Capstone Project Document

Project Title: Honey Buzz
Student Name: Saki Ajiro

Note: The following are the candidate sections of the document. They are presented here for guidance. Questions in each section could be used as possible aspects to cover. Some questions may not be applied to each project. On the other hand, additional information may be needed.

Introduction

Purpose

- What is the problem or the opportunity that the project is investigating?
    - The ecommerce industry is expected to continue growing as more and more customers find it handy to shop online and the business owners with a traditional brick-and-mortar store are facing needs to incorporate an online shop in their business as well as the fact that more and more startups want to sell products online.

- Why is this problem valuable to address?

    - Allow businesses and startups to sell their products more easily and customers can have more variety of products available online.

- What is the current state (e.g. unsatisfied users, lost revenue)?
    - Many small business/ sole traders are selling their product on Facebook or TradeMe without having their own online shops.

- What is the desired state?
        - Small businesses and startups can open their own online shops easily with no technical knowledge.

- Has this problem been addressed by other projects? What were the outcomes?
    - There is Shopfiy. Shopify gives an option to manage your business by adopting extension and customization. But to do these customization it requires a certain level of technical skills and knowledge and adopting a lot of extensions can be costly.

Industry/ domain

- What is the industry/ domain?

    - E-Commerce. For the demonstration purpose, I used honey products in this project. But it could be anything that the business owner wants to sell.

- What is the current state of this industry? (e.g. challenges from startups)
    - The ecommerce industry is expected to continue growing as more and more people find it handy to shop online and the business owners with the traditional brick-and-mortar store are facing needs to incorporate an online shop. They struggle to find a solution where they can easily use/ maintain their customer base and business logics.

- What is the overall industry value-chain?
    - Business buys products from suppliers or manufacture their own.
    - Post their products on the online shop, maintain the product data.
    - Create a lead to pull customers to the shop.
    - Get customers to sign up and maintain the user data.
    - User place an order online
    - Business deliver the products to the customer.
    - Stock deducted.

- What are the key concepts in the industry?
    - SMEs often cannot afford to stock up the large volume of products, hence, from selling products to top-up-stock cycle is shorter. So they need to keep a close eye on the inventory level to avoid out of stock situation where they would lose potential customers.

- Is the project relevant to other industries?
    - Yes, it can be used in any industry where an easy-to-run online shop is needed.

Stakeholders

- Who are the stakeholders? (be as specific as possible as to who would have access to the software)
    - SME business owners/ sole traders that are after an ecommerce integrated with simple and easy-to-use business management system.
    - Customers of the online shop who are after a handy online purchase.

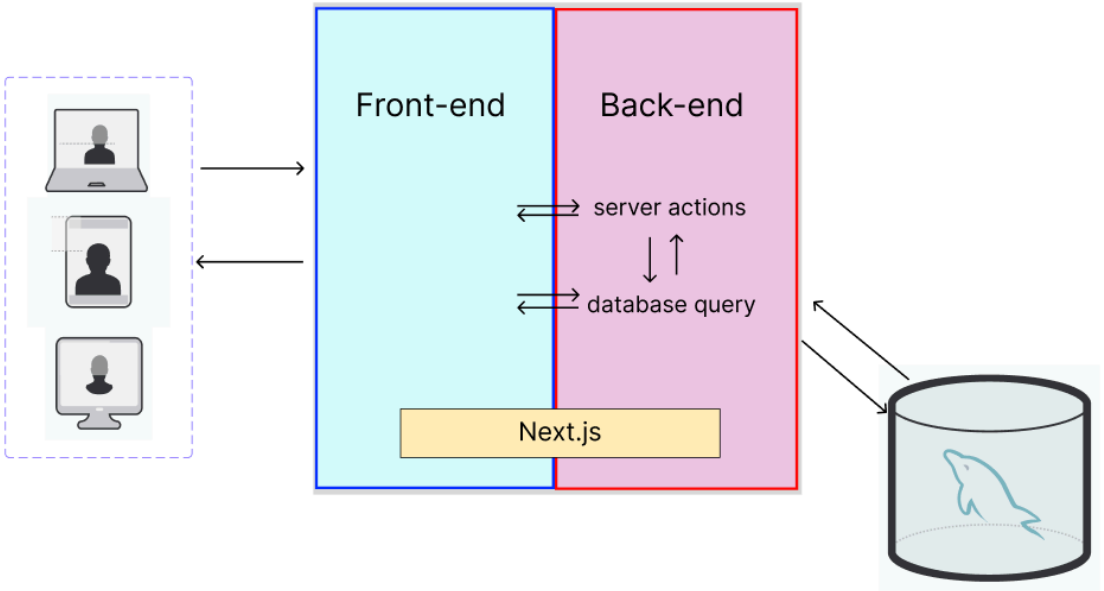
- Why do they care about this software?
    - Business owners: They need a platform to trade their product online 24/7.
    - Customers: They want to see more products available online for a handy online shopping experience.

- What are the stakeholders' expectations?
        - Business owners: They are after an ecommerce platform integrated with a simple and easy-to-use efficient business management system.
    - Customers: 24/7 availability

