Average at Best

Made for the Average Runner

A Website for athletic clothing made for runners.

COP 4331 Team 7 Large Project

Creators

Katelyn Campbell

Front-end/Mobile Dev

Created About & Home Page

Noah Fuhrman

Project Manager/Front-end

MERN Set Up & Created Products Page

Randy Nguyen

Front-end

Created Navigation Bar & About Page

Nick Piazza

Front-end

Created Cart Page

Creators

Santos Solanet

API

Created API Portion

Ashton Becker

Database

Created Database

Jacob Adams

Mobile Developer

Created Mobile App

Joshua Orlian

Founder

Created the Brand

Technologies Used

DigitalOcean

GoDaddy

Android Studio

Postman

Figma

VSCode

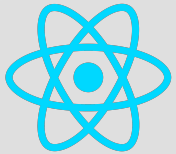
SwaggerHub

Jira

Draw.io

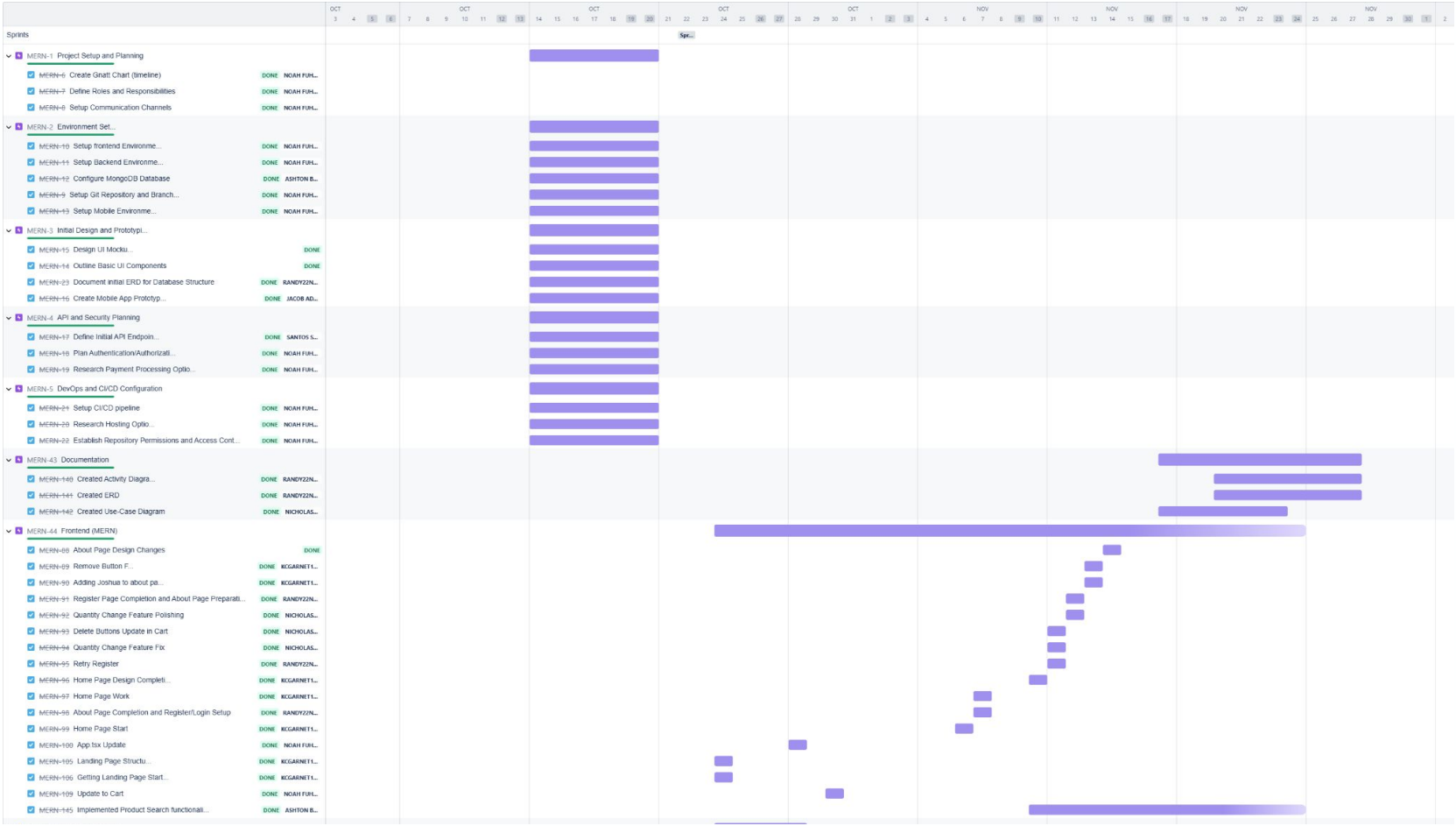
Discord

GitHub

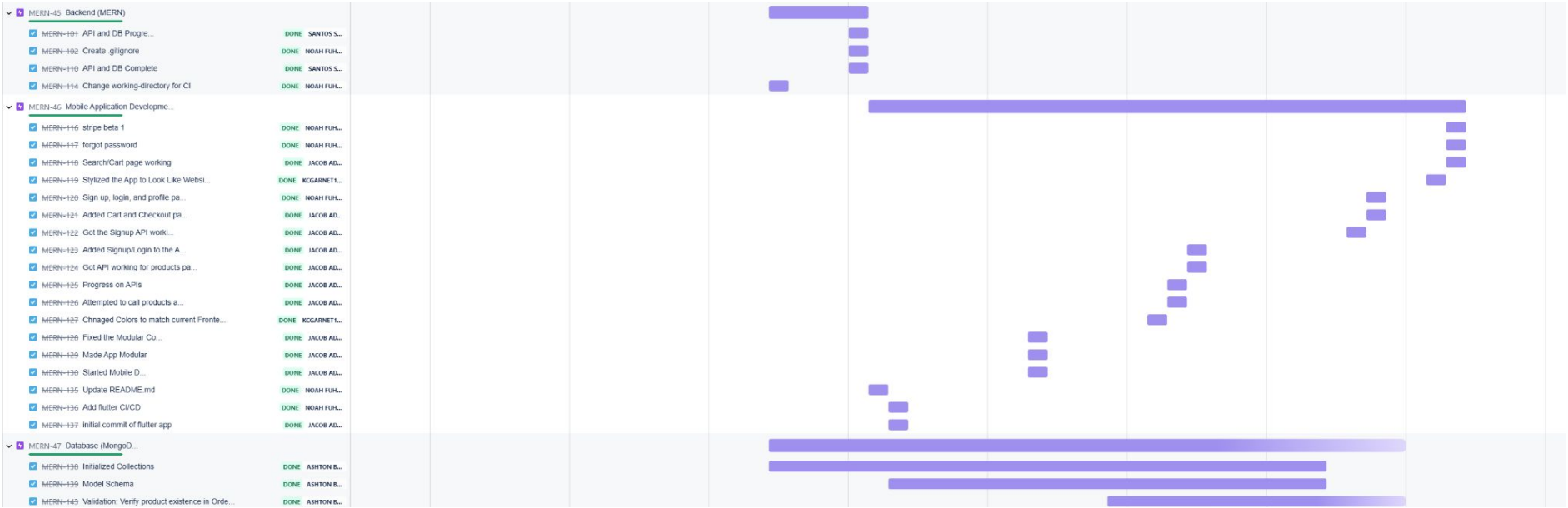

React

MongoDB

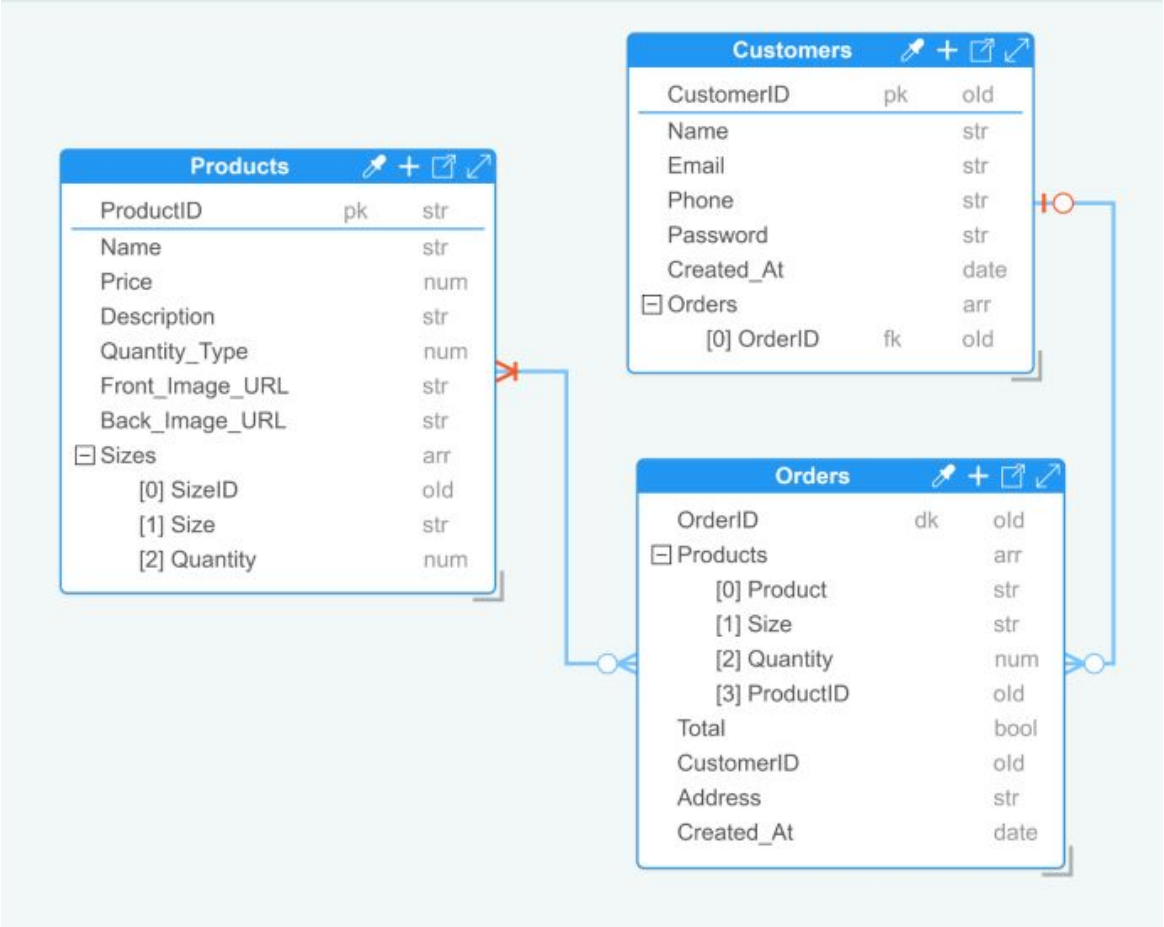
Gantt Chart


Sprints
MERN-1 Project Setup and Planning
MERN-6 Create Gnatt Chart (timeline) DONE NOAH FUH...
MERN-7 Define Roles and Responsibilities DONE NOAH FUH...
