Application Load Balancer (ALB)** is responsible for redirecting traffic to an EC2 Instance.
- Each **EC2 Instance** hosts a version of the Culinary Command app. We ensure each **EC2 Instance** hosts the same app by using a *launch template*.
- We can define a *launch template* that our **Auto Scaling Group** references as a blueprint for each **EC2 instance**:
- An **Auto Scaling Group** determines the configuration of the deployed EC2 instances. *For example, we can say we want a minimum of 2 EC2 instances deployed at all times and a maximum of 4 EC2 instances deployed at once.*
- A **Target Group** tells the **Application Load Balancer** where to forward traffic.
- The **EC2 Security Group** will serve as a basic firewall. *e.g. allow inbound traffic from SSH (22), HTTP (80), and HTTPS (443) and allow outbound traffic to anywhere*

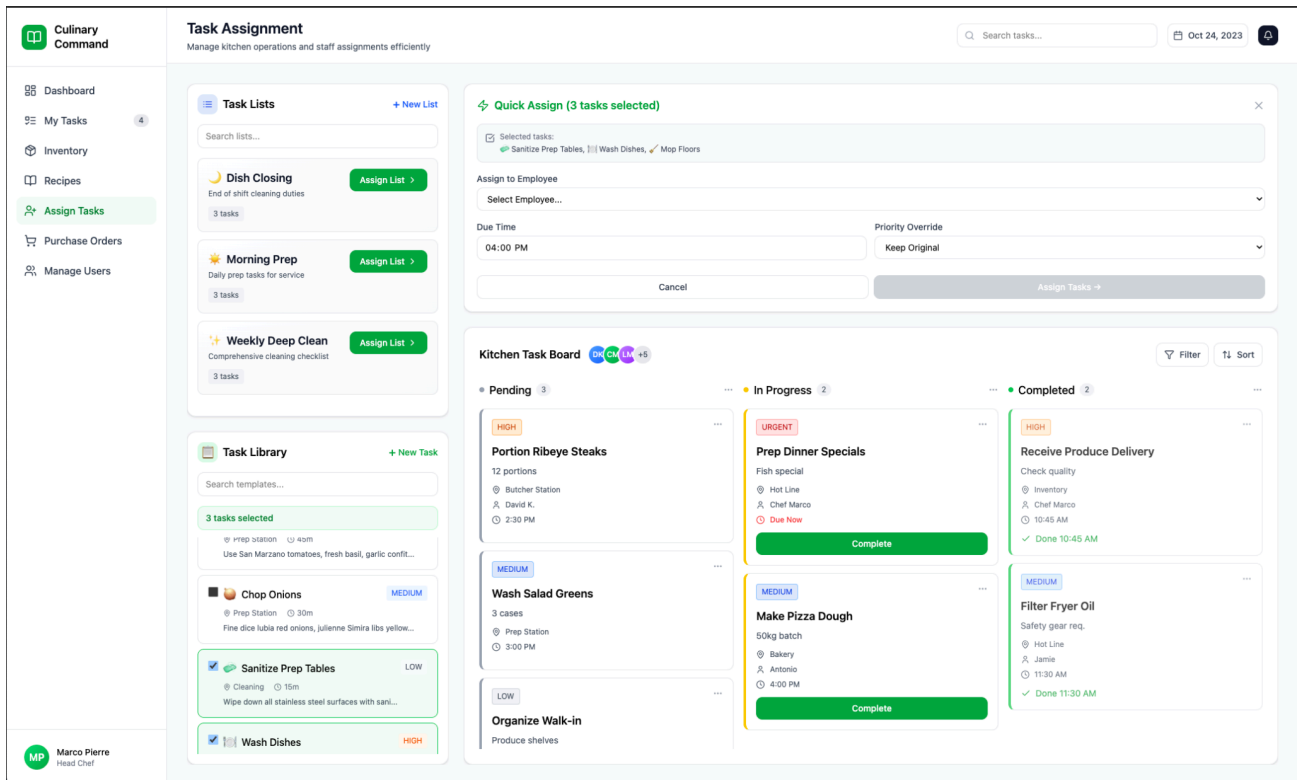
- Ryan Rockey
 - Created MarginEdge API testing tool, equipped with a JSON view and card view
 - Created feedback button for the in-person demos