Product Description

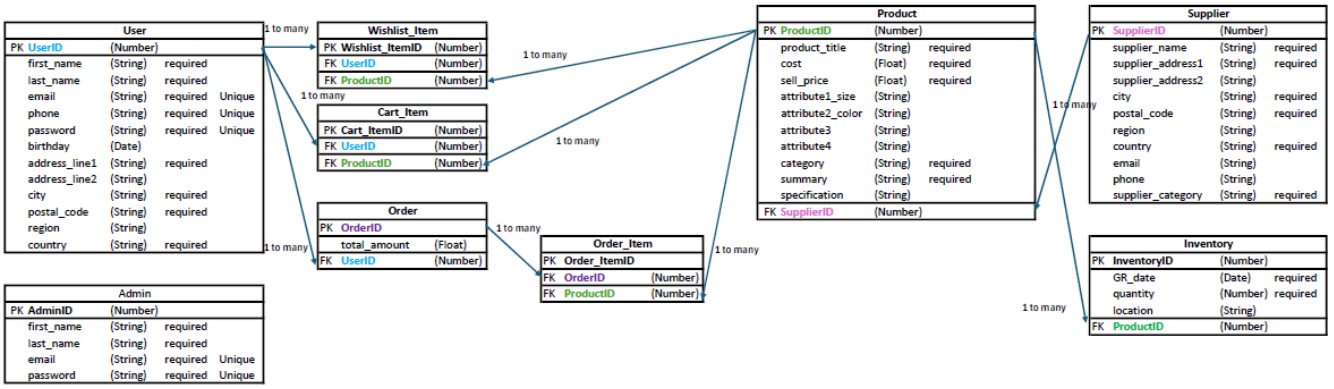
Architecture Diagram

Include a diagram of the building blocks of the design including users and how they interact with the product.



Database Modelling

<mySQL>



User Stories

<USER>

#	User Story Title	User Story Description	Priority	Additional Notes
1	Sign Up	As a user, I want to be able to sign up so that I can use the online shop.	high	
2	Login	As a user, I want to be able to login so that I can use my saved details when I come back to the website.	high	
3	Wishlist	As a user, I want to be able to add items in wishlist so that I can consider them later on.	high	
4	Shopping cart	As a user, I want to be able to add items in cart so that I can continue shopping and proceed to checkout once shopping is done.	high	
… 5	Shipping address	As a user, I want to be able to use my saved address to fill out shipping address so that I don't have to enter the same info twice.	high	

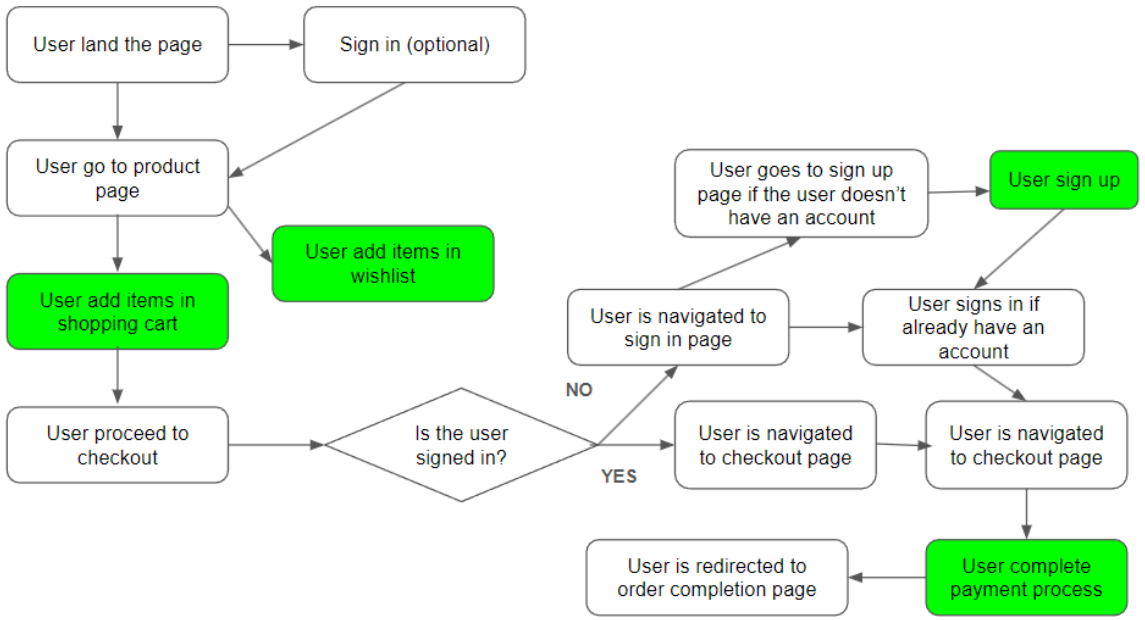
<ADMIN>

#	User Story Title	User Story Description	Priority	Additional Notes
1	Store business data	As a user, I want to be able to store products and supplier info in the database.	high	
2	Manipulate data	As a user, I want to be able to update/delete products and supplier info manually so that I can maintain the database up tp date.	high	
3	Inventory check	As a user, I want to be able to keep the stock level of each product in check so that I can top up the stock before they become out of stock.	high	