MERN-8 Setup Communication Channels DONE NOAH FUH...
MERN-2 Environment Set...
MERN-10 Setup frontend Environme... DONE NOAH FUH...
MERN-11 Setup Backend Environme... DONE NOAH FUH...
MERN-12 Configure MongoDB Database DONE ASHTON B...
MERN-9 Setup Git Repository and Branch... DONE NOAH FUH...
MERN-13 Setup Mobile Environme... DONE NOAH FUH...
MERN-3 Initial Design and Prototypi...
MERN-15 Design UI Mocku... DONE
MERN-14 Outline Basic UI Components DONE
MERN-23 Document initial ERD for Database Structure DONE RANDY22N...
MERN-16 Create Mobile App Prototyp... DONE JACOB AD...
MERN-4 API and Security Planning
MERN-17 Define Initial API Endpoin... DONE SANTOS S...
MERN-18 Plan Authentication/Authorizati... DONE NOAH FUH...
MERN-19 Research Payment Processing Optio... DONE NOAH FUH...
MERN-5 DevOps and CI/CD Configuration
MERN-21 Setup CI/CD pipeline DONE NOAH FUH...
MERN-20 Research Hosting Optio... DONE NOAH FUH...
MERN-22 Establish Repository Permissions and Access Cont... DONE NOAH FUH...
MERN-43 Documentation
MERN-140 Created Activity Diagra... DONE RANDY22N...
MERN-141 Created ERD DONE RANDY22N...
MERN-142 Created Use-Case Diagram DONE NICHOLAS...
MERN-44 Frontend (MERN)
MERN-88 About Page Design Changes DONE
MERN-89 Remove Button F... DONE KCGARNET1...
MERN-90 Adding Joshua to about pa... DONE KCGARNET1...
MERN-91 Register Page Completion and About Page Preparati... DONE RANDY22N...
MERN-92 Quantity Change Feature Polishing DONE NICHOLAS...
MERN-93 Delete Buttons Update in Cart DONE NICHOLAS...
MERN-94 Quantity Change Feature Fix DONE NICHOLAS...
MERN-95 Retry Register DONE RANDY22N...
MERN-96 Home Page Design Completi... DONE KCGARNET1...
MERN-97 Home Page Work DONE KCGARNET1...
MERN-98 About Page Completion and Register/Login Setup DONE RANDY22N...
MERN-99 Home Page Start DONE KCGARNET1...
MERN-100 App.tsx Update DONE NOAH FUH...
MERN-105 Landing Page Structu... DONE KCGARNET1...
MERN-106 Getting Landing Page Start... DONE KCGARNET1...
MERN-109 Update to Cart DONE NOAH FUH...
MERN-145 Implemented Product Search functionali... DONE ASHTON B...
OCT
NOV

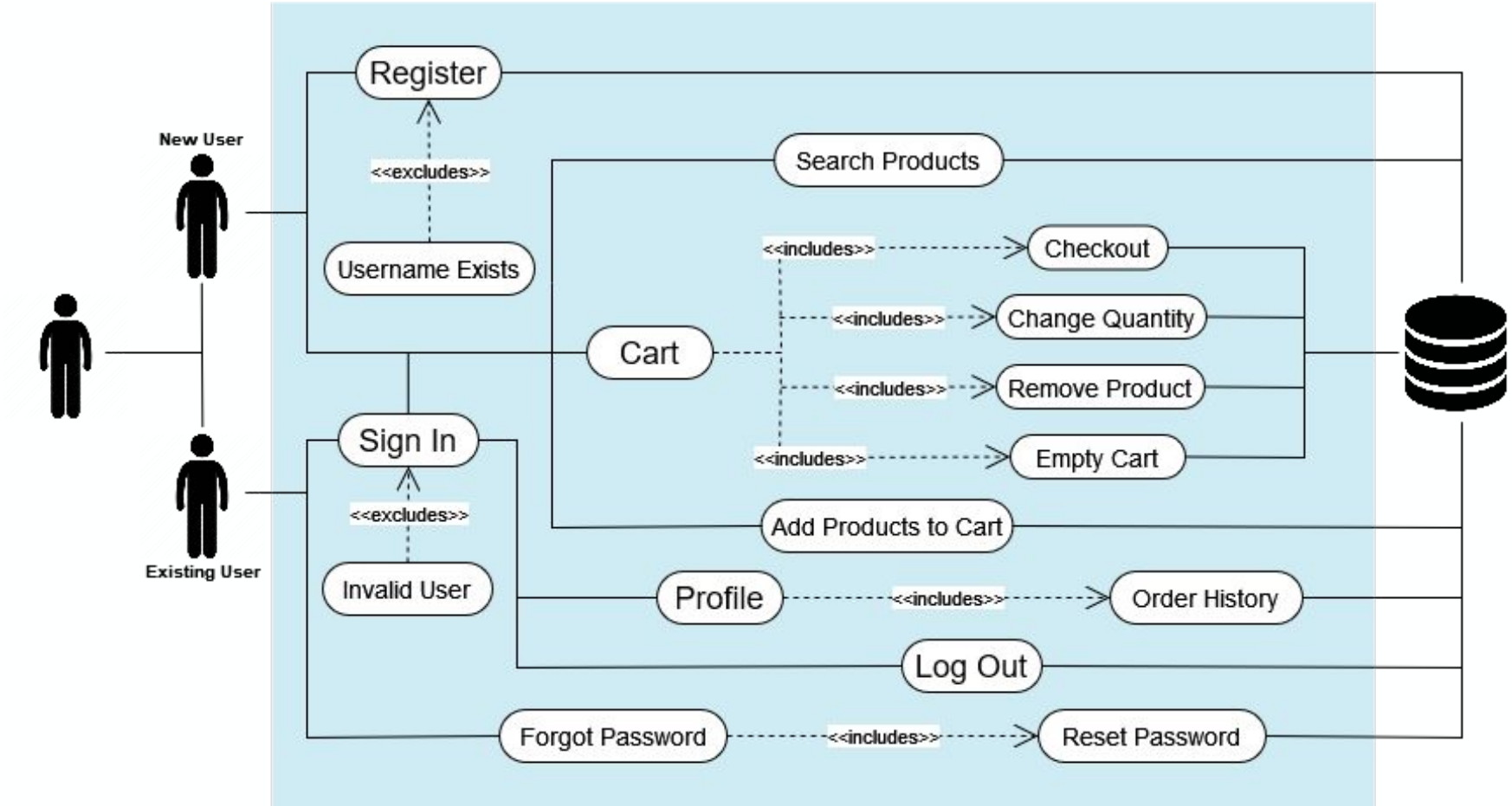
Gantt Chart


MERN-45 Backend (MERN)
MERN-101 API and DB Progre... DONE SANTOS S...
MERN-102 Create .gitignore DONE NOAH FUH...
MERN-110 API and DB Complete DONE SANTOS S...
MERN-114 Change working-directory for CI DONE NOAH FUH...
MERN-46 Mobile Application Developme...
MERN-116 stripe beta 1 DONE NOAH FUH...
MERN-117 forgot password DONE NOAH FUH...
MERN-118 Search/Cart page working DONE JACOB AD...
MERN-119 Stylized the App to Look Like Websi... DONE KCGARNET1...
MERN-120 Sign up, login, and profile pa... DONE NOAH FUH...
MERN-121 Added Cart and Checkout pa... DONE JACOB AD...
MERN-122 Got the Signup API worki... DONE JACOB AD...
MERN-123 Added Signup/Login to the A... DONE JACOB AD...
MERN-124 Got API working for products pa... DONE JACOB AD...
MERN-125 Progress on APIs DONE JACOB AD...
MERN-126 Attempted to call products a... DONE JACOB AD...
MERN-127 Chnaged Colors to match current Fronte... DONE KCGARNET1...
MERN-128 Fixed the Modular Co... DONE JACOB AD...
MERN-129 Made App Modular DONE JACOB AD...
MERN-130 Started Mobile D... DONE JACOB AD...
MERN-135 Update README.md DONE NOAH FUH...
MERN-136 Add flutter CI/CD DONE NOAH FUH...
MERN-137 Initial commit of flutter app DONE JACOB AD...
MERN-47 Database (MongoD...
MERN-138 Initialized Collections DONE ASHTON B...
MERN-139 Model Schema DONE ASHTON B...
MERN-143 Validation: Verify product existence in Orde... DONE ASHTON B...

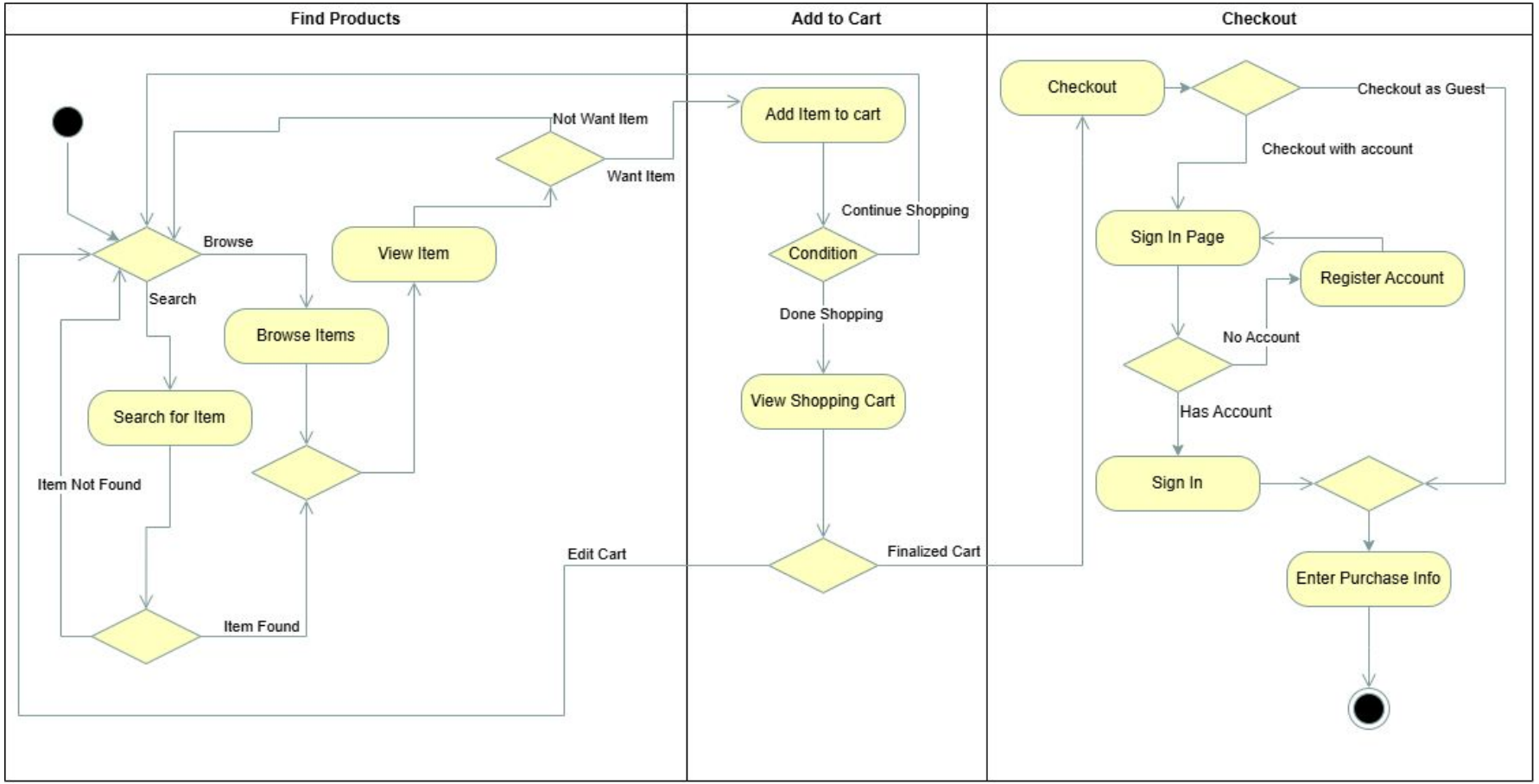
Entity Relationship Diagram



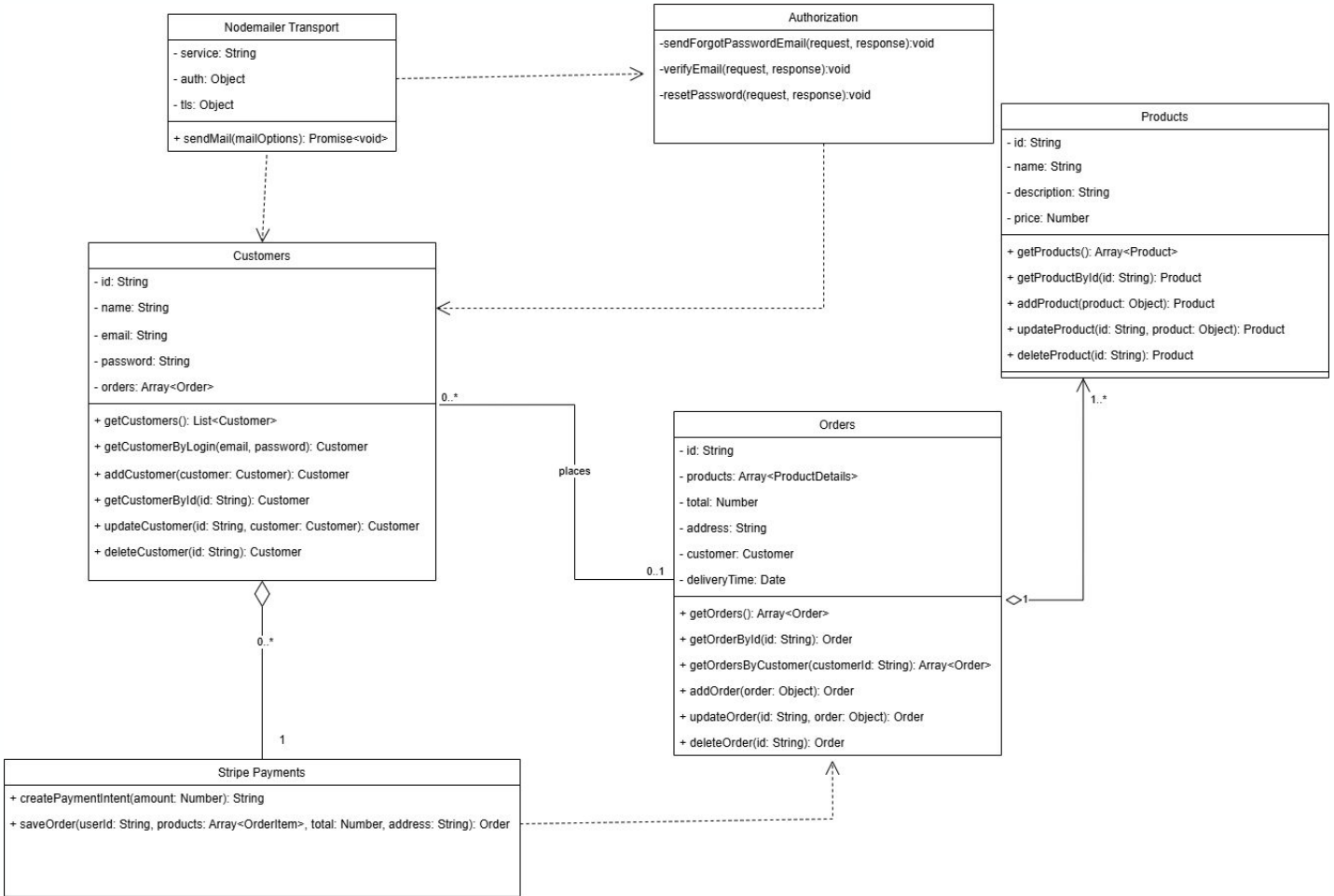
Use Case Diagram



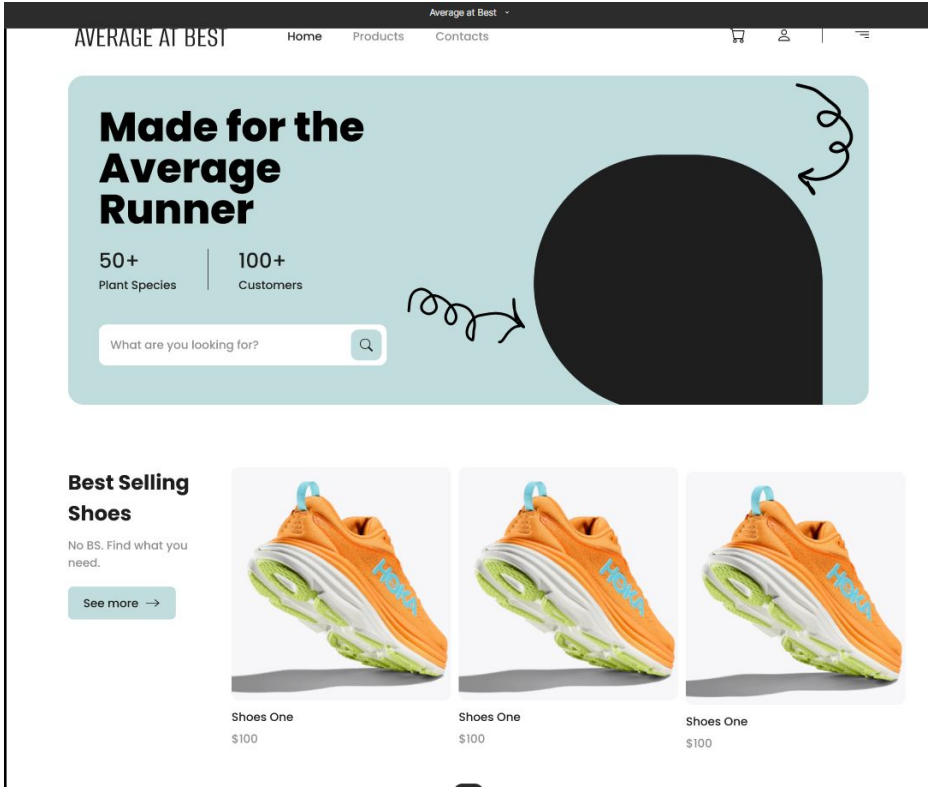
Activity Diagram

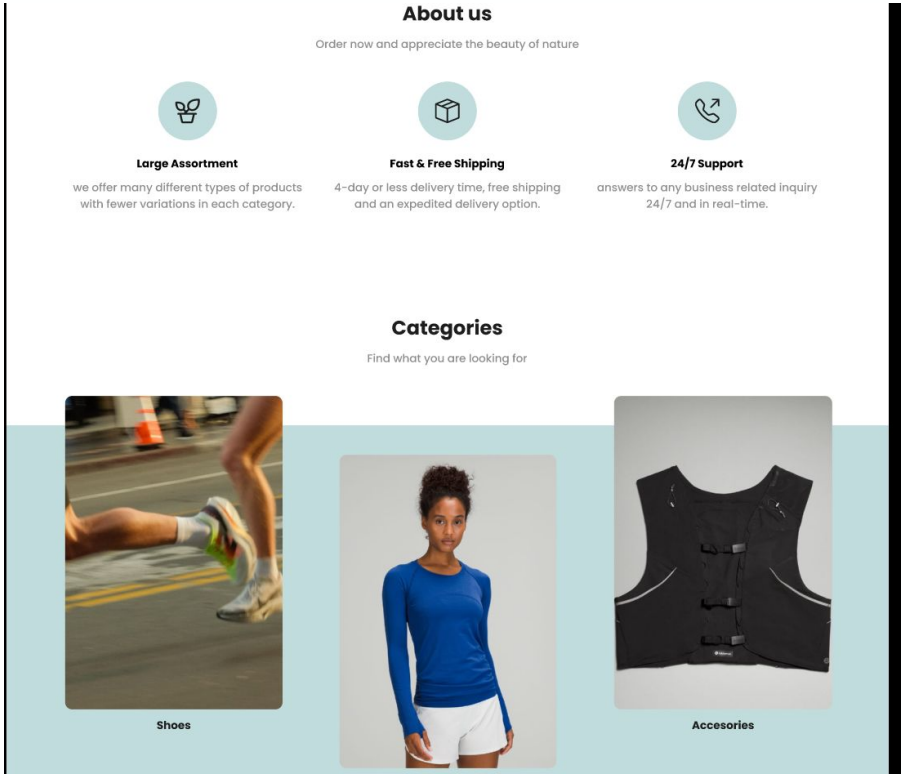


Class Diagram

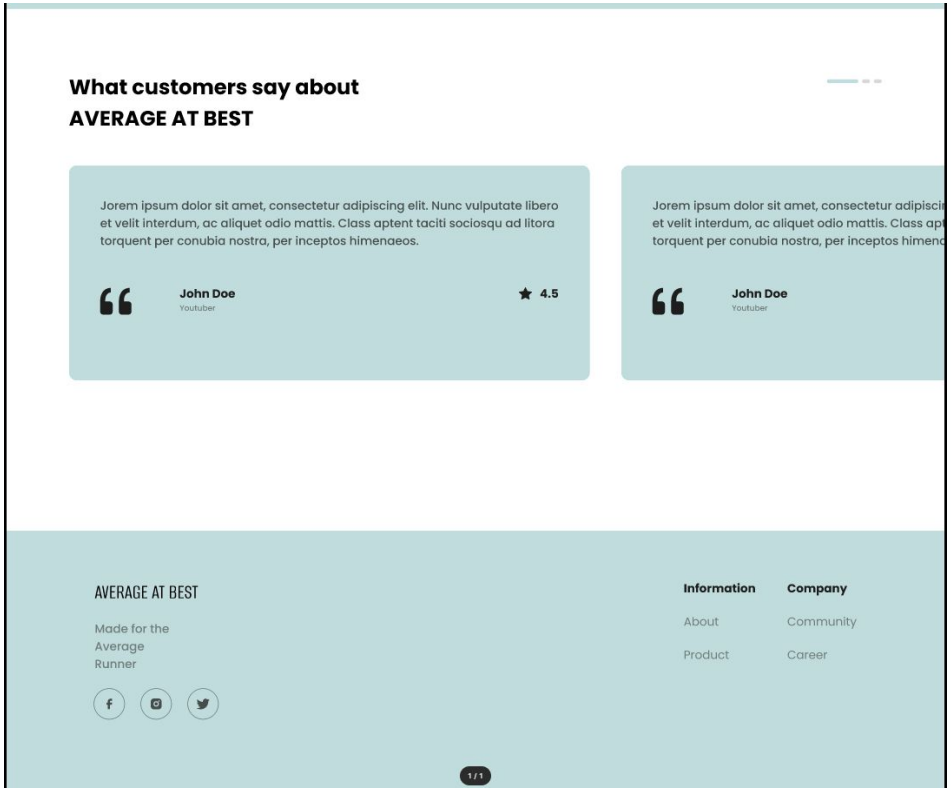


Prototype


Average at Best
AVERAGE AT BEST
Home
Products
Contacts
Made for the Average Runner
50+
Plant Species
100+
Customers
What are you looking for?
Best Selling Shoes
No BS. Find what you need.
See more
Shoes One
$100
Shoes One
$100
Shoes One
$100


About us
Order now and appreciate the beauty of nature
Large Assortment
we offer many different types of products with fewer variations in each category.
Fast & Free Shipping
4-day or less delivery time, free shipping and an expedited delivery option.
24/7 Support
answers to any business related inquiry 24/7 and in real-time.
