

September 18th:

[1:51 PM]Kiran:

@everyone Are you guys down to do the optional stuff in Milestone 3 for the higher grade?

[1:52 PM]Kiran:

I'm partially familiar with AWS, so we could do it on my account

[1:52 PM]Kiran:

But I haven't actually done what they are asking us to do in AWS before

[1:53 PM]Ben:

Is the optional stuff just deploying to aws

[1:53 PM]Jamie:

should we have a meeting tonight

[1:53 PM]Jamie:

to assign tasks for this sprint or

[1:53 PM]Kiran:

Optional : complete these to achieve a higher grade Deploy Docker Images to AWS Completed Custom User story – this is a feature that was not present in the initial story. Refactoring code --- teams should deliver a "refactoring report".

Technical Enhancements, researched on your own, that could be considered "further enhancements" include: Use of Kubernetes or Kubernetes-like platforms Serverless style deployments

Other technology additions agreed with your Product Owner Note that there will be an associated Final Presentation to describe your project and your application; details to follow.

@Jamie

should we have a meeting tonight

[1:54 PM]Kiran:

I'm free after 8:30 pm

[1:54 PM]Ben:

Yeah I'm down for doing the extra stuff I'm free after 8:30 too

[1:55 PM]Jamie:

i see peter is rock and stoning

[1:55 PM]Jamie:

but yeah

[1:55 PM]Jamie:

maybe 9pm?

[1:55 PM]Ben:

I think we will probably have to add lots of tasks for this sprint

[1:55 PM]Tyler:

I should be free ~9

[1:55 PM]Jamie:

bro they expect us to have implemented all this in sprint 2

[1:55 PM]Kiran:

Ok let's do 9 then

[1:56 PM]Jamie:

whatever happened to incremental and iterative design

[1:56 PM]Jamie:

this is not agile

[1:56 PM]Jamie:
this is the opposite of agile

[1:56 PM]Kiran:
@everyone Meeting 9pm today

[1:56 PM]Jamie:
its clumsy

[1:56 PM]Ben:
Not all of it just some of it

[5:16 PM]Andy:
ill be picking up my dad from the airport around that time tonight but will try
and listen in from my phone

[5:19 PM]Ben:
Is there anything particular you want to do Andy? Because if you just let us know
what you wanted to do you can probably skip the meeting

[5:29 PM]Andy:
nah i'm not actually fully sure so coming to this meeting is prob a good idea

[8:59 PM]Kiran:
@everyone Meeting reminder

[9:11 PM]Ben:
/search?name=NAME&category=CATEGORY&supermarket=SUPERMARKET&priceUpper=PRICE&priceLower=PRICE&sortBy=NAME&sortOrder=ORDER

[9:44 PM]Kiran:
@everyone For the work log, could you please make sure the date is a weekday.
If you did work on the weekend, just say that you did it on the next Monday.
Thanks

[10:36 PM]Andy:

Action Items			
No.	Item	Who	By
1	Implement cart price calculation Implement adding items to favourites	Tyler	28/09
2	Connecting backend with frontend: Search Page Product Page Cart Page User Profile Page Notifications Page	Kiran " " Andy "	28/09
3	Implementing search term capture from UI Create user repository Port test data into database	Ben	28/09
4	Add filter menu Add filter functionality	Jamie	28/09
5	Implement adding new account to database Create user login	Peter	28/09
6	Finish continuous integration with frontend	Andy	28/09

[10:37 PM]Andy:

Information / Decisions

No.	Item
1	Discuss further on adding a favourites feature and optional task for higher grade
2	Begin connecting backend to respective pages (Search, Product, User Profile, Notifications)
3	Search term format: <code>/search?name=NAME&category=CATEGORY&supermarket=SUPERMARKET&priceUpper=PRICE&priceLower=PRICE&sortBy=NAME&sortOrder=ORDER</code>
4	Reminder to link burndown chart tasks with GitHub project tasks
5	Need to create a *PROJECT REPORT* a PROJECT REPORT: <ul style="list-style-type: none"> ◦ a "Vision" statement, i.e., a description of the why the product is "valuable" (about 1/2 page) ◦ a figure showing the final basic system architecture / design --- this does not need to be a UML diagram: a box-and-lines sketch architecture is fine; focus on the tiers : front layers, backend (services and their layers), datastore and what components are in each ◦ a report describing any refactoring, including refactoring to a Design Pattern, if appropriate, as well as the "smells" that you found (i.e., purpose for refactoring) ◦ an overview of your Git organisation and how often you committed (include in report) ◦ a description of your Scrum process: how often you met, who was Scrum Master (show the velocity of the team through all milestones for all sprints) ◦ a diagram of your Deployment pipeline setup: i.e., CI/CD, steps in the pipeline, automation tools in the pipeline
6	Next meeting Thursday 21st September
7	Challenge 2 Monday 2 nd October 4:30pm – Tuesday 3 rd October 4:30pm
8	Milestone 3 Due Sunday 8 th October

September 19th:

[10:12 AM]Ben:

Hey just wanted to shoot yall a quick message so just looking at the product data Andy has generated there is a bit of a disconnect between what it has and how I've implemented it so bascially I have implemented it so each thing is an individual product object for each supermarket because I thought there might be disconnects between the exact naming and sizing between supermarkets. Andy has collected the same products with the same name and found prices for each supermarket. So I think the way that Andy has implemented it is probably better and easier for us to compare prices between products and I can change the backend to work this way it'll just take some time maybe a day or so

[10:13 AM]Ben:

alternatively I can just build upon my existing implementation to work this way

[10:14 AM]Andy:

In similar fashion I can change the data so works either way for me

[10:15 AM]Andy:

I think do whatever is easiest for you

[10:16 AM]Ben:

I think changing the backend will take the most time but it will probably give us the easier time when implementing

[12:44 PM]Ben:

[databaseERD4.pdf](#)

25.78 KB

[12:44 PM]Ben:

Data base ERD 4.0

@Ben

Hey just wanted to shoot yall a quick message so just looking at the product data Andy has generated there is a bit of a disconnect between what it has and how I've implemented it so bascially I have implemented it so each thing is an individual product object for each supermarket because I thought there might be disconnects between the exact naming and sizing between supermarkets. Andy has collected the same products with the same name and found prices for each supermarket. So I think the way that Andy has implemented it is probably better and easier for us to compare prices between products and I can change the backend to work this way it'll just take some time maybe a day or so

[3:28 PM]Jamie:

make sure you document this change in SRS

[3:28 PM]Jamie:

as a design choice

[9:32 PM]Jamie:

@Tyler erm did we never make a user database

[9:33 PM]Tyler:

nah your gonna needa make a .sql file for it and do a bunch of flyway

[9:33 PM]Jamie:

i thought you guys had it ready

[9:33 PM]Tyler:

nah, dont think so

[9:33 PM]Tyler:

ask ben he did it all

[9:33 PM]Jamie:

well i can't exactly work on login functionality

[9:34 PM]Tyler:

are you doing the backend part of it?

[9:34 PM]Jamie:

i thought i was doing front end

[9:34 PM]Tyler:

did we have anyone assigned to the backend part of it?

[9:35 PM]Jamie:

doesn't say

[9:35 PM]Tyler:

right, well i could get on the backend part of it i guess

[9:35 PM]Tyler:

ill get it done tommorow, your gonna have to work on other parts till then though

[9:36 PM]Jamie:

no worries

[9:38 PM]Ben:

The sql database is there but there is no actual implementation for users and notifications I think

[9:38 PM]Tyler:

@Ben

The sql database is there but there is no actual implementation for users and notifications I think

[9:42 PM]Jamie:

whats the schema for the user account

[9:42 PM]Jamie:

and what basic verification do i need

[9:45 PM]Ben:

customer_id - int autogenerated primary eky customer_name - var char(255)

customer_email - var char(255) customer_address - var char(255)

password_hash - binary

[9:45 PM]Ben:

the customer password could probably be changed to just be a string for simplicity

September 20th:

3:13 PM]Tyler:

@Jamie User Functionality is now implemented (edited)

[3:14 PM]Tyler:

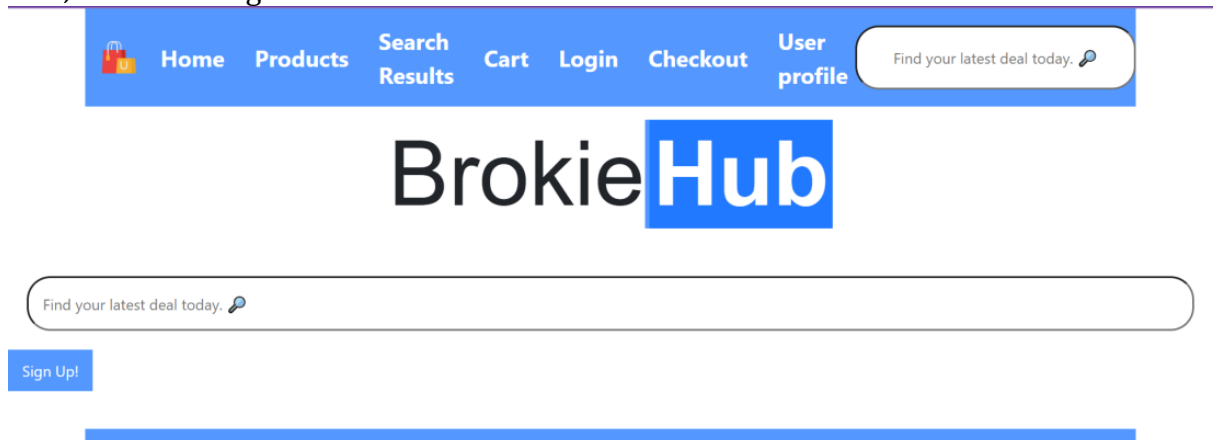
you probs only need to use attemptLogin

[3:14 PM]Tyler:

it uses their email as a username

[4:16 PM]Jamie:

lads, i have some good news and some bad news



[6:04 PM]Kiran:

@Jamie What's the news?