Feedback popup

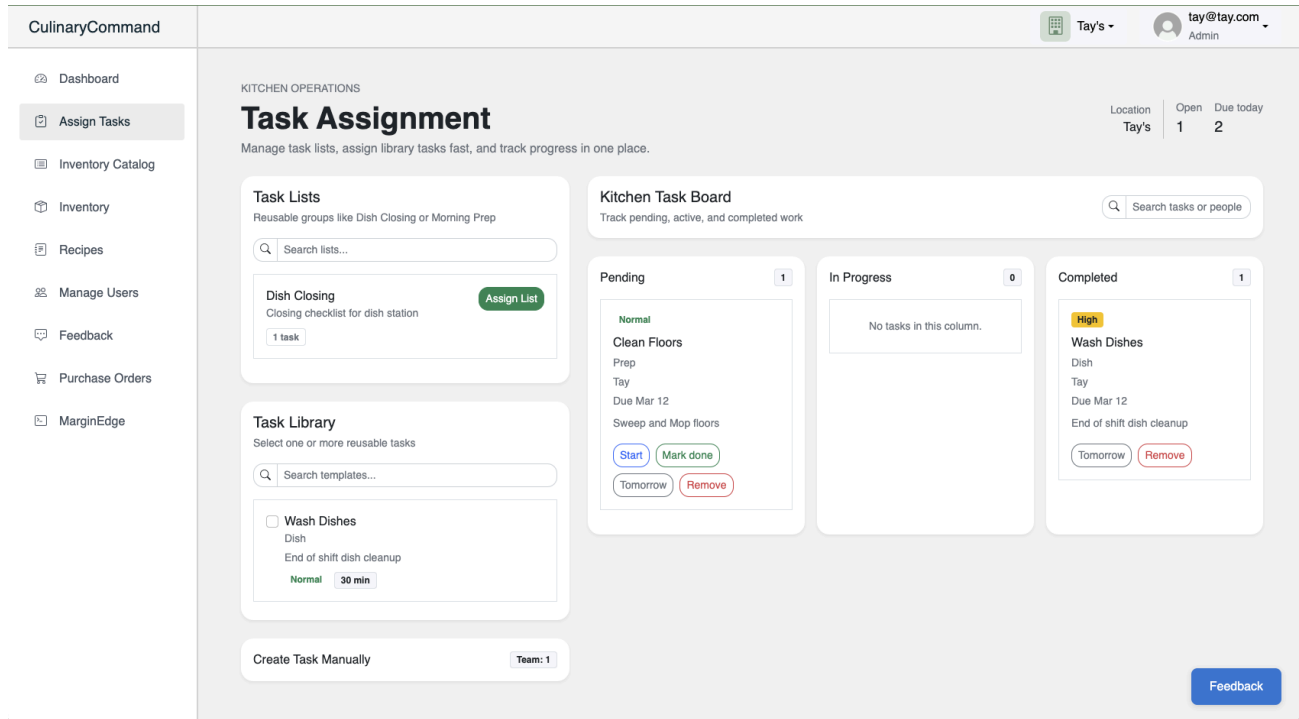
- Selection for Bug, General Feedback, or Feature request
- Description box for feedback
- Screenshot attachment

Feedback view page at `/feedback`

- View for all fields
- Screenshot viewer (if applicable)
- Delete button
- Add auto-refresh to task page
- Matayas Durr
 - Redesigned the Task Assignment page to improve manager usability based on demo feedback.
 - Implemented a task library system that allows managers to select reusable tasks.
 - Made Task Lists (groups of tasks) so multiple tasks can be assigned at once with one click.
 - Added database support for TaskTemplates, TaskLists, and TaskListItems.
 - Integrated the new task system with the existing task assignment workflow.
 - Updated the Task Assignment page with Bootstrap based of the Figma design.
- Figma Design for Task Assignment page



- Updated Task Assignment page



- Wyatt Hunter
 - I worked on getting our Culinary Command domain connected to the application email system, setting it up through Cloudflare, and making sure it works with our AWS backend so the app can send things like user invites. I also spent time reworking the parts of the login flow, including how users move from the landing page to the login from invites, and making sure everything works with the Cognito (Sign-in AWS system).
Along with that, I started working more with Playwright to begin building automated tests for the site. First, getting the Playwright tests to get through the sign-in system and getting into the app.
- **Pending issues** (If applicable: Were there any unexpected complications? Please elaborate.)
 - Anthony Phan:
 - Authentication Error after attempting to login through Cognito, was able to fix it by adding statement lines/fixes to [Program.cs](#) file
 - Kevin Tran:
 - No issues, however, to close out the Recipe feature, the Recipe view page needs to be designed.
 - Ryan Rockey:

- The current deployment of my tasks page update PR is not passing our GitHub actions pipeline.
 - I think it may have to do with the new transition to the AWS EC2 instances
- Matayas Durr:
 - None for what I am working on.
- Wyatt Hunter:
 - None

○ **Plans for the upcoming week** *(Please describe duties for the upcoming week for each member. What is(are) the task(s)? Who will contribute to it? Be as concise as possible.)*

- Anthony Phan:
 - Continue to add need features, slowly add in animations and updated new UI components, for better interactability.
- Kevin Tran:
 - Look into implementing the AI engine pipeline and also integrating Margin Edge data within the app.
- Ryan Rockey:
 - Work on AI pipeline or incorporating AI task features
 - Add new features as needed
- Matayas Durr:
 - Refine the task management system
 - Create the task library functionality
 - Add additional task management features to the system
 - Improve UI and layout
 - Add
- Wyatt Hunter:
 - For the next phase, I plan on working more with testing, especially getting into the app and adding data so it can go down paths throughout the Culinary Command application. This will really help improve our system and check that everything flows as it should.

○ **Individual contributions**

<u>NAME</u>	<u>Individual Contributions</u> <i>(Quick list of contributions. This should be short.)</i>	<u>Hours this week</u>	<u>HOURS cumulative</u>
Anthony Phan	Page Renderings	12	31
Kevin Tran	Recipe logic, load testing, EC2 autoscaling	22	50

	architecture		
Matayas Durr	Task assignment redesign, task library system, backend DB implementation	12	30
Ryan Rockey	MarginEdge API testing tool, feedback button, Auto-refresh to task page	12	38
Wyatt Hunter	Email System verification and setup, and playwright testing.	12	44

o **Summary of advisor meeting(s)**

No meetings this week