



University
of Glasgow | School of
Computing Science

Honours Individual Project Dissertation

MySUPERTROLLEY: DELIVERING MOBILE RESPONSIVE WEB APPLICATION FOR ONLINE GROCERY SHOPPING FROM MULTIPLE SHOPS IN A SINGLE BASKET

Nikolas Finiotis
April 8, 2021

Abstract

The idea of purchasing groceries online, from multiple shops simultaneously within in a single order has never been implemented by any grocery shop website. Current systems and websites are comparing shop prices or ordering grocery items from multiple shops but not in a single order. MySuperTrolley is a new solution that gives the opportunity for users to easily navigate through an application without the need of much of technical experience, compare prices of items from multiple shops and order items from multiple shops in a single order. Requirements were gathered from background research and already existing related applications, as well as prioritising these through the use of a survey according to the participants responses. Various prototypes were created and using A/B testing, the most voted prototypes were selected to be implemented in MySuperTrolley application. MySuperTrolley is a mobile responsive web application which provides several features including log-in, sign-up, check for availability of delivery in users location, compare prices, select items for categories, filter for subcategories, item quantity, favourites, past orders, offers, alter basket items, order preview and checkout process. Following manual unit testing and user evaluation, the application was found to be "excellent" in usability by calculating the System Usability Scale (SUS) score across 37 participants, which gave a total average value of 91.4/100. Future enhancements were found in user evaluation that could further improve user satisfaction and allow for more consumers to consider using MySuperTrolley.

Acknowledgements

I would like to thank my supervisor, Dr Mireilla Bikanga Ada, for her invaluable support and guidance throughout the project. I would also like to thank the user evaluation participants for their time and contribution, which was extremely valuable and helpful.

Most importantly, I would like to thank my parents who supported me throughout this four years of my undergraduate degree and who also tried to push me to achieve everything that I manage to do. Additionally, I would like to thank my friends and girlfriend who were always there for me through all the difficulties faced.

Education Use Consent

I hereby grant my permission for this project to be stored, distributed and shown to other University of Glasgow students and staff for educational purposes. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Signature: Nikolas Finiotis Date: April 8, 2021

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Aims	1
1.3	Chapter Structure	2
2	Background	3
2.1	Background research	3
2.1.1	Why do people shop online?	3
2.1.2	Habitual categories bought from the internet	5
2.1.3	Obstacles to online shopping	5
2.1.4	Website design & features	6
2.1.5	Customer service	6
2.1.6	Summary	7
2.2	Pre-existing Related Applications	8
2.2.1	Trolley (Web App)	8
2.2.2	Buymie (Android & iOS Mobile App)	9
2.2.3	Devo (Web App, Android & iOS Mobile App)	10
2.2.4	Summary	11
3	Requirements Analysis	12
3.1	Requirement Process	12
3.2	IDEAS	12
3.3	User Personas	12
3.4	User Stories	13
3.4.1	Grocery User	13
3.4.2	Admin/Manager User	14
3.5	Survey	14
3.6	Final Requirements	16
3.6.1	Functional Requirements	16
3.6.2	Non-functional Requirements	17
4	Design	18
4.1	Agile Scrum Development Methodology	18
4.1.1	Stand-Up Meetings	18
4.1.2	Version Control	19
4.1.3	Iterative Development	19
4.2	System Architecture	19

4.3	ER Diagram	20
4.4	Activity Diagram	21
4.5	Prototypes	22
4.5.1	Paper Prototypes	22
4.5.2	A/B Testing Survey	23
4.5.3	Wireframes	23
4.6	Tools and Technologies	24
4.6.1	Backend - Maven Spring Boot MVC	24
4.6.2	Frontend - React	25
4.6.3	Database - MySQL	25
5	Implementation	26
5.1	Postcode Validation	26
5.2	Price Comparison	27
5.3	Features for people with limited technology skills	28
5.4	Item Quantity	29
5.5	Order Preview	30
5.6	Checkout process	31
5.7	Mobile Responsive	32
5.8	Deployment	32
5.9	Demo Presentation video	33
6	Evaluation	34
6.1	Unit Testing	34
6.2	User Evaluation	35
6.2.1	Methodology	35
6.2.2	Tasks	36
6.2.3	Multiple Choice Questions - Quantitative Information	37
6.2.4	Open Ended Questions - Qualitative Information	38
6.3	Evaluation Summary	39
7	Conclusion	40
7.1	Summary	40
7.2	Future work	40
7.3	Reflection	41
Appendices		42
A	Ethics Checklist	42
B	Comparison of existing similar Applications	44
C	IDEAS	45
D	User Personas	47
E	User Stories	49
F	Survey - Requirements	50
G	Agenda	51
H	ER-diagram	53

I	Activity-Diagram	54
J	Paper-Prototypes	55
K	A-B Testing	60
L	Wireframes	61
M	Unit Testing	62
N	User Evaluation - Survey	63
O	SUS Score	63
P	Requirements Validation	64
	Bibliography	65

1 | Introduction

This chapter introduces MySuperTrolley project, explaining the motivation and aims behind the development of this application. The chapter then concludes with an overview of this paper's subsequent chapters and contents.