@Kiran

@Jamie What's the news?

[6:04 PM]Jamie:

good news, i fixed bootstrap

[6:04 PM]Jamie:

bad news, now it looks like that

[6:04 PM]Jamie:

even better news

[6:04 PM]Jamie:

i've got the sign up looking like a sign up

[6:05 PM]Kiran:

Sweet, looks good to me

[6:05 PM]Kiran:

The search bar is a bit long. But we can say it's like that because we have a LONG list of products

[6:27 PM]Kiran:

@Ben Hey mate, do we have any products to test the search results page with?

[6:28 PM]Ben:

Not yet we don't have images yet for the products

[6:29 PM]Kiran:

Ok no worries

[6:29 PM]Kiran:

I'll try to use some dummy products to test the website then

[6:31 PM]Ben:

yeah sorry im able to add some products tomorrow itll just take me maybe 2 hours I'm just a bit preoccupied getting my OSP done atm then I can shift to focus onto SEPT

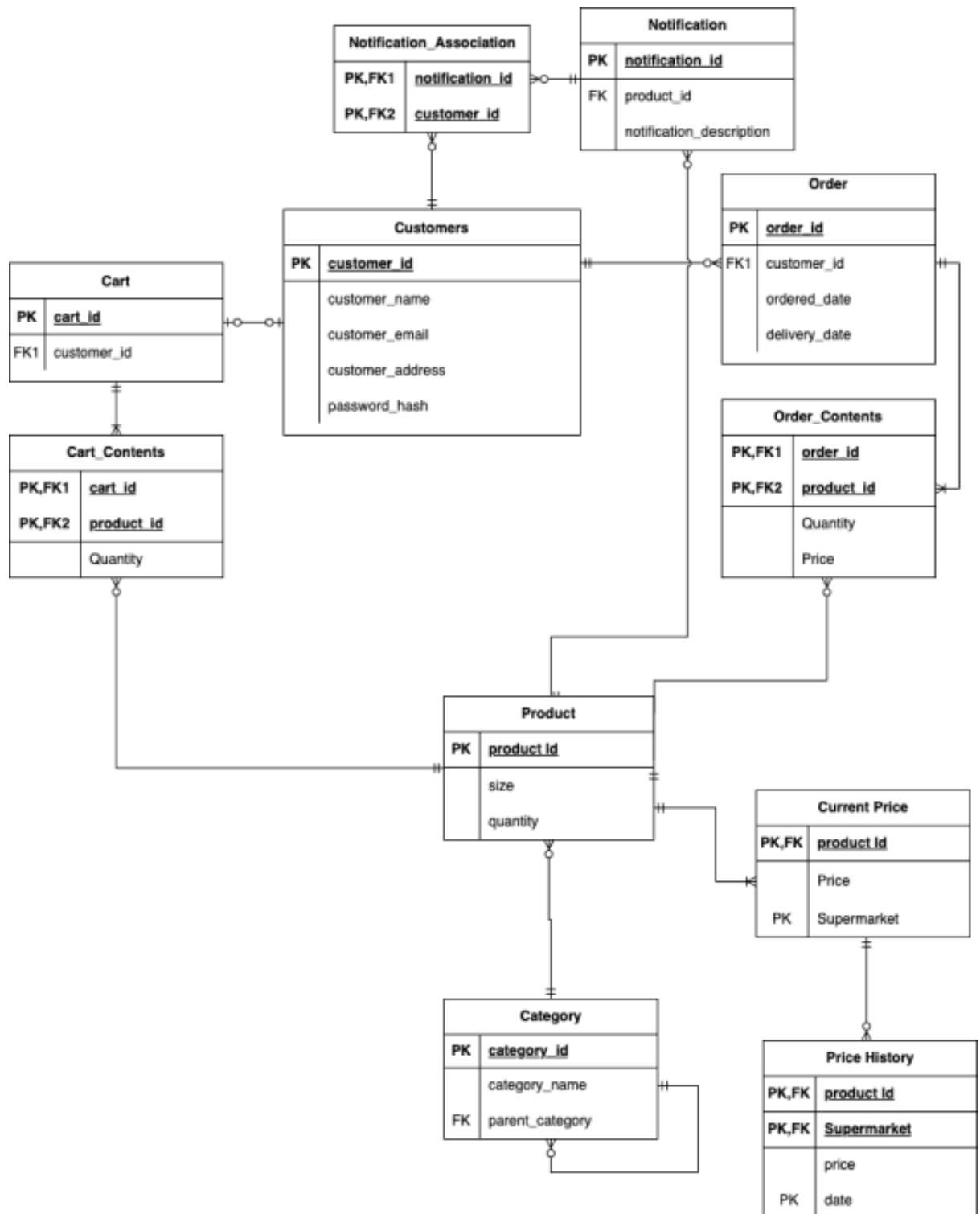
September 21st:

[12:05 AM]Kiran:

Don't stress Ben, I was just wondering

[12:06 PM]Ben:

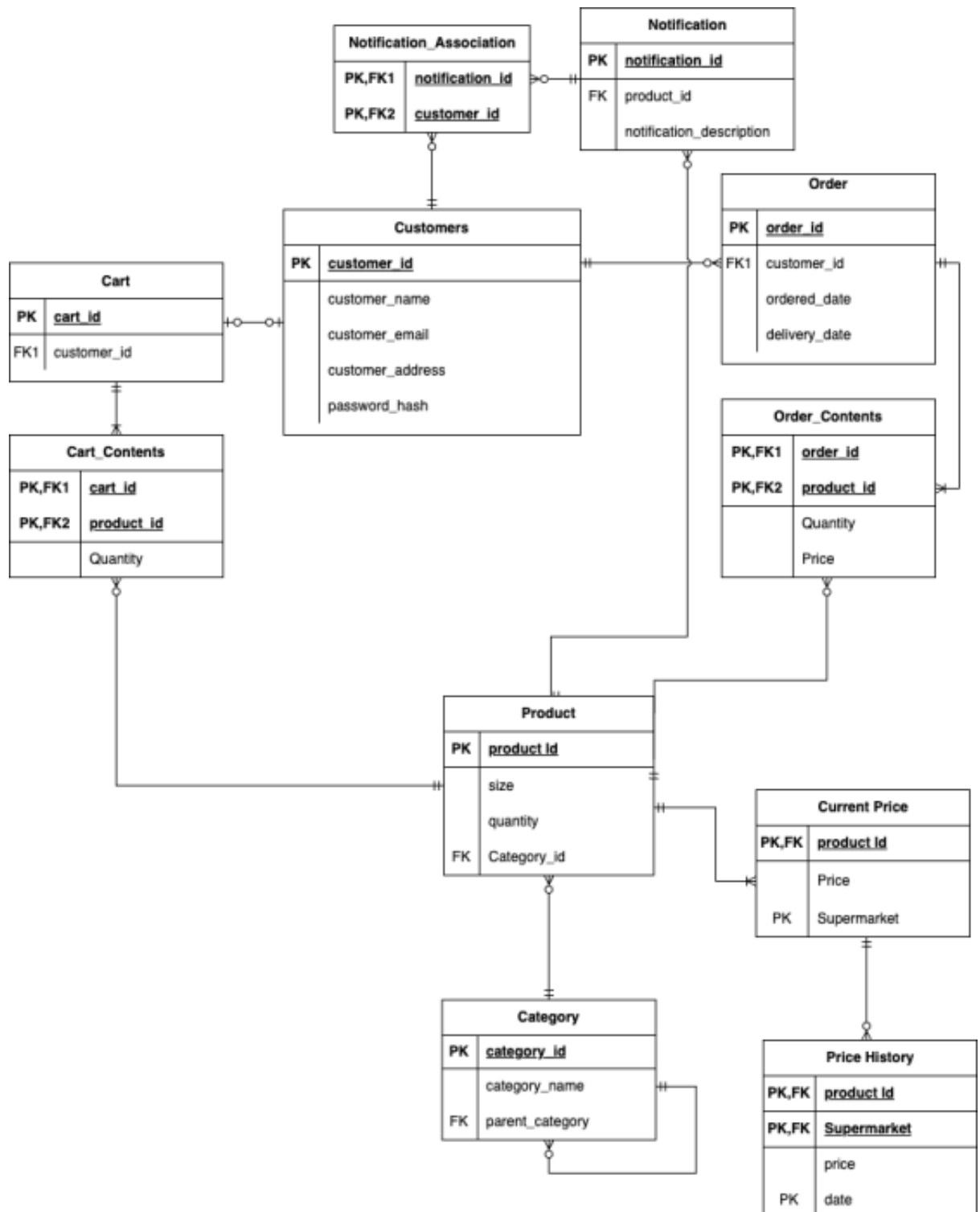
ERD 5



[12:19 PM]Ben:

ITS WRONG

[12:20 PM]Ben:



[1:12 PM]Jamie:

@Ben do you know how i'm supposed to do a post request

[1:13 PM]Ben:

like the format of the post request url or how to actually do one?

@Ben

like the format of the post request url or how to actually do one?

[1:13 PM]Jamie:

so i understand using a fetch() to call the right URL mapping for a function, but not sure how to do it for a post request

[1:14 PM]Ben:

<https://jasonwatmore.com/post/2020/02/01/react-fetch-http-post-request-examples>

[React + Fetch - HTTP POST Request Examples | Jason Watmore's Blog](#)

A quick set of examples to show how to send HTTP POST requests from React to a backend API using fetch()

[1:14 PM]Ben:

```
componentDidMount() {  
  // Simple POST request with a JSON body using fetch  
  const requestOptions = {  
    method: 'POST',  
    headers: { 'Content-Type': 'application/json' },  
    body: JSON.stringify({ title: 'React POST Request Example' })  
  };  
  fetch('https://reqres.in/api/posts', requestOptions)  
    .then(response => response.json())  
    .then(data => this.setState({ postId: data.id }));  
}
```

[1:30 PM]Jamie:

@Ben so in the fetch URL, it'd just be http://localhost:8080/v1/users?

[1:31 PM]Ben:

Yes that would work the mapping just needs to be added because it is missing

[1:31 PM]Ben:

to the backend

[1:31 PM]Jamie:

yeah i got tyler with me doing that rn

[3:21 PM]Ben:

Okay I've finished the product rework and am ready to pull that into Main but seeing as I am coming into class i'll wait to do it till then because it may take some time

[3:28 PM]Peter:

Did someone say merging issues...

[3:29 PM]Ben:

If we ignore them they don't exist

[4:32 PM]Kiran:

Hey guys I'm coming in

[4:32 PM]Kiran:

Just will be a little late

[4:50 PM]Andy:

@Peter git config core.protectNTFS false

[6:00 PM]Ben:

@everyone meeting with Dale @12pm

[6:00 PM]Ben:

MOnday

[7:55 PM]Ben:

can somebody just give a rundown of how to run the frontend?

[7:56 PM]Peter:

```
PS C:\Users\peter\OneDrive\Documents\2023\sept\team-project-group-p03-08\frontend\comparison-shopper> npm run
```

[7:59 PM]Ben:

```
● (base) ben@Bens-MacBook-Pro comparison-shopper % npm run
Lifecycle scripts included in comparison-shopper@0.1.0:
  start
    react-scripts start
  test
    react-scripts test

available via `npm run-script`:
  build
    react-scripts build
  eject
    react-scripts eject

○ (base) ben@Bens-MacBook-Pro comparison-shopper %
```

[7:59 PM]Ben:

just get this

[7:59 PM]Peter:

```
PS C:\Users\peter\OneDrive\Documents\2023\sept\team-project-group-p03-08\frontend\comparison-shopper> npm start
```

[7:59 PM]Ben:

thx

[8:00 PM]Ben:

nvm

[8:00 PM]Ben:

```
● (base) ben@Bens-MacBook-Pro comparison-shopper % npm run
Lifecycle scripts included in comparison-shopper@0.1.0:
  start
    react-scripts start
  test
    react-scripts test

available via `npm run-script`:
  build
    react-scripts build
  eject
    react-scripts eject

✖ (base) ben@Bens-MacBook-Pro comparison-shopper % npm start

> comparison-shopper@0.1.0 start
> react-scripts start

sh: react-scripts: command not found
```

[8:00 PM]Peter:

huh start works for me

[8:00 PM]Andy:

go back a directory with 'cd ..' and run it there?

[8:01 PM]Ben:

nah neither of those work in front end

[8:03 PM]Ben:
might need to run npm install stuff

[8:03 PM]Andy:
yes good idea

[8:04 PM]Ben:
thanks for the help all good now

@Ben
Click to see attachment


[8:04 PM]Jamie:
npm install react-scripts --save

[8:04 PM]Jamie:
do it in terminal

[8:04 PM]Jamie:
in comparison shopper

[8:05 PM]Ben:
yeah thanks all working now!

[9:34 PM]Ben:



[Home](#)

[Products](#)


[Search Results](#)

[Cart](#)

[Login](#)

[Checkout](#)

[User profile](#)

Find your latest deal today. 

Showing results for:

Total results: 100

Displaying: 10

Sort by

Filters

Product Name	Size	Unit	Category ID
Heinz Baked Beans	1	kg	5
Cottees Jam Strawberry Conserve	1	kg	5
Whole Chicken	2	kg	12
Light Rye Bread	1	kg	4
Australian Barramundi Portions	0	kg	13
Free Range Chicken Breast	1	kg	12
Sourdough Bread	1	kg	4
Espresso Coffee Beans	1	kg	5

3000/SearchResultsPage.js#

I haven't pushed the changes because I kinda gutted everything that was there but I can

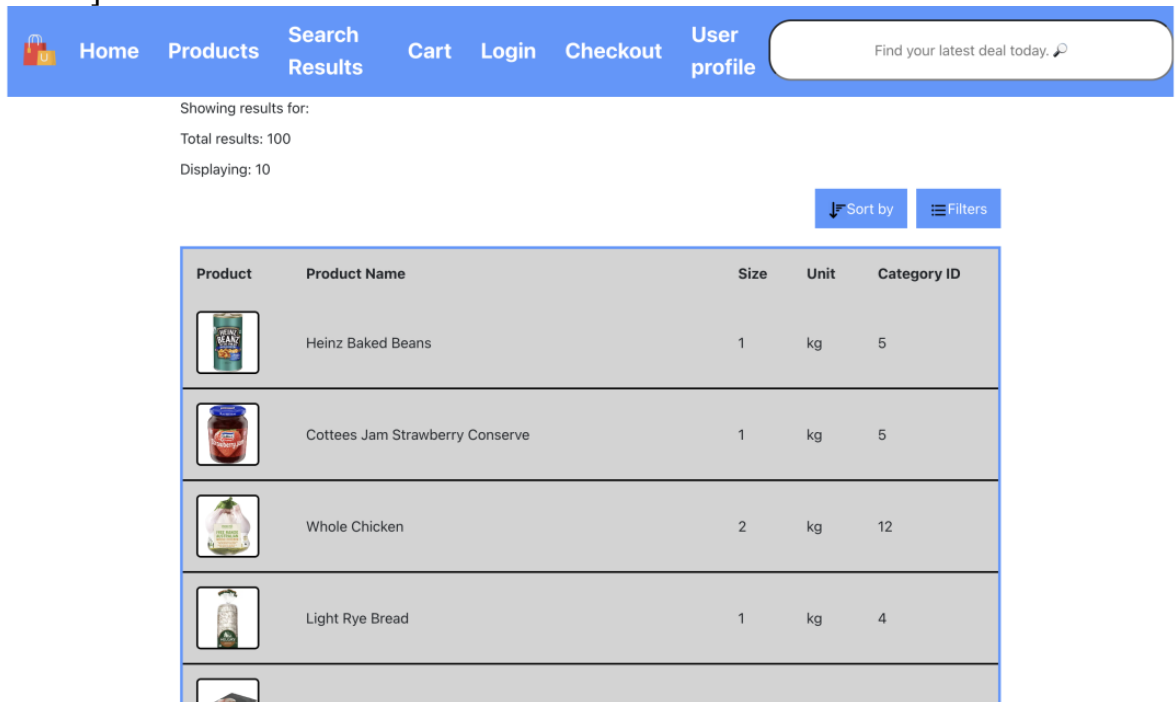
1

[10:52 PM]Ben:

Me and this assignment are in an abusive relationship I hate it and love it at the same time

1

[10:52 PM]Ben:



[10:52 PM]Ben:

imgs

[10:53 PM]Ben:

so when we pull into main there is now .webp images for each product

[12:55 PM]Kiran:

Ayooo the images really pull the page together (edited)

September 22nd:

[12:55 PM]Kiran:

Nice Ben

[12:56 PM]Kiran:

Hey guys, how do we want to get to the product page

[12:57 PM]Ben:

I was thinking covert the list items into buttons that link to the correct pages

[12:57 PM]Kiran:

Should we add a another column to the search results page that says 'more info' or make the product name the button?

@Ben

I was thinking covert the list items into buttons that link to the correct pages

[12:58 PM]Kiran:

Ok sweet, that's the second idea I wrote right? Or is it different?

[12:58 PM]Kiran:

By list items do you mean the product names?

[12:59 PM]Kiran:

It would be sick to have the image as the button, but it might be hard to do

[1:00 PM]Ben:

Well you should be able to make the whole row in the table into a button by wrapping it inside a wrapper element so anytime you click on it takes you to that products page

[1:08 PM]Kiran:

I'll give it a go

[3:03 PM]Jamie:

@Tyler can you change the back end so that if the database already has an existing user, it returns a 404 error or something like that

[3:03 PM]Tyler:

cant you just use an if statement?

[3:04 PM]Ben:

Just add a flway schema if you know how to do that

[3:04 PM]Tyler:

i dont

[3:05 PM]Ben:

Actually maybe not because you need the password hash :/

@Tyler

cant you just use an if statement?

[3:05 PM]Jamie:

problem is, when i try to sign up with a user that should hypothetically already exist

[3:05 PM]Jamie:

it still returns a http 200 response

[3:05 PM]Jamie:

you kinda need to change it in the router

[3:05 PM]Jamie:

so i can differentiate

[3:05 PM]Jamie:

<https://stackoverflow.com/questions/72754197/username-already-exists-in-the-database-in-express-js-mysql-react-js>

Stack Overflow

[username already exists in the database in express js, mysql, react js](https://stackoverflow.com/questions/72754197/username-already-exists-in-the-database-in-express-js-mysql-react-js)

I was trying to perform post, get, update and delete data(CRUD). Until now everything is perfectly fine. Now I was trying to add a functionality in my CRUD website like data already exists in the



[3:05 PM]Jamie:
and handle it on front end

[3:06 PM]Tyler:
the function itself returns true or false already, can you not use that?

[3:06 PM]Jamie:
no?

[3:07 PM]Jamie:
all i get is a Promise thats still pending

[3:10 PM]Jamie:
yeah i can't exactly handle errors if i'm not being given any errors in the first place

```
▼ Promise {<pending>} ⓘ main.7cf3ba08f1c12e2...9d.hot-update.js:34
  ► [[Prototype]]: Promise
  [[PromiseState]]: "fulfilled"
  [[PromiseResult]]: false

[Violation] Forced reflow while executing JavaScript took 37ms

repository.js:17
▼ Response {type: 'cors', url: 'http://localhost:8080/v1/users/create', redirected: false, status: 200, ok: true, ...} ⓘ
  body: (...)
  bodyUsed: false
  ► headers: Headers {}
  ok: true
  redirected: false
  status: 200
  statusText: ""
  type: "cors"
  url: "http://localhost:8080/v1/users/create"
  ► [[Prototype]]: Response
```

[3:11 PM]Tyler:
fair enough, on it now

[3:14 PM]Tyler:
just pushed a change, create user should throw an exception now

@Tyler
just pushed a change, create user should throw an exception now

[3:15 PM]Jamie:
you're on user mapping branch right

[3:15 PM]Tyler:

yep, im on vs code now forgot to push the commit

[3:15 PM]Tyler:
try now

@Tyler
yep, im on vs code now forgot to push the commit

[3:18 PM]Jamie:
its still giving me a status 200

[3:18 PM]Tyler:
should be throwing status 500 internal server error

[3:19 PM]Jamie:
did you use the stack overflow i sent

[3:20 PM]Tyler:
hol up im a lil confused

[3:21 PM]Tyler:
so how do i send the exception to the frontend

@Tyler
so how do i send the exception to the frontend

[3:22 PM]Jamie:

```
try{
  let {firstName} = req.body
  await try_insert_user(firstName)
  res.status(200).send({message: 'User inserted successfully!'})
}catch(err){
  //this part will catch the error thrown if the user exists
  res.status(400).send({message: err.message})
}
```

[3:23 PM]Jamie:
so your controller will call on the service

[3:23 PM]Jamie:
but catches the error itself

[3:23 PM]Jamie:
and then change result status to 400

[3:23 PM]Jamie:
i think

[3:23 PM]Tyler:
res doesnt exist

[3:23 PM]Tyler:
i cant call res.status

[3:24 PM]Tyler:
wtf is router.post even

[3:25 PM]Jamie:
@Ben help

[3:25 PM]Jamie:
brain is fried

[3:25 PM]Jamie:
maybe change the function so it returns HTTP status instead??

[3:28 PM]Tyler:
surely you can do something with this jamie

[3:28 PM]Tyler:
<https://stackoverflow.com/questions/55405187/react-class-how-to-return-a-boolean-after-a-promise-fetch-call>

Stack Overflow

React Class how to return a Boolean after a promise fetch call

hello i have this class when i call `Auth.isAuthenticated()` from another component in react, it always return false (its the default value), even if the server return a 200 response, witch sets this.



[3:30 PM]Ben:

Okay so I'm gonna say this it might not all make sense but I can elaborate if you don't understand. So basically you want to change the controller so it calls the function and returns the new user object or you can change the controller so if it returns true you send `https.CREATED 201` and if you return false you send a different https message like `HTTPS.BADREQUEST 400`

@Tyler

surely you can do something with this jamie

[3:30 PM]Jamie:

problem is the HTTP returns 200 regardless of whether the user exists in sign up or not

@Ben

Okay so I'm gonna say this it might not all make sense but I can elaborate if you don't understand. So basically you want to change the controller so it calls the function and returns the new user object or you can change the controller so if it returns true you send `https.CREATED 201` and if you return false you send a different https message like `HTTPS.BADREQUEST 400`

[3:31 PM]Tyler:

okay, how do i send a https message in the first place

[3:33 PM]Ben:

In the Java controller you can return a `RESPONSE ENTITY` with `http.status` that matches

[3:33 PM]Ben:

<https://stackabuse.com/how-to-return-http-status-codes-in-a-spring-boot-application/>

Stack Abuse

How to Return HTTP Status Codes in a Spring Boot Application

In this guide, we'll take a look at how to return HTTP status codes in Spring Boot applications with `@ResponseStatus`, `ResponseEntity` and `ResponseStatusException`.

[3:33 PM]Tyler:

okay cheers

[3:37 PM]Jamie:

>stackabuse

[3:38 PM]Tyler:
okay, check if that works now

[3:40 PM]Jamie:
@Tyler nope still getting http 200

[3:41 PM]Tyler:
check usercontroller on your side

[3:41 PM]Tyler:
you wanted it for user creation right?

[3:41 PM]Jamie:
ye

[3:42 PM]Tyler:
and did you restart the backend

[3:42 PM]Ben:
Maybe check that your internal logic in the repo and service if that doesn't work

[3:42 PM]Jamie:
oh right

[3:42 PM]Jamie:
back end

[3:42 PM]Jamie:
lemme restart that

[3:43 PM]Jamie:
LETS GOOOOO

[3:43 PM]Tyler:
YEAH

[3:53 PM]Jamie:
i hate my life

[3:53 PM]Jamie:
ok

[3:54 PM]Tyler:
elaborate
@Tyler
elaborate

[3:55 PM]Jamie:
so i've made it so if it detects a result thats a 400 error

[3:55 PM]Jamie:
it console logs

[3:55 PM]Jamie:
so i know it made it there

[3:55 PM]Jamie:
and what it should be doing

[3:55 PM]Jamie:
is returning false

[3:55 PM]Jamie:
however

[3:56 PM]Jamie:
const result = createUser(emailInput, usernameInput, addressInput, passwordInput);
console.log(result);

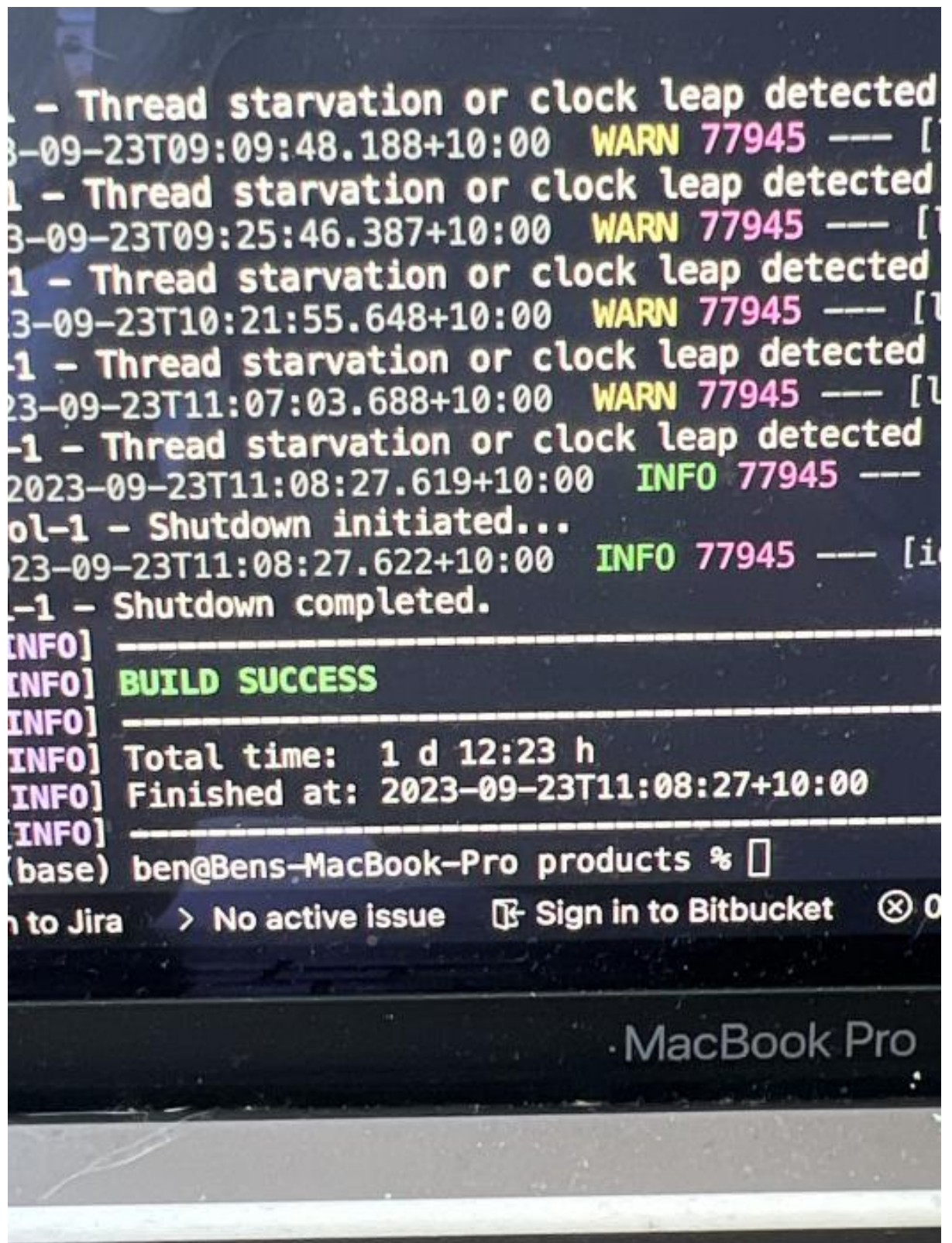
[3:56 PM]Jamie:
when i try to log the result