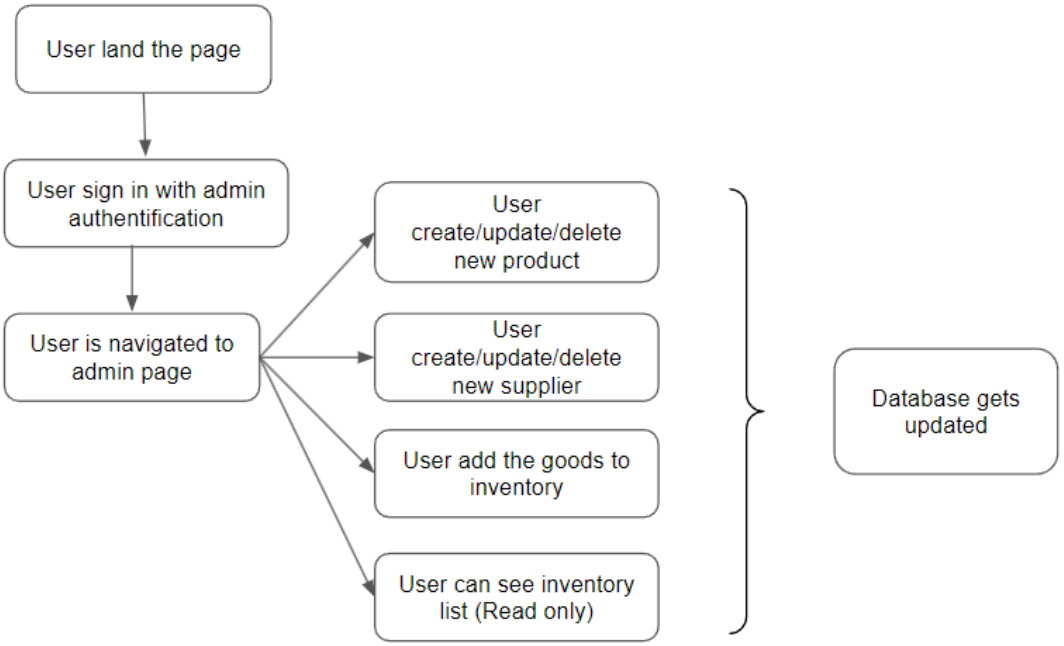
User Flow

Present as a flow diagram the steps a user may make in interacting with the software.

USER Perspective

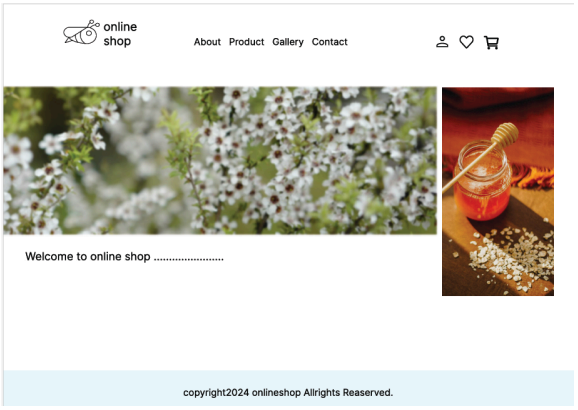


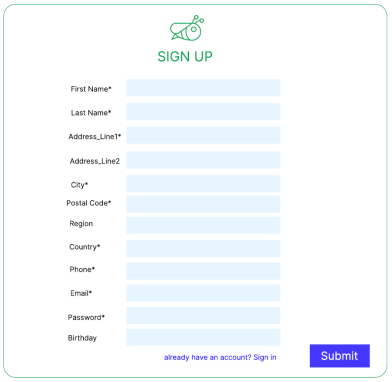
ADMIN Perspective

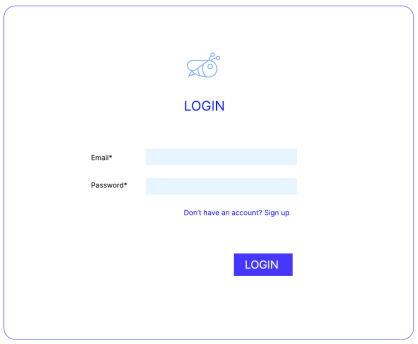


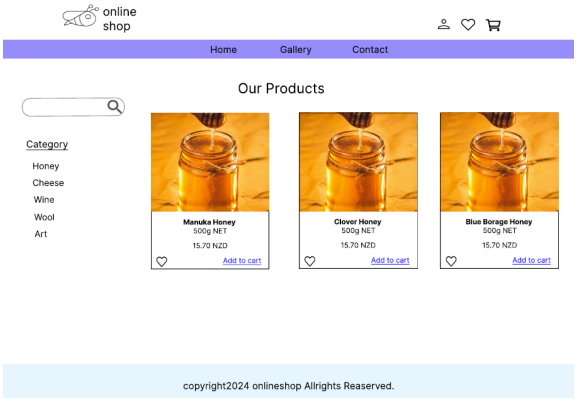
Wireframe Design

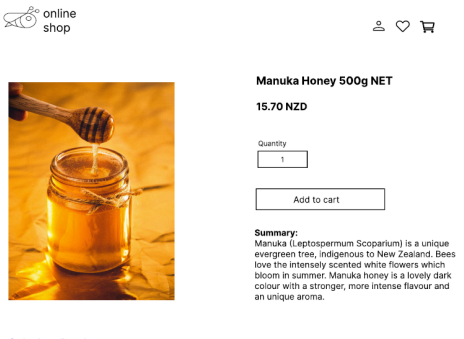
Show elements of the user interface, either manually or via a tool such as Figma.

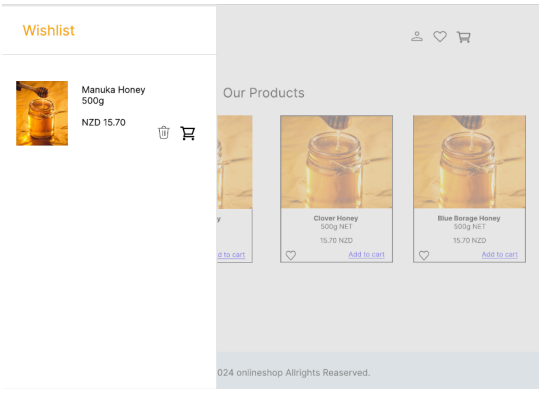

online shop
About Product Gallery Contact
Welcome to online shop .....
copyright2024 onlineshop Allrights Reaserved.

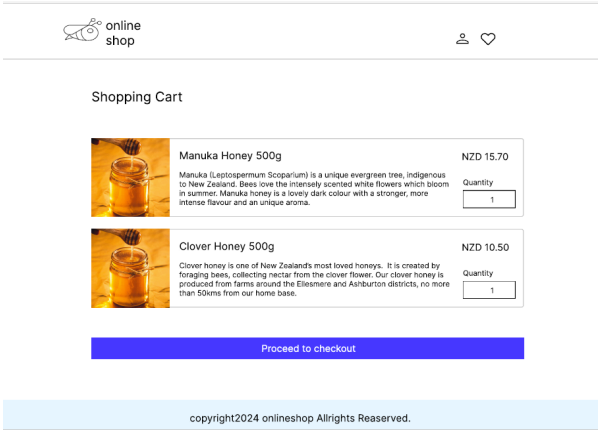

SIGN UP
First Name*
Last Name*
Address_Line1*
Address_Line2
City*
Postal Code*
Region
Country*
Phone*
Email*
Password*
Birthday
already have an account? Sign in
Submit

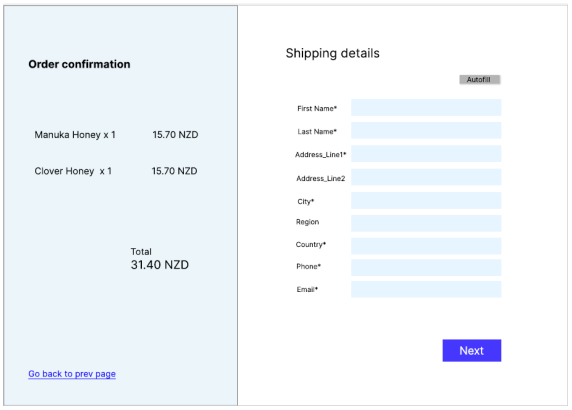

LOGIN
Email*
Password*
Don't have an account? Sign up
LOGIN

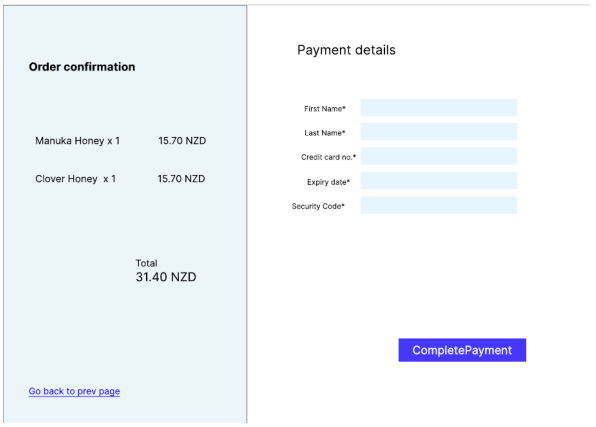

online shop
Home Gallery Contact
Our Products
Category
Honey
Cheese
Wine
Wool
Art
Manuka Honey
500g NET
15.70 NZD
Add to cart
Clover Honey
500g NET
15.70 NZD
Add to cart
Blue Borage Honey
500g NET
15.70 NZD
Add to cart
copyright2024 onlineshop Allrights Reaserved.

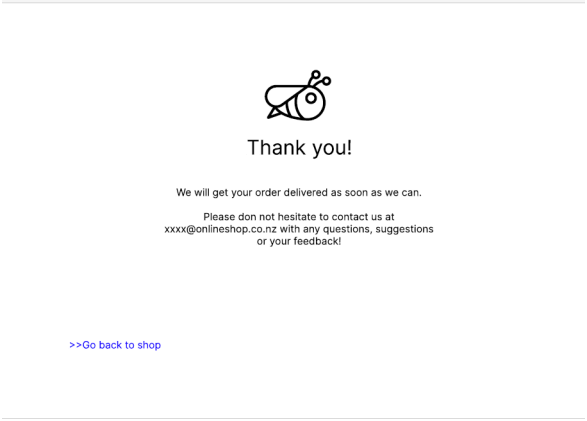

online
shop
Manuka Honey 500g NET
15.70 NZD
Quantity
1
Add to cart
Summary:
Manuka (Leptospermum Scoparium) is a unique evergreen tree, indigenous to New Zealand. Bees love the intensely scented white flowers which bloom in summer. Manuka honey is a lovely dark colour with a stronger, more intense flavour and an unique aroma.
Go back to all products


Wishlist
Manuka Honey
500g
NZD 15.70
Our Products
Clover Honey
500g NET
15.70 NZD
Add to cart
Blue Borage Honey
500g NET
15.70 NZD
Add to cart
024 onlineshop Allrights Reaserved.

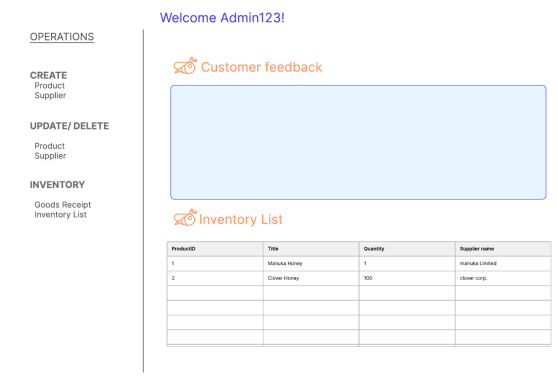

online
shop
Shopping Cart
Manuka Honey 500g
NZD 15.70
Manuka (Leptospermum Scoparium) is a unique evergreen tree, indigenous to New Zealand. Bees love the intensely scented white flowers which bloom in summer. Manuka honey is a lovely dark colour with a stronger, more intense flavour and an unique aroma.
Quantity
1
Clover Honey 500g
NZD 10.50
Clover honey is one of New Zealand's most loved honeys. It is created by foraging bees, collecting nectar from the clover flower. Our clover honey is produced from farms around the Ellesmere and Ashburton districts, no more than 50kms from our home base.
Quantity
1
Proceed to checkout
copyright2024 onlineshop Allrights Reaserved.

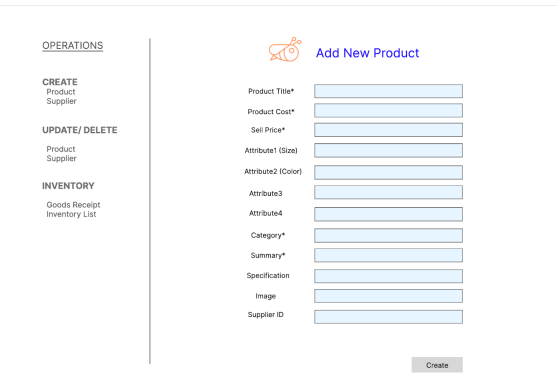

Order confirmation
Manuka Honey x 1
15.70 NZD
Clover Honey x 1
15.70 NZD
Total
31.40 NZD
Go back to prev page
Shipping details
Autofill
First Name*
Last Name*
Address_Line1*
Address_Line2
City*
Region
Country*
Phone*
Email*
Next

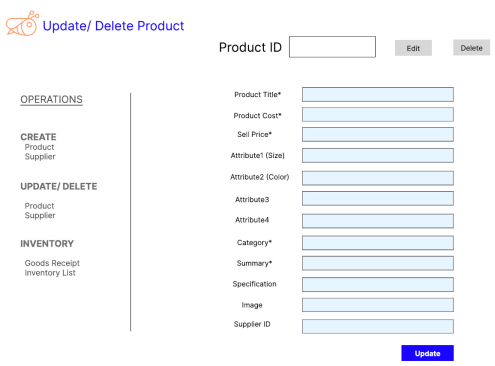

Order confirmation
Manuka Honey x 1
15.70 NZD
Clover Honey x 1
15.70 NZD
Total
31.40 NZD
Go back to prev page
Payment details
First Name*
Last Name*
Credit card no.*
Expiry date*
Security Code*
CompletePayment

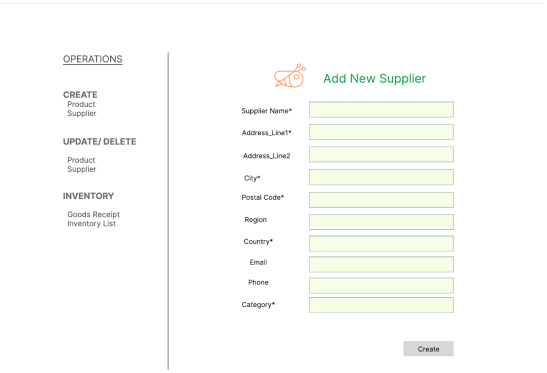

Thank you!
We will get your order delivered as soon as we can.
Please don not hesitate to contact us at
xxxx@onlineshop.co.nz with any questions, suggestions
or your feedback!
>>Go back to shop

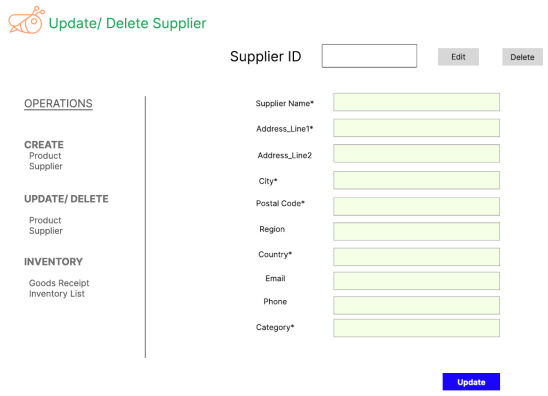
<admin pages>

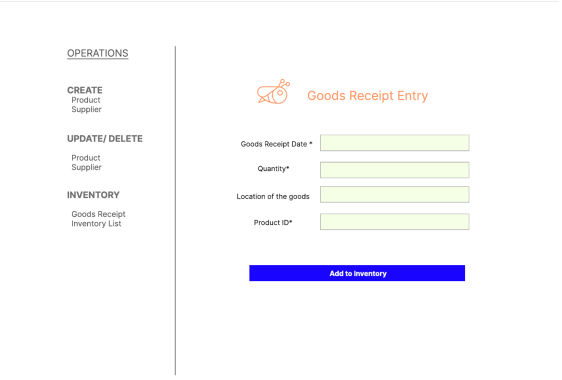

OPERATIONS
CREATE
Product
Supplier
UPDATE/ DELETE
Product
Supplier
INVENTORY
Goods Receipt
Inventory List
Welcome Admin123!
Customer feedback
Inventory List
ProductID
Title
Quantity
Supplier name
1
Manuka Honey
1
manuka Limited
2
Clover Honey
100
clover corp.


OPERATIONS
CREATE
Product
Supplier
UPDATE/ DELETE
Product
Supplier
INVENTORY
Goods Receipt
Inventory List
Add New Product
Product Title*
Product Cost*
Sell Price*
Attribute1 (Size)
Attribute2 (Color)
Attribute3
Attribute4
Category*
Summary*
Specification
Image
Supplier ID
Create









Open Questions/Out of Scope

- What features are considered out of scope?
    - Proper payment transaction
    - User profile management page
    - Customer review functionality
    - Deployment

Non-functional Requirements

- What are the key security requirements? (e.g. login, storage of personal details, inactivity timeout, data encryption)
    - Out of scope
- How many transactions should be enabled at peak time?
    - Initially expect 100, if the website became more popular it is expected to be increased up to 500.

- How easy to use does the software need to be?
    - It has to be very easy as it intends to be used by ordinary users (both shoppers/ business owners) with no technical software knowledge.

- How quickly should the application respond to user requests?
    - As quickly as possible to serve better user experience.

- How reliable must the application be? (e.g. mean time between failures)
    - As it deals with and holds a lot of sensitive information, the security must be highly reliable. Also the order/ inventory system has to be very reliable to ensure the smooth business operation. Ideally, the uptime for the shopping platform should be close to 100% but less than 5% of downtime would not cause critical issues.

- Does the software conform to any technical standards to ease maintainability?
    + Data normalisation
    + Github version control
    + Folder structure (Database has its own separate folder, server actions and database queries are stored in a separate ‘backend’ folder.)

Project Planning

Include a Gantt chart or screenshot of a Trello board showing key milestones (with dates) to complete the project.

Project planning	Task	week 1 3 -5/Jul	week 2 8 - 12/Jul	week 3 15 - 18/Jul	19-Jul
Documentation	Fill up template document			18-Jul	Demo
	Application flow design		9-Jul		
	Architecture Diagram			18-Jul	
	User Story		10-Jul		
	Hi-Fi(ish) design(Wireframe)		9-Jul		
	Database logical/physical model	3-Jul			
	Testing steps			18-Jul	
	ReadMe			18-Jul	
	Slide			18-Jul	
	Next.js set up	4-Jul			
	Sign up/ Login page/ user data creation		11-Jul		
	Header, Footer and Navigation	4-Jul			
	Wishlist (drawer)		11-12/Jul		
	Landing page			15-Jul	
	Product list page		11-Jul		
	Product detail page		11-12/Jul		
	Shopping cart page displaying all the products in the cart with ability for users to change quantities and delete items. Button to allow users to proceed to checkout from here.		12-Jul		
	Checkout page with autofill functionality for those who have registed the delivery address.		12-Jul		
	Payment page		14-Jul		
	Thank you page/ empty cart and move the data to order		14-Jul		
	CSS for all pages			15-Jul	
	Responsive			15-Jul	
	Create admin reg page		13-Jul		
	Create product & supplier page		10-Jul		
	Update/Delete product & supplier page		10-Jul		
	Goods receipt entry page		10-Jul		
	Inventory list		10-Jul		
Back-end	CRUD operations / server actions		10-16/Jul		
Database	Database table migrations	5-Jul			
	MySQL connection with Next.js		9-Jul		

Testing Strategy

- What were steps undertaken to achieve product quality?
    - Used comments to maintain the clarity of the code
    - Used/ tested the features/ functionalities from a user perspective.
    - Ensured the UI matches (or close enough to) the Figma Hi-Fi design.

- How was each feature of the application tested?

- By using console.log to ensure variables are getting the expected values from server actions, context, etc.

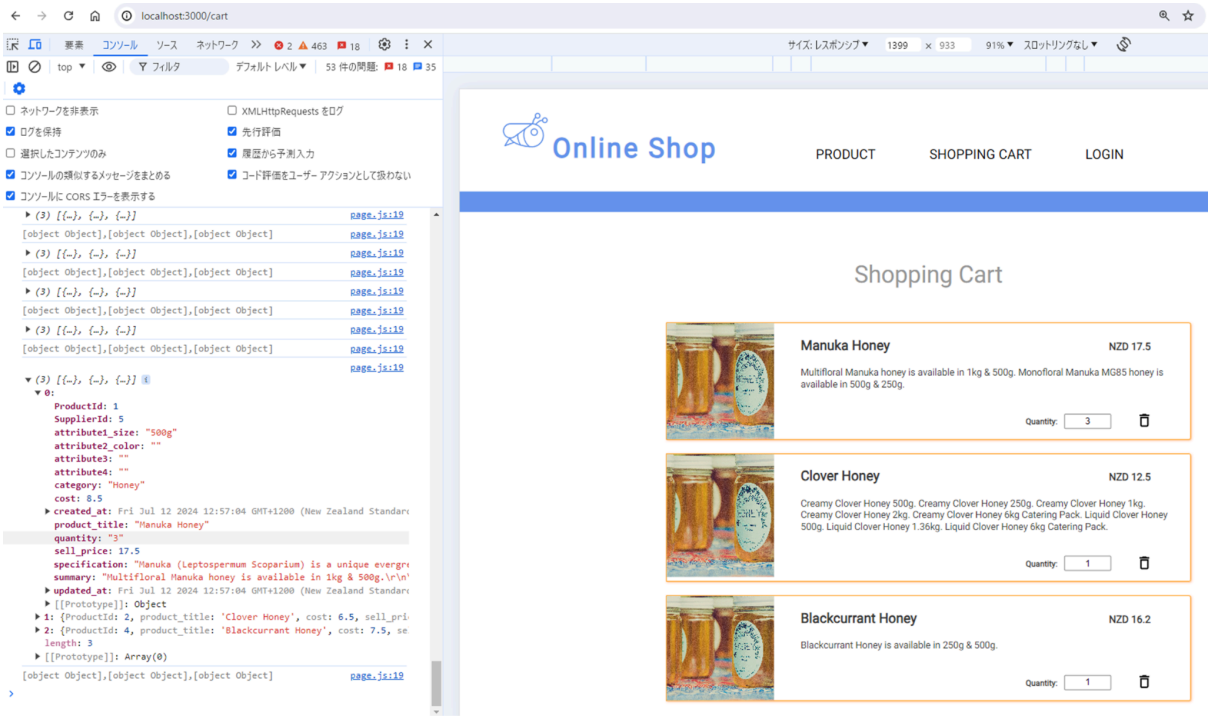
- How did you handle edge cases?
  - By using console.log and developer tools in the browser. The following screenshots demonstrate the edge case where I used console.log to ensure the quantity property appended to each product in the shopping cart context on line 32 is reflected in the context properly and updateQuantity function on line 47 does update the quantity according to user input as expected. (logging in line 19). This quantity held in the context is important as the same will be used to deduct the quantity from the inventory table in the database in the following transaction.

```
15  export default function ShoppingCartPage() {
16    const [productList, setProductList] = useState([])
17    const {loginDetails} = useContext(LoginContext)
18    const {cartContent, handleChange} = useContext(ShoppingCartContext)
19    console.log(cartContent)
20
21
22    useEffect(()=>{
23
24      async function getData() {
25        const cartItemData = await getCartItems()
26        const productData = await getProducts()
27
28        const filteredCartItems = cartItemData.filter((item)=>item.UserId===loginDetails.UserId)
29        const productsInCart = productData.filter((product)=>filteredCartItems.find((item)=>item.ProductId===product.ProductId))
30
31        //appending quantity property to each product in cart and setting the default value as 1
32        const product = productsInCart.map((product)=> ({...product, quantity:1}))
33
34        setProductList(product)
35        handleChange(product)
36      }
37      getData()
38    },[])
39
40    const deleteCard = (ID) => {···
44    }
45
46    //function to update the quantity with user input value which is passed from the CartItemCard component
47    const updateQuantity = (ID, quantity) => {
48      const items = [...productList]
49      items.find((product) => product.ProductId === ID).quantity=quantity
50      setProductList(items)
51      handleChange(items)
52    }
```

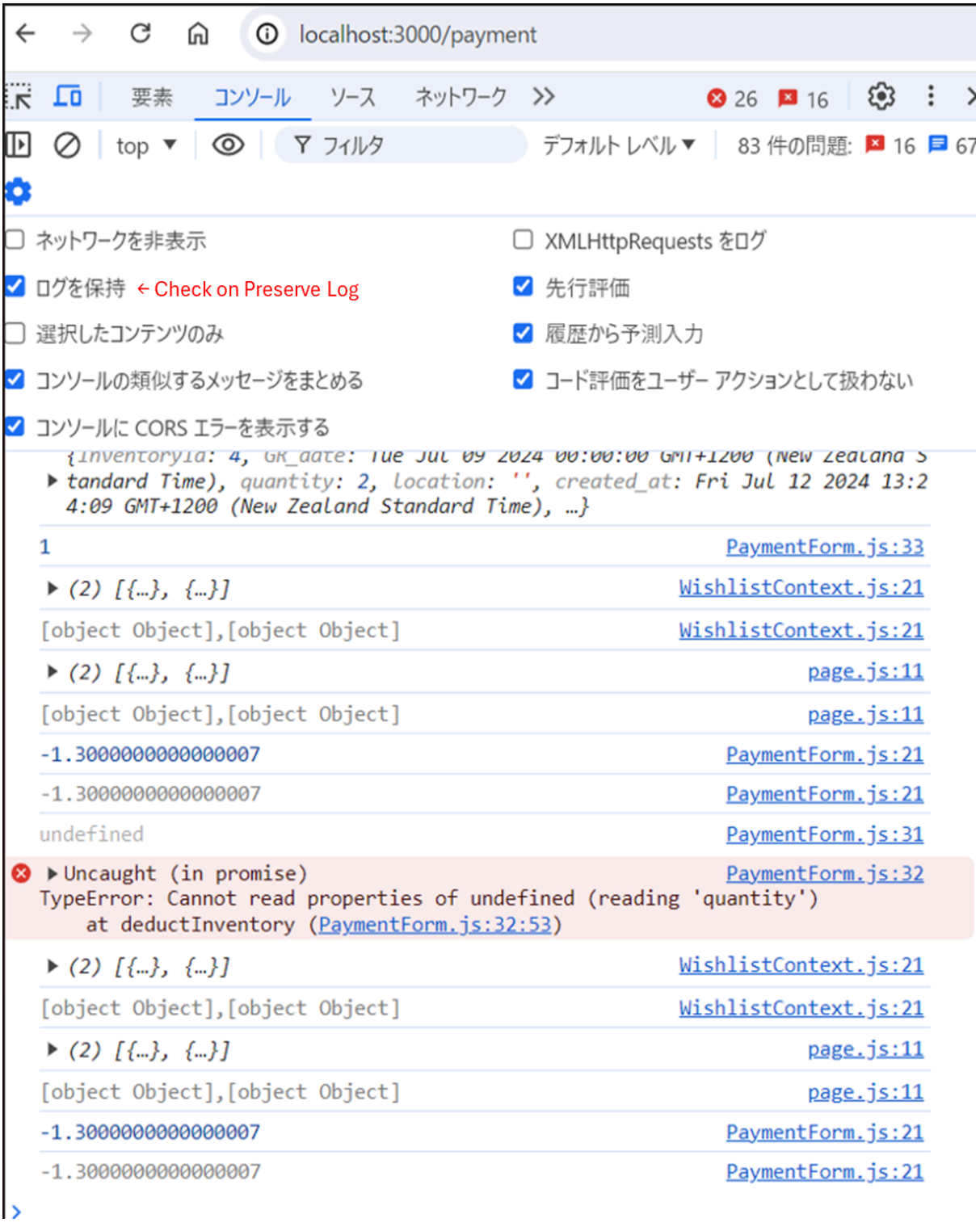
updateQuantity function is called in the following component.(line 15)