1.1 Motivation

Over the past few years, the rapid increase of technology and the Internet has impacted our daily lives. Online shopping was also a majorly influenced aspect of our lives as it has increasingly overtaken in-store shopping, especially due to the current circumstances of COVID-19. According to Office for National Statistics (2019), only 2.8% of the total retail sales were attributed to internet sales across the UK in November 2006, whereas by November 2019 the percentage was 10 times more, up to 21.6%. COVID-19, led to people having to face a period of isolation and uncertainty, leading to overnight changes in shopping behaviours and an exponential increase in overall online sales. By November 2020, online shopping achieved 36.2% of the total retail sales.

In regards to grocery shopping, consumers are required to choose a specific supermarket for their online purchases limiting their ability to compare prices and order from multiple stores at once. Customers that want products from various stores are required to visit each individual website and place separate orders which can be laborious and tiring. Lack of Internet experience, minimum order value and additional delivery costs, deter and frustrate consumers which may end up abandoning their online grocery shopping. Thereby, MySuperTrolley app is a platform which will fill the aforementioned limitations and bridge gaps from other applications, by allowing consumers to shop from multiple stores simultaneously, satisfying their online grocery shopping needs as well as making it easier for less tech-savvy consumers.

1.2 Aims

This project aims to create an application which will support these three main objectives:

- Simple and easy navigating app where consumers with various levels of technological experience can use it.
- Compare product prices across various grocery shops.
- Order items from multiple shops in a single basket.

The main aim of this application is to satisfy consumers' online grocery shopping needs. Consumers will be able to check if the delivery services are available in their area before continuing any further, while also being aware of the supermarkets available to purchase from. Even if the consumers' location is unavailable for delivery, consumers can have the option to collect their order from store or use the website to compare item prices across different supermarket stores.

MySuperTrolley will provide information about the platform trying to convince consumers to select the website instead of others. By focusing on the older consumers with limited technological experience, which are advised by the government to remain indoors due to COVID-19, easy navigation and access to some features such as adding items to favourites and having access to past orders can attract them to purchasing their groceries from MySuperTrolley.

1.3 Chapter Structure

This chapter summarized the motivation and aims behind the development of MySuperTrolley. The remaining chapters will describe how the motivation of MySuperTrolley was transformed into a functionally working application and how the aims were successfully met. The structure of the remaining chapters is given below:

- **Chapter 2:** Background research of online shopping behaviour over the past 20 years in three different phases: early 2000s, 2010s and during COVID-19. Already existing applications related to MySuperTrolley will also be compared, identifying prominent features, strengths and weaknesses.
- **Chapter 3:** This chapter will outline the functional and non-functional requirements analysis process constructed from ideas, creation of user personas, user stories, and a conducted survey.
- **Chapter 4:** The design process will be discussed by explaining the software development methodology including agile development. Also, the system architecture, ER-diagram, activity flow diagram, tools and technologies to make use of and construct the user interface as well as prototypes (Paper Prototypes, Wireframes) will also be discussed.
- **Chapter 5:** The implementation of MySuperTrolley will be reviewed, highlighting the various implemented features.
- **Chapter 6:** The penultimate chapter will provide an evaluation of the system, including unit testing, user evaluation using survey and requirement validation.
- **Chapter 7:** The final chapter will summarise the paper, discussing any future implementation which could be carried out while also providing an overall reflection of the project.

2 | Background

This chapter will discuss the background research conducted on online shopping behaviours over the past 20 years, in three different phases: early 2000s, 2010s and during COVID-19 pandemic. The chapter will also compare the features, strengths and weaknesses of pre-existing related applications. These pre-existing applications will be compared due to their similarity in goals and functionalities to what is needed for MySuperTrolley, forming the foundations on which the system is built upon.