[3:56 PM]Jamie:

it doesn't work
[3:56 PM]Tyler:
right
[3:56 PM]Tyler:
i have no clue man
[4:01 PM]Jamie:
i gotta go to work
[4:01 PM]Jamie:
ughhhh
[4:01 PM]Tyler:
well, looks like this will be continuing tommorow
[4:12 PM]Jamie:
@Tyler good news, i've found a solution
[4:12 PM]Jamie:
bad news, i have no clue how this will affect the future
[4:12 PM]Tyler:
oh that sounds fun and definetly a good idea to implement into our software

September 23rd:

[11:08 AM]Ben:



1

[11:09 AM]Ben:
Uptime looking good
[12:25 PM]Kiran:

Hey guys You know how we access the products with
localhost:8080/v1/products

[12:25 PM]Kiran:

How would I access other things, such as the prices (edited)

[12:26 PM]Tyler:

you can access everything in the database at localhost:8080/h2-console/

[12:26 PM]Tyler:

just make sure you have the right db name

[12:26 PM]Kiran:

I'm on that now

[12:26 PM]Kiran:

Wdym

[12:26 PM]Tyler:

so you can see all the databases?

[12:26 PM]Kiran:

Yeah

[12:27 PM]Tyler:

so prices should be listed as a field in products?

[12:27 PM]Ben:

For the prices specifically you can find the price of a specific product by calling
localhost:8080/v1/products/price/id but for the price of all items there isn't
anything

[12:28 PM]Ben:

I've said this before because we have multiple prices for items to return the price
of all items i need to create a new function which searches the prices of all
products and chooses the smallest price

[12:28 PM]Kiran:

That's didn't work for me

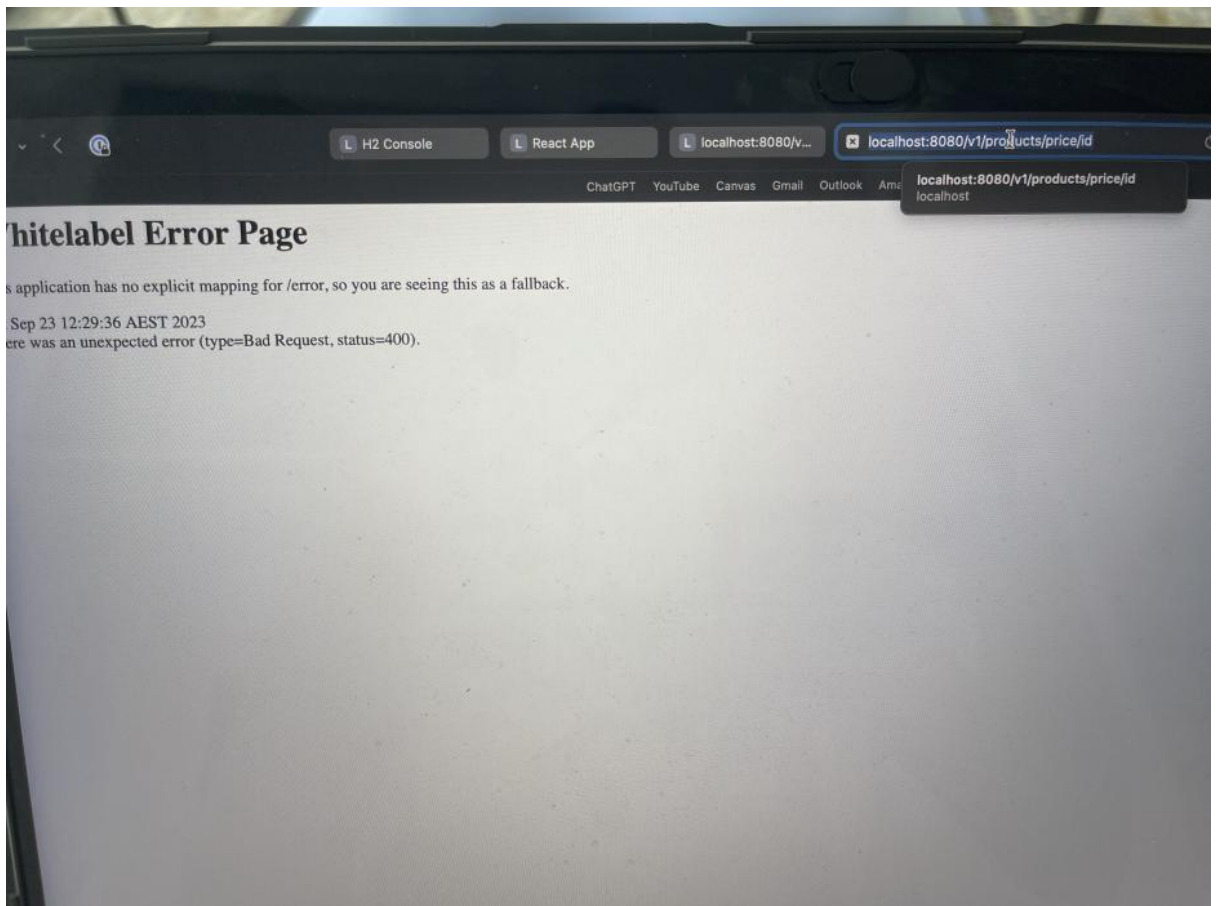
[12:29 PM]Ben:

Really?

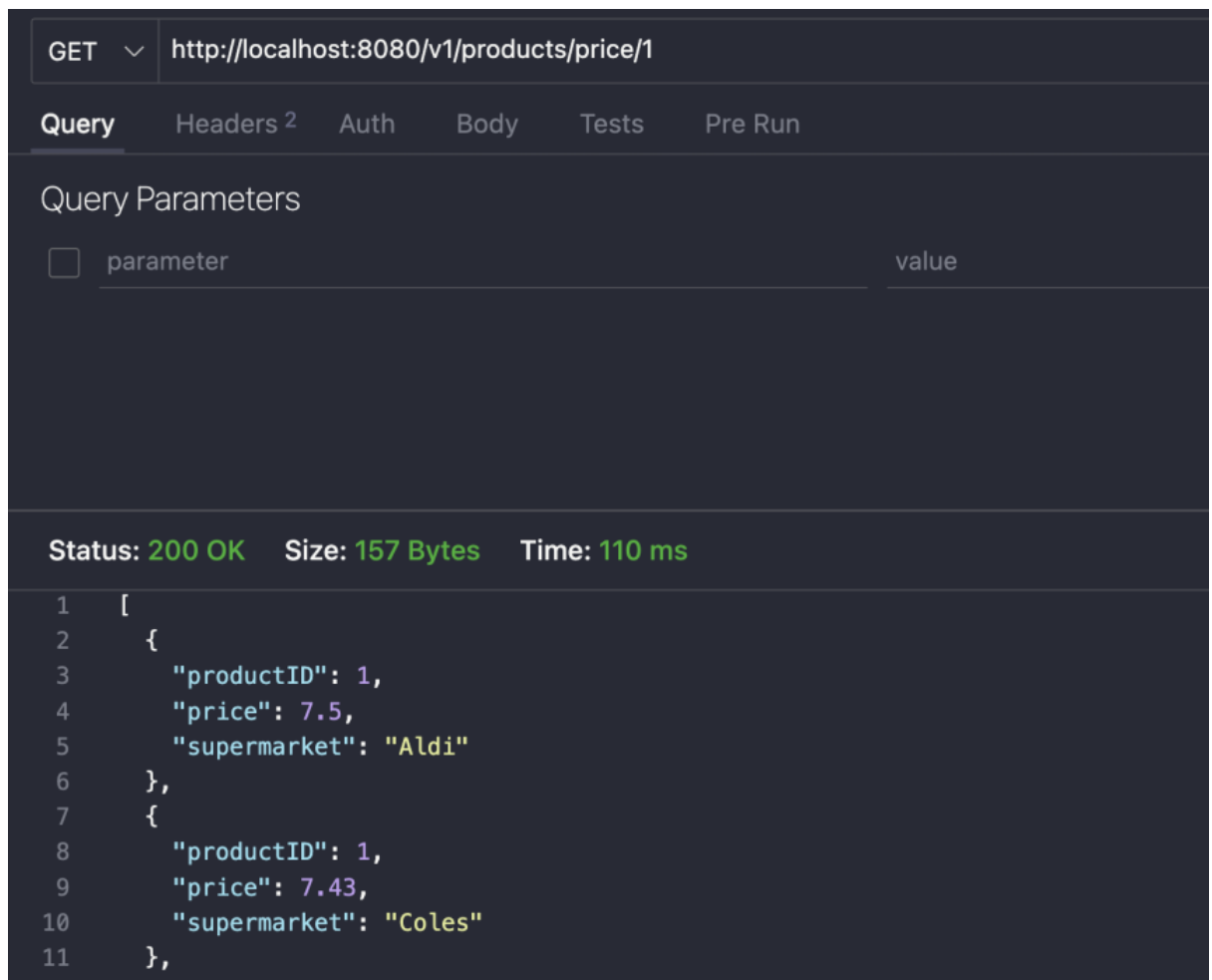
[12:29 PM]Kiran:

Yeah

[12:30 PM]Kiran:



[12:30 PM]Ben:



[12:30 PM]Ben:

nah you need to include an id

[12:30 PM]Ben:

and it will return the price information for that item with that id

[12:30 PM]Kiran:

Ahhh

[12:30 PM]Kiran:

Awesome it works now

[12:30 PM]Kiran:

Thanks

[1:19 PM]Kiran:

Hey guys

[1:19 PM]Kiran:

I'm working on the product page

[1:19 PM]Kiran:

And I get this error

[1:19 PM]Kiran:

ntend > comparison-shopper > src > Components > js ProductPage.js > [e] ProductPage > [e] useEffect() call

You, 3 minutes ago | 2 authors (ChromiOnAndSlam and others)

//import React from 'react';

import React, {useState, useEffect} from 'react';

import './ProductPage.css';

import Table from 'react-bootstrap/Table';

const addToCart = () => {

window.location.href = '/CartPage.js';

};

const ProductPage = () => {

const [products, setProducts] = useState([]);

const [loading, setLoading] = useState(true);

useEffect(() => {

async function fetchData() {

try {

const response = await fetch("http://localhost:8080/v1/products/price/1");

if (!response.ok) {

throw new Error('Failed to fetch data');

}

const data = await response.json();

setProducts(data);

setLoading(false);

} catch (error) {

console.error("Failed to fetch products:", error);

setLoading(false);

}

}

fetchData();

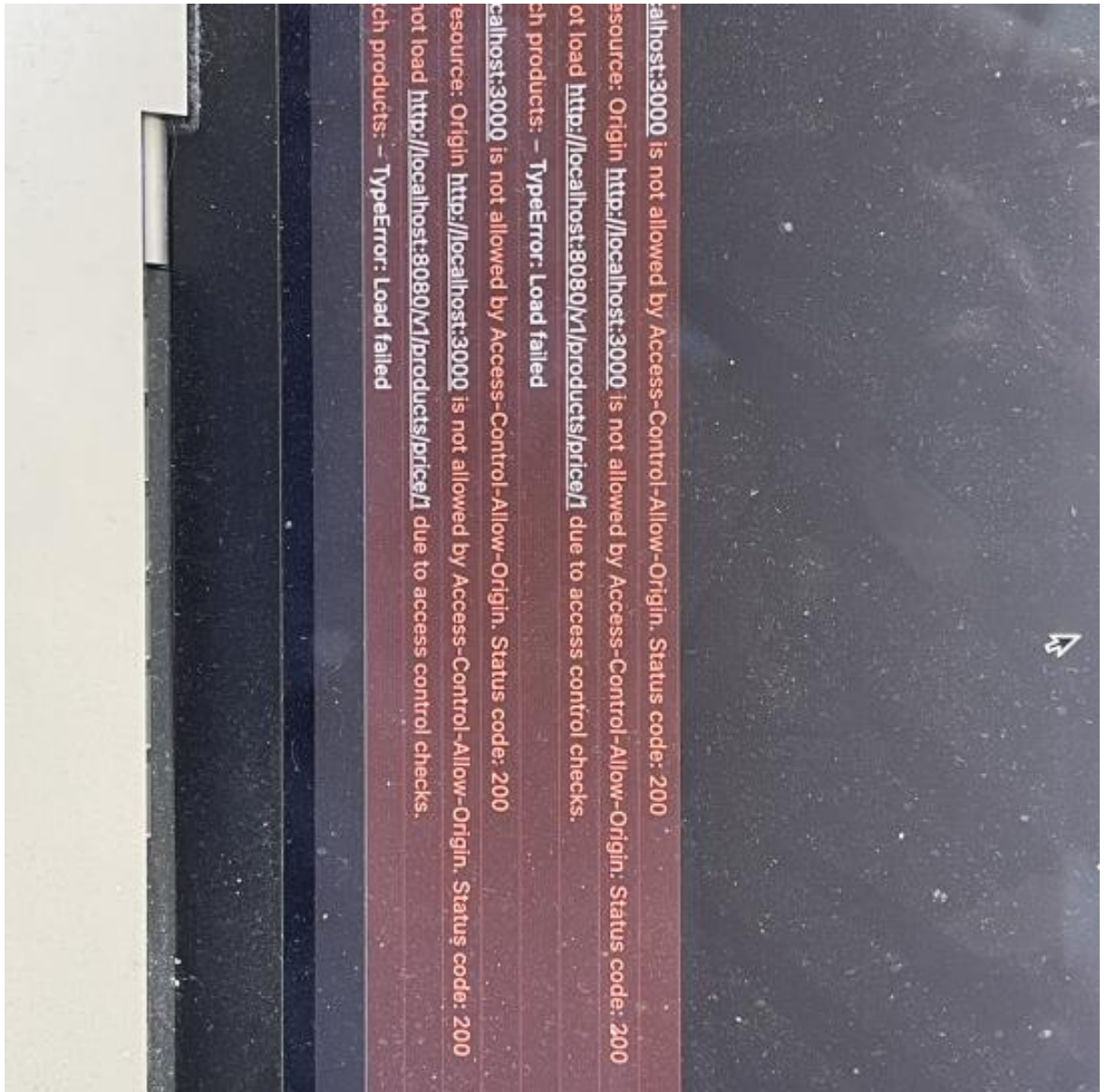
}, []);

if (loading) {

return <div>Loading...</div>;

}

34



[1:20 PM]Tyler:

can we get some non sideways photos please

[1:20 PM]Kiran:

It seems to be having trouble fetching

[1:20 PM]Tyler:

@Jamie how do you implement cors?

[1:20 PM]Tyler:

jamie managed to fix this on thursday

[1:21 PM]Jamie:

i put @CrossOrigin

[1:21 PM]Jamie:

atop the method

[1:21 PM]Jamie:

@Tyler btw have you also made the changes for the login as well

[1:21 PM]Kiran:


```
1 //import React from 'react';
2 import React, {useState, useEffect} from 'react';
3 import './ProductPage.css';
4 import Table from 'react-bootstrap/Table';
5
6 const addToCart = () => {
7   window.location.href = '/CartPage.js';
8 };
9
10 const ProductPage = () => {
11   const [products, setProducts] = useState([]);
12   const [loading, setLoading] = useState(true);
13
14   useEffect(() => {
15     async function fetchData() {
16       try {
17         const response = await fetch("http://localhost:8080/v1/products/price/1");
18         if (!response.ok) {
19           throw new Error('Failed to fetch data');
20         }
21         const data = await response.json();
22         setProducts(data);
23         setLoading(false);
24       } catch (error) {
25         console.error("Failed to fetch products:", error);
26         setLoading(false);
27       }
28     }
29     fetchData();
30   }, []);
31   if (loading) {
32     return <div>Loading...</div>;
33   }
34
35   return (
36     <div>
```

[1:21 PM]Jamie:

the http response

[1:21 PM]Tyler:

all the login bools now return a https response instead now yeah

[1:22 PM]Tyler:

@Kiran go to the product controller file in the backend and add @CrossOrigin
above the method your tryin gto use

[1:22 PM]Ben:

Add @crossorigin in the controller I think

[1:23 PM]Kiran:

I'll try that

[1:24 PM]Kiran:

```
team-project-group-p03-08

JS ProductPage.js M ProductController.java 4, M X UserController.java Git Graph
backend > products > src > main > java > au > edu > rmlt > sept > supermarkets > products > controllers > ProductController.java > @P
import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.service.annotation.PutExchange;
import java.util.HashSet;
import java.util.HashSet;
You, 10 seconds ago | 3 authors (BenNippard and others)
@RestController
@RequestMapping(value = "/v1/products")
public class ProductController {
    private ProductService service;
    @Autowired
    public ProductController(ProductService service){this.service = service;}
    @CrossOrigin
    @GetMapping
    public HashSet<Product> all() {return service.getProducts();}
    @CrossOrigin
    @GetMapping("/{id}")
    public Product getProduct(@PathVariable int id) {
        return service.getProduct(id);
    }
    @CrossOrigin
    @GetMapping("/{price/{id}}")
    public HashSet<ItemPrice> getPrice(@PathVariable int id) {
        return service.getProductPrice(id);
    }
}
//Searching must take the form of /search?name=NAME&category=CATEGORY&supermarket=SUPERMARKET&price=PRICE
//Searching must take the form of /search?name=NAME&category=CATEGORY&supermarket=SUPERMARKET&price=PRICE
```

[1:24 PM]Kiran:


```
team-project-group-p03-08

ProductPage.js M ProductController.java 4, M X UserController.java Git Graph

end > products > src > main > java > au > edu > rmit > sept > supermarkets > products > controllers > J Pr

import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.service.annotation.PutExchange;

import java.util.HashSet;

You, 10 seconds ago | 3 authors (BenNippard and others)
@RestController
@RequestMapping(value = "v1/products")

public class ProductController {
    private ProductService service;

    @Autowired
    public ProductController(ProductService service){this.service = service;}

    @CrossOrigin
    @GetMapping
    public HashSet<Product> all() {return service.GetProducts();}

    @CrossOrigin
    @GetMapping("/{id}")
    public Product getProduct(@PathVariable int id) {
        return service.GetProduct(id);
    }

    @CrossOrigin
    @GetMapping("/price/{id}")
    public HashSet<ItemPrice> getPrice(@PathVariable int id) {
        return service.GetProductPrice(id);
    }

    //Searching must take the form of /search?name=NAME&category=CATEGORY&supermarket=SL
    //Sort by is either DEFAULT, NAME, CATEGORY, SUPERMARKET, PRICE
    //Where ORDER is either ASC or DESC
    //If a parameter is not required, leave it blank
    @GetMapping("/search")
    public HashSet<Product> search(
        @RequestParam("name")String name,

```

PROBLEMS 45 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS

WSP: DefaultHandlerExceptionRe

[1:25 PM]Tyler:

give it a go

[1:25 PM]Kiran:

It didn't work

[1:25 PM]Tyler:

your gonna have to restart the backend btw

[1:25 PM]Kiran:
Oh ok
[1:26 PM]Kiran:
Thanks guys
[1:26 PM]Kiran:
It fixed the problem
[3:25 PM]Tyler:
@Jamie ive done all my changes on the usermapping branch so you just merge when your done
@Tyler
@Jamie ive done all my changes on the usermapping branch so you just merge when your done
[3:26 PM]Jamie:
lovely
[3:26 PM]Jamie:
i'm probably gonna work on it tonight or tomorrow
[3:26 PM]Tyler:

[10:03 PM]Jamie:
help my commits are becoming more deranged as this project continues
[10:17 PM]Jamie:
@Tyler why is login a post request
[10:21 PM]Tyler:
as you need several parameters
[10:22 PM]Tyler:
i thought it was logical enough
@Tyler
as you need several parameters
[10:29 PM]Jamie:
wait so why doesn't GetMapping work
[10:29 PM]Tyler:
for getmapping to work the parameters need to be declared in the url
[10:29 PM]Tyler:
and you need several parameters for login
[10:30 PM]Tyler:
so i just made it post

September 24th:

[1:53 PM]Jamie:
@Tyler in the Get method, is ID just the email
[1:54 PM]Tyler:
which method?
[1:54 PM]Jamie:
get user by ID
[1:54 PM]Jamie:
i'm assuming ID is just email in this instance right
[1:54 PM]Tyler:
oh no sorry i need to make a method to do that
[1:54 PM]Tyler:

i forgot to make get user by email

[1:54 PM]Tyler:
one second

[1:58 PM]Tyler:
okay if you pull findByEmail should be there

[2:00 PM]Tyler:
also it is mapped to /email/{email}

[2:31 PM]Kiran:
Good news lads, The search results page images now link to the product page and show the correct information on each individual product page!

[2:31 PM]Kiran:
Still have some clean up to do, but I merged into main

[2:33 PM]Jamie:
@Tyler yo tyler can you change the method so that it returns not a user object but like

[2:33 PM]Jamie:
converts the user object to JSON?

[2:33 PM]Jamie:
or something i can read on js side?

[2:37 PM]Ben:
If it returns a user object it should be a json format

[2:37 PM]Ben:
Unless it's not being formatted as a response body maybe

[2:38 PM]Tyler:
im running thundclient tests on it rn and its not working

[2:38 PM]Tyler:
sorry i shouldve done this earlier

[2:40 PM]Tyler:
okay pull that and it should be working now sorry

[2:40 PM]Tyler:
i had a capital in the method variable but not in the mapping

@Ben
If it returns a user object it should be a json format

[2:43 PM]Jamie:
`const data = await retrieveUserData(emailInput);`
`console.log(data);`
console logging the data

[2:43 PM]Jamie:
`async function retrieveUserData(emailInput) {`
`const res = await fetch("http://localhost:8080/v1/users/Email/" + emailInput)`
`.then((result) => {`
`return result.json();`
`});`
`}`

[2:43 PM]Jamie:
returns undefined

[2:43 PM]Tyler:
undefined means it returned null and there is no corresponding email

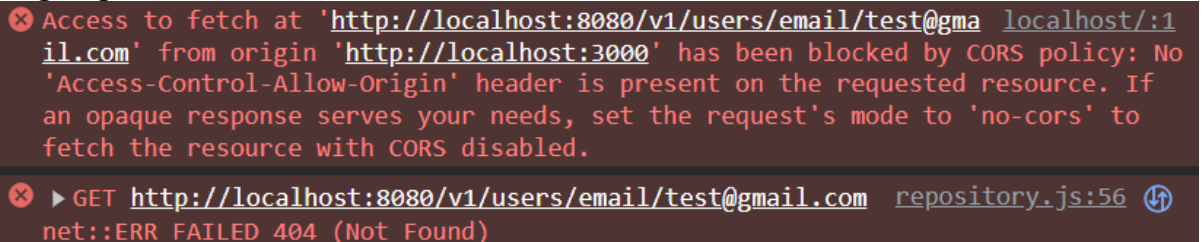
[2:43 PM]Jamie:
but it is recognising the email exists in database

[2:44 PM]Tyler:
and the e in email is lowercase

[2:44 PM]Tyler:
yours is uppercase

[2:45 PM]Jamie:
huh

[2:45 PM]Jamie:
im going to kill a bitch



```
✖ Access to fetch at 'http://localhost:8080/v1/users/email/test@gmail.com' from origin 'http://localhost:3000' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource. If an opaque response serves your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled.
✖ ▶ GET http://localhost:8080/v1/users/email/test@gmail.com repository.js:56
net::ERR_FAILED 404 (Not Found)
```

[2:45 PM]Tyler:

[2:45 PM]Tyler:
time to enable cors

@Tyler
time to enable cors

[2:51 PM]Jamie:
I'VE PUT @CROSSORIGIN ON EVERYTHING

[2:52 PM]Tyler:
did you restart the backend?


@Tyler
did you restart the backend?

[2:57 PM]Jamie:
going to go kashoot myself

[2:59 PM]Jamie:

```
async function retrieveUserData(emailInput) {
  const res = await fetch("http://localhost:8080/v1/users/email/" + emailInput)
  .then((result) => {
    console.log(result);
    return result.json();
  });
}
```

@Ben what do you mean it returns a JSON



```
repository.js:62
Response {type: 'cors', url: 'http://localhost:8080/v1/
▶ users/email/test@gmail.com', redirected: false, status:
200, ok: true, ...}
undefined loginModal.js:19
```

[3:01 PM]Tyler:
damn, it should just return a json

[3:01 PM]Tyler:
thats a bit odd

@Tyler
damn, it should just return a json
[3:04 PM]Jamie:
found a fix
[3:09 PM]Tyler:
lovely
[3:10 PM]Jamie:
gentlemen
[3:10 PM]Jamie:
its been a total of
[3:10 PM]Jamie:
12 hours
[3:10 PM]Jamie:
but
[3:10 PM]Jamie:
we have a functioning login system
[3:14 PM]Tyler:
yippee
[3:49 PM]Jamie:
@Tyler merge complete
[3:49 PM]Jamie:
im going to now destress
[3:49 PM]Tyler:
sounds good, good job jamie
[7:11 PM]Kiran:
@Jamie Well done
@Kiran
@Jamie Well done
[7:11 PM]Jamie:
Bonus, the rest of you can access said state
[7:11 PM]Jamie:
From the way ive set it up
[7:12 PM]Kiran:
Awesome

September 25th:

[9:51 AM]Ben:
@everyone Meeting today 12pm with Dale
[9:51 AM]Jamie:
On teams?
[9:51 AM]Ben:
probably
[11:56 AM]Jamie:
@everyone meeting in 4 min
[12:16 PM]Jamie:
should we just have a quick meeting right now
[12:16 PM]Jamie:
since everyones here
[12:16 PM]Andy:

ya
[12:17 PM]Ben:
@everyone meeting now
[4:37 PM]Tyler:
i forgot about the meeting
[4:37 PM]Tyler:
How did it go?
[5:47 PM]Andy:
ya was all good, we just had a brief presentation of the website and our progress (jamie's login and kiran's search page) and dale said that we were caught up in relation to the others groups
[6:15 PM]Tyler:

Lovely
[6:26 PM]Andy:
the gist from today's meeting

Information / Decisions	
No.	Item
1	Plan to complete functionality of website by end of this week
2	Last week of sprint (<u>i.e.</u> next week) will involve: acceptance testing, containerising application, porting to AWS?
3	Client wants to be able to browse and view <u>ALL</u> of the products (Products Page needs to be completed) (Kiran)
4	Search and Filter functionality still to be implemented (Ben & Peter)
5	Front-end testing (Jamie) + Integration testing and Continuous Integration still to be completed
6	Favourites functionality (Tyler) and Notifications functionality (Andy) still to be completed
7	Next meeting in person Thursday 28th Sept
8	Challenge 2 NEXT WEEK (Monday 2nd October 4:30pm – Tuesday 3rd October 4:30pm)

September 26th:

[12:49 AM]Kiran:

I finally got the cart functionality to work
[12:54 AM]Kiran:
But it saves the cart locally to a json file
[12:55 AM]Kiran:
Do we need to link the cart to an account?
[12:55 AM]Kiran:
Please tell me we don't have to do that
[10:42 AM]Tyler:
you kinda need to do that, the cart is already linked to an account by the customerid variable
[10:42 AM]Tyler:
so you just need to use the customer id to fetch the corresponding cart
[10:42 AM]Tyler:
and you will also use the customer id to put items into/ take out of carts

[10:45 AM]Kiran:
Ok thanks mate

[10:45 AM]Kiran:
I'll try do that

[10:46 AM]Tyler:
all of the necessary mapping should've already been pushed to main so you should be good

[12:01 PM]Tyler:
So for favourites I've made a new table which will need to be included in the relational diagram

[3:21 PM]Kiran:
Hey guys, does the login save when you leave to a different page?

[3:21 PM]Kiran:
Cause I'm struggling to stay logged when navigating to the other webpages

[3:22 PM]Tyler:
@Jamie ^ (edited)

@Kiran
Cause I'm struggling to stay logged when navigating to the other webpages

[3:32 PM]Jamie:
You need to import the context from app.js

[3:32 PM]Jamie:
And then consume the context

[3:32 PM]Jamie:
Check out the modal pages

[3:33 PM]Jamie:
The context should pretty much cover the entire front end application

[4:53 PM]Kiran:
Ok thx, I'll try that

[7:54 PM]Jamie:
@Ben under 09 - releasing docker image in aws there's this step where it's like
export REPO=403954307333.dkr.ecr.us-east-1.amazonaws.com/movies
docker build --platform linux/amd64 -t \$REPO .
where the am I supposed to do the command

[7:54 PM]Jamie:
I'm trying to do it in terminal

[7:55 PM]Jamie:
but export is not recognised

[7:56 PM]Ben:
I don't know about the docker stuff I'm not up to date with that

[7:57 PM]Peter:
Did you ask online?

[7:57 PM]Peter:
He should still be "teaching" it (edited)

Deploy your images to ECR

If you haven't yet, deploy your Docker images to ECR as described in [here](#)

[7:59 PM]Peter:
Point it out on the forum and in class and it may be fixed

September 29th:

[11:23 AM]Jamie:

[Presentation Guide-1.pdf](#)

286.52 KB

[11:23 AM]Jamie:

welp heres our guide for presentation

[11:23 AM]Jamie:

10 minutes

[11:24 AM]Jamie:

1-3 slides for product and project each

[11:24 AM]Jamie:

1 slide per person

[8:13 PM]Jamie:

good news

[8:13 PM]Jamie:

i've got front end testing working

[8:13 PM]Jamie:

and all it took was 5 hours of my sanity

September 30th:

[3:49 PM]Ben:

We probably need to have a meeting sometime this week I know we've got the challenge but there is a several things needed to be ticked off such as containerisation, acceptance testing, integration testing if we want and maybe some other things to do

[3:59 PM]Ben:

also if this challenge requires AWS im gonna be sad

[4:53 PM]Tyler:

yea sounds good

@Ben

We probably need to have a meeting sometime this week I know we've got the challenge but there is a several things needed to be ticked off such as containerisation, acceptance testing, integration testing if we want and maybe some other things to do

[7:41 PM]Kiran:

Mabye Tuesday evening, so we don't have to worry about the challenge. If that works for everyone

[8:00 PM]Kiran:

@everyone Does 8 pm work for you guys?

[8:00 PM]Ben:

Yeah

October 3rd:

[7:42 PM]Kiran:

@everyone meeting at 8:30

[8:35 PM]Andy:

- a PROJECT REPORT:
 - a "Vision" statement, i.e., a description of the why the product is "valuable" (about 1/2 page)
 - a figure showing the final basic system architecture / design --- this does not need to be a UML diagram: a box-and-lines sketch architecture is fine; focus on the tiers : front layers, backend (services and their layers), datastore and what components are in each
 - a report describing any refactoring, including refactoring to a Design Pattern, if appropriate, as well as the "smells" that you found (i.e., purpose for refactoring)
 - an overview of your Git organisation and how often you committed (include in report)
 - a description of your Scrum process: how often you met, who was Scrum Master (show the velocity of the team through all milestones for all sprints)
 - a diagram of your Deployment pipeline setup: i.e., CI/CD, steps in the pipeline, automation tools in the pipeline

[8:36 PM]Jamie:

@Tyler yo you available for meeting

[8:47 PM]Ben:

THINGS TO BE DONE Acceptance Testing - Ben Integration Testing - Tyler
 Project Report - Jamie Docker Containers - Ben Docker Compose ~Docker AWS
 Deployment - Leave till last~ Filtering - Peter Cart page - Kiran User page - Andy
 Update SRS - All
 Testing Plan - Tyler Sprint retro - Kiran New tasks and Adjusted/Removed tasks
 - Ben

[8:47 PM]Ben:

Benjamin Nippard 8:47 PM

THINGS TO BE DONE

- Acceptance Testing - Ben
- Integration Testing - Tyler
- Project Report - Jamie
- Docker Containers - Ben
- Docker Compose
- ~Docker AWS Deployment - Leave till last~
- Filtering - Peter
- Cart page - Kiran
- User page - Andy
- Update SRS - All
 - Testing Plan - Tyler
 - Sprint retro - Kiran
 - New tasks and Adjusted/Removed tasks - Ben

Red = Aim to Complete by Thursday

Green = Complete by submission

Purple = Complete if time permits/optional

October 4th:

[8:36 PM]Kiran:

@Tyler Hey mate, I was wondering if you could just go over the cart functionality in the backend and confirm it's working as intended

[8:36 PM]Kiran:

Cause I've tried adding the front end functionality but it's seems to really be struggling with the backend

[8:37 PM]Kiran:

Thanks mate

October 5th:

[12:57 PM]Tyler:

ill give it a check but we can sort it out in class

[1:21 PM]Kiran:

easy

[2:21 PM]Ben:

Ive left my docker containers running since Monday

[2:34 PM]Peter:

I'm going to need assistance with the filtering I'm lost

[8:25 PM]Peter:

i merged the filtering changes

[11:01 PM]Kiran:

Lads I got the products to display based on each users cart!!!!!!

[11:01 PM]Kiran:

I haven't finished the other functionality yet, but the hard part is done

October 6th:

[10:39 AM]Ben:

I've got the docker containers working I just can't get the react api url to work properly but it is now passing into the react app thankfully but everytime I use the url it times out so IDK but it works with hte local host variable

[11:42 AM]Kiran:

@Tyler @Jamie Hey guys, would you mind jumping in a call sometime today for a bit. I'm having trouble updating the backend for product quantity even though all the variables being passed are correct

[11:43 AM]Kiran:

I've added a lot of console debugging but still can't find the problem

[12:06 PM]Tyler:

yeah im free whenever

[12:07 PM]Kiran:

Would 3pm work for you guys?

[12:07 PM]Tyler:

sure

[12:07 PM]Kiran:

Thanks Tyler

[12:18 PM]Ben:

bruh I hate flyways so h2 console doesn't use mySQL because it wants to be differnt and special and so to change to mySQL I have to change my migrations to use hte mySQL commands for them to work but the tests still want to run in h2 so I either have to make them share the same language or to just swap fully to one or the other

[12:18 PM]Tyler:

probs easier to do a full swap aye

[12:18 PM]Tyler:

thats a bit of fun though

[12:19 PM]Ben:

well the mysql is in the docker containers so to change the tests to use mySQL I need to be running a server AHAHAHAHAH

[12:19 PM]Ben:

Im going to do the crappiest change because I know it will work and I can't be assed

[12:21 PM]Tyler:

if it works it works aye

[1:23 PM]Ben:

Okay i've got the compose working but without mySQL at the momemnt

[2:50 PM]Kiran:

@Tyler I'll be 15 mins late sorry

[2:58 PM]Tyler:

all g

[3:17 PM]Kiran:

@Tyler ready when you are

[3:19 PM]Ben:

Now docker compose is working with the mySQL database so I will merge my stuff with main it just might require some further changes to other peoples frontends

[3:56 PM]Ben:

@Andy your free to do docker actions with github now