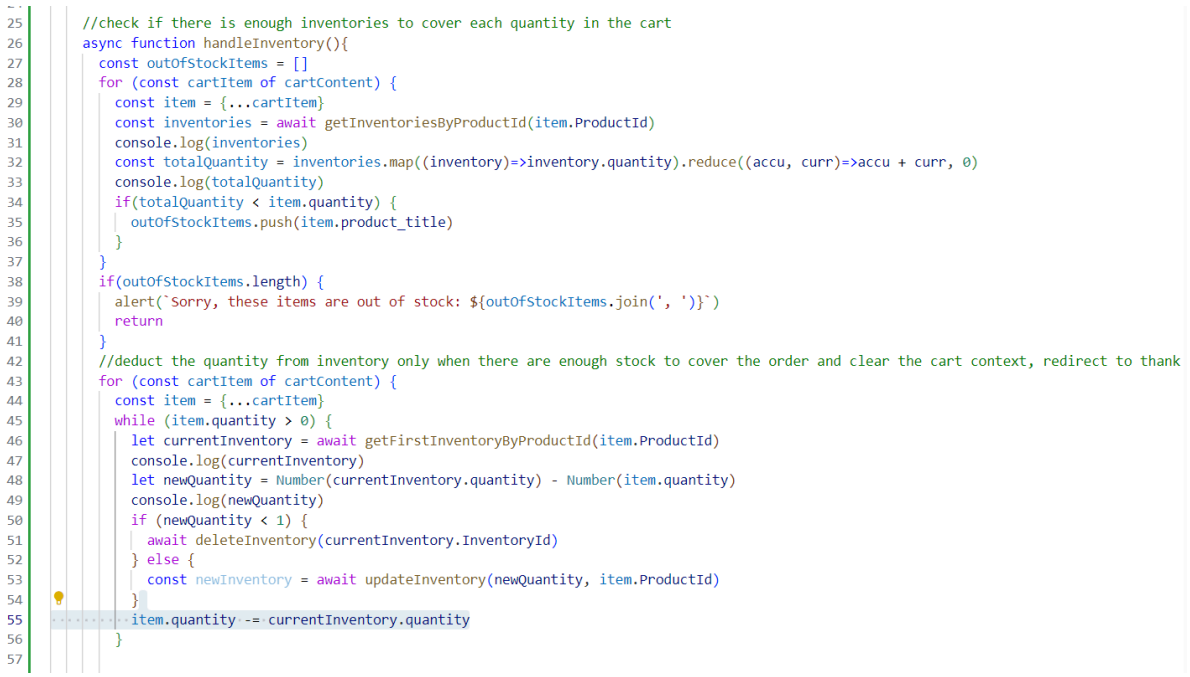
```
9   export default function CartItemCard({product, deleteCard, updateQuantity}) {
10    const {loginDetails} = useContext(LoginContext)
11
12    // Change quantity in input and save it in context
13    function handleQuantityChange(e) {
14      e.preventDefault()
15      updateQuantity(product.ProductId, e.target.value)
16    }
17
```

The following is the result in the browser console.



-The following is another edge case where the error has been detected when redirecting from Payment page to Thankyou page upon completion. The payment completion triggers inventory deduction and redirect user to the Thankyou page. This error occurred when there was not enough inventory for one of the product in the database. So I fixed my code not to deduct anything if any of the items cannot be fulfilled with quantity in the inventory.


localhost:3000/payment
要素
コンソール
ソース
ネットワーク
26
16
top
フィルタ
デフォルト レベル
83 件の問題:
16
67
ネットワークを非表示
XMLHttpRequests をログ
ログを保持 ← Check on Preserve Log
先行評価
選択したコンテンツのみ
履歴から予測入力
コンソールの類似するメッセージをまとめる
コード評価をユーザー アクションとして扱わない
コンソールに CORS エラーを表示する
tandard Time), quantity: 2, location: '', created_at: Fri Jul 12 2024 13:24:09 GMT+1200 (New Zealand Standard Time), …}
1 PaymentForm.js:33
(2) [{…}, {…}] WishlistContext.js:21
[object Object],[object Object] WishlistContext.js:21
(2) [{…}, {…}] page.js:11
[object Object],[object Object] page.js:11
-1.3000000000000007 PaymentForm.js:21
-1.3000000000000007 PaymentForm.js:21
undefined PaymentForm.js:31
Uncaught (in promise) PaymentForm.js:32
TypeError: Cannot read properties of undefined (reading 'quantity')
at deductInventory (PaymentForm.js:32:53)
(2) [{…}, {…}] WishlistContext.js:21
[object Object],[object Object] WishlistContext.js:21
(2) [{…}, {…}] page.js:11
[object Object],[object Object] page.js:11
-1.3000000000000007 PaymentForm.js:21
-1.3000000000000007 PaymentForm.js:21



```
//check if there is enough inventories to cover each quantity in the cart
async function handleInventory(){
  const outOfStockItems = []
  for (const cartItem of cartContent) {
    const item = {...cartItem}
    const inventories = await getInventoriesByProductId(item.ProductId)
    console.log(inventories)
    const totalQuantity = inventories.map((inventory)=>inventory.quantity).reduce((accu, curr)=>accu + curr, 0)
    console.log(totalQuantity)
    if(totalQuantity < item.quantity) {
      outOfStockItems.push(item.product_title)
    }
  }
  if(outOfStockItems.length) {
    alert(`Sorry, these items are out of stock: ${outOfStockItems.join(', ')}`)
    return
  }
  //deduct the quantity from inventory only when there are enough stock to cover the order and clear the cart context, redirect to thank
  for (const cartItem of cartContent) {
    const item = {...cartItem}
    while (item.quantity > 0) {
      let currentInventory = await getFirstInventoryByProductId(item.ProductId)
      console.log(currentInventory)
      let newQuantity = Number(currentInventory.quantity) - Number(item.quantity)
      console.log(newQuantity)
      if (newQuantity < 1) {
        await deleteInventory(currentInventory.InventoryId)
      } else {
        const newInventory = await updateInventory(newQuantity, item.ProductId)
      }
      item.quantity -= currentInventory.quantity
    }

```

Implementation

- What were the considerations for deploying the software?
    - Out of scope as it wasn't required. Before deployment of this application, it would need more vigorous security as the application holds a lot of sensitive information. If it is all secure and not out of scope, I would use Vercel to deploy this application.

End-to-end solution

- How well did the software meet its objectives?
    - It does have the backbone and foundation to keep building up on with a lot of useful and practical functionality, however, in reality, it does require more vigorous security to protect user data, business data before it becomes usable. Also, a secure payment transaction is required.

References

- Where is the code used in the project? (link to GitHub)
  https://github.com/Saki-1003/Capstone-Project

- What are the resources used in the project? (libraries, APIs, databases, tools, etc)
  Visual Studio Code

MySQL
Iconify

```
"@emotion/react": "^11.11.4",
"@emotion/styled": "^11.11.5",
"@mui/icons-material": "^5.15.21",
"@mui/material": "^5.15.21",
"dotenv": "^16.4.5",
"mysql2": "^3.10.2",
"next": "14.2.4",
"react": "^18",
"react-dom": "^18",
"sequelize": "^6.37.3",
"sequelize-cli": "^6.6.2"
```