2.1 Background research

Online shopping was not accessible by most people, in the early 2000s. Ahuja et al. (2003) talk about advantages of online shopping. The authors posit that consumers perceive the advantages of online shopping being more convenient, consumers having a wider selection of products, having privacy when shopping as well as receiving more information faster than when they are in store. Alternatively during the 2010s, internet shopping was widely accepted as a purchasing platform of products and services and has become widely popular ever since (Bourlakis et al. 2008). According to Lim et al. (2016) the exponential sales within the Internet, proposed an attractive market for many companies to thrive into. Many industry giants like Alibaba, Tenecent, Amazon and Groupon have set an example of what consumers should consider as being benefits of online shopping, which in conjunction with the tremendous technological development in recent years, has driven the success of the industry even further (Wu et al. 2018). Both the traffic generated from online shoppers as well as money spend within the online platforms rose steadily in the past decade (Wu et al. 2018). During the COVID-19 pandemic, Prashar and Mohali (2020) state that online grocery consumers face numerous challenges, with the most threatening one being lack of consumer experience. Grocery shopping has become a routine consumer behavior as it became habitual, requiring less cognitive effort as a result, has become automatic and less time consuming in comparison to in-store shopping (Prashar and Mohali 2020). Prashar and Mohali (2020) who investigated consumer behavior, found that most consumers switch to online shopping due to discounts, incentives and offers available to them. Grocery owners must take this into consideration for future success of their businesses.

On the other hand, disadvantages do exist that may deter consumers from online grocery shopping. Some of the reasons affecting online shopping behaviours will be discussed in the next sections.

2.1.1 Why do people shop online?

There are various reasons why people shop online. Convenience, avoiding checkout lines and interacting with salespeople, including better prices and saving time, as shown in Table 2.1, where observed to be the top three reasons persistent across all three different eras (Ahuja et al. 2003; Katawetawarks and Wang 2011; Prashar and Mohali 2020). Moreover, starting from the 2010s, comparison of prices between multiple providers, the availability of smart phones that led to people having access to information on the go in conjunction with widespread internet

access globally, were pivotal reasons that attracted increasing amount of people to consider online shopping (Katawetawarks and Wang 2011).

Reasons people shop online in three eras			
↓ Reasons/Era →	2000s	2010s	COVID-19
Convenience (avoid checkout lines/sales person)	✓	✓	✓
Better prices	✓	✓	✓
Saving time	✓	✓	✓
Compare prices between multiple providers		✓	✓
Availability of smart phones and wide internet		✓	✓

Table 2.1: Reasons people shop online

In the early 2000s, availability of products and services, as well as increasing variety, was also indicated as a significant reason for online shopping (Ahuja et al. 2003).

In the 2010s, Wu et al. (2018) mention that previous studies touch on the importance of accepting the new shopping environments and the emergence in technology as being indicators of new consumer behaviors, while simultaneously influencing such behaviors. When it comes to online communications, the constant promotions and banners plastered in different social media and other platforms that consumers frequently visit, can easily attract and influence consumers towards particular products and behaviors.

According to a study by Laguna et al. (2020), during the start of the pandemic, consumers were asked about their grocery shopping habits in terms of place and frequency of shopping. Results show that prior to the lockdown, approximately 96.6% of consumers purchased their groceries from supermarkets in person, 39.21% from small local shops, and just 7.56% purchased them online. Consumer purchasing frequency was mostly twice a week 50%, weekly 35% and daily 15% prior the lockdown period. During the lockdown, shopping habits naturally changed, however, the place of purchasing did not. What changed was the frequency of shopping. Of the respondents, 76.5% shopped weekly, whereas the percentage of the consumers who now shopped twice a week reduced to 13.65%. According to the study, consumers who shopped daily were just a small percentage that now almost disappeared as it is reduced to 1.12%.

By analysing the volume of food people purchased, Laguna et al. (2020) were able to identify that over 20% of participants changed their shopping volumes. The fact that consumers were able to change their purchasing volume may also indicate that the changing circumstances made consumers change their purchasing habits. This may prove helpful as many online grocery shops have a minimum order value, which makes consumers eligible for purchasing online. By having this minimum order value, grocery shops 'force' consumers to purchase a specific volume which consumers can adapt to, as it is necessary for their online purchase.

Gao et al. (2020) posited that since the pandemic, the likelihood of consumers choosing to shop groceries online has increased due to the great risk of exposure to the virus. When people physically went to stores they were more vulnerable to get the virus. Therefore, online shopping was a natural transition for many, especially those in vulnerable groups where they wanted to have reduced contact with other people that could reduce the chances to infect them with the virus.

2.1.2 Habitual categories bought from the internet

This sub-chapter identifies the most common types of products bought as the internet progressed. According to a survey by Ahuja et al. (2003), most common categories of products bought from the internet in the early 2000s, were travel and audio-video, followed by computers and apparel, while groceries being at the bottom of that list.

Some products are less likely to be purchased online because of their intangibility, therefore making clothes, for example, less likely to be purchased as consumers cannot try them on or touch them before their purchase (Lim et al. 2016; Comegys et al. 2009). However, opinions on the matter fluctuate. For example, in the early 2010s, many stated that product information on clothing sites made it possible for consumers to gather enough information to make an informed decision (Liu and Guo 2008).

Online or digital marketing is a new trend which has gