

Tasks Completed:

- Krisanahari Siva (s3941378):
 - Renamed microservices to correct formatting.
 - product-service
 - user-service
 - delivery-service
 - Created additional fail and passed repository tests in Product Service microservice.
 - Implemented new feature called Detailed Product Information – feature gives additional stores, prices and rating to the front end
 - Implemented tests for service and controller for the new feature.
 - Created tests in User Service for:
 - Create Cart
 - Read Cart
 - Update Cart
 - Delete Cart
 - Implemented all the cart features.
 - Created tests in Product Service for:
 - Get Review
 - Set Review
 - Implemented all the review features.
 - Set up AWS Lab.
 - Created three ECR repositories for the three microservices.
 - Build docker images for each microservices.
 - Pushed images to each ECR repository accordingly.
 - Created a new docker compose using AWS ECR links. (Backend is complete).
 - Created docker images for front end.
 - Pushed image to front end ECR.
 - Updated docker compose file for front end to use AWS ECR.
- Lawrence Dawood (s3668011):
 - Database Migration from H2 to MySQL:
 - Data Model Analysis: Review the existing H2 data model and determine any modifications needed for MySQL.
 - Migration Scripts: Write scripts or utilize migration tools to transfer data from H2 to MySQL.
 - Update Configuration: Modify application configurations to connect to the MySQL database instead of H2.
 - Dependency Management: Ensure the application has the necessary JDBC drivers or dependencies for MySQL.
 - Data Validation: After migration, validate that all data in MySQL matches the original data in H2.

- Performance Testing: Test the application's performance on MySQL to ensure it meets or exceeds prior performance on H2.
- Dockerization and Backend Deployment:
 - Dockerfile Creation: Write a Dockerfile for the backend application ensuring it has the necessary environment and configurations.
 - Database Container: Create a separate Dockerfile for the MySQL database or use an official MySQL Docker image.
 - Docker Compose: Use Docker Compose to manage multi-container applications, ensuring the backend and the MySQL database work in tandem.
 - Volume Management: Configure Docker volumes to persist MySQL data, ensuring data isn't lost when the container is stopped or removed.
 - Testing in Containers: Test the application running inside the Docker containers to ensure all functionality remains intact.
- Lance Belen (s3944846):
 - Tests for User Service:
 - Sign-up
 - User Details
 - Backend implementation of User Service
 - Sign-up
 - User Details
 - Notifications
 - Base code for delivery service
 - Delivery service model
 - Delivery service database schema
 - Create Schema
 - Insert Schema
 - Tests for Delivery Service
 - set delivery
 - get delivery details by delivery id
 - get delivery details by username
 - Backend implementation of Delivery Service
 - set delivery
 - get delivery details by delivery id
 - get delivery details by username
 - User service model update
 - added notifications and card payment info
 - added address
 - Backend implementation of User Service feature - Sign-up update for address addition
 - Tests for User Service feature - Sign-up update for address addition

- Laura Gatt (s3945256):
 - Sign up page:
 - All the fields: firstname, lastname, username, email, password, notifications, address, card name, card number, card expiration, card cvv
 - Made link to sign in after user signed up
 - User Details page:
 - Reveals the currently signed in user's details
 - Hides card details until clicked that it's wanted to be viewed
 - All documentation for milestone 3 submission
 - SRS document updating
 - SRS document finalisation
 - Delta report
 - Meeting minutes
- Saksham Jain (s3923854):
 - Product Details Webpage UI and Functionality
 - Cart Webpage UI and Functionality
 - Checkout Webpage UI and Functionality
 - Order List Webpage UI and Functionality
 - Frontend AWS Images
 - Frontend Docker-Compose
 - Refinement of entire website
 - Connected backend images to frontend
 - Product List Page:
 - Added functional Add to Cart
 - Documentation for milestone 3 submission
 - Project Report