Categories
Find what you are looking for
Shoes
Accesories

Prototype


What customers say about
AVERAGE AT BEST
Jorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc vulputate libero et velit interdum, ac aliquet odio mattis. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.
John Doe
Youtuber
4.5
Jorem ipsum dolor sit amet, consectetur adipisci et velit interdum, ac aliquet odio mattis. Class apt torquent per conubia nostra, per inceptos himen
John Doe
Youtuber
AVERAGE AT BEST
Made for the Average Runner
Information
About
Product
Company
Community
Career
1 / 1

Things that went well

- **Creating Registration Page**
- **Products Page**
- **Creating the database**
- **Mobile**

Things that did not go well

- **Scheduling deadlines**
- **Learning mobile development**
- **Connecting Products to Everything**

Unit Testing

```
PS C:\Users\noah1\Documents\GitHub\GITMERN> npx jest
 PASS  tests/controllers/product.controller.test.js
  Product Controller
    getProducts
      √ should fetch all products successfully (3 ms)
      √ should throw an error if fetching products fails (6 ms)
    getProductById
      √ should fetch a specific product by ID
      √ should throw an error if fetching a product by ID fails
    updateProduct
      √ should update a product successfully
      √ should throw an error if updating a product fails
    deleteProduct
      √ should delete a product successfully (1 ms)
      √ should throw an error if deleting a product fails (1 ms)

----------------------|---------|----------|---------|---------|-------------------
File                  | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
----------------------|---------|----------|---------|---------|-------------------
All files             |   84.37 |      100 |      80 |   84.37 |
 controllers          |   82.75 |      100 |      80 |   82.75 |
  productController.js |   82.75 |      100 |      80 |   82.75 | 26-31
 models               |     100 |      100 |     100 |     100 |
  Product.js          |     100 |      100 |     100 |     100 |
----------------------|---------|----------|---------|---------|-------------------
Test Suites: 1 passed, 1 total
Tests:       8 passed, 8 total
Snapshots:   0 total
Time:        1.041 s, estimated 2 s
Ran all test suites.
```

API DEMO

Swaggerhub

APP DEMO

email: averageatbestdev@outlook.com
pass: AABDevStripe
Card Number: 4242 4242 4242 4242
Expiration Date: Any valid date in the future (e.g., 12/34).
CVV: Any 3-digit number (e.g., 123).

http://www.aab.run/
https://dashboard.stripe.com/test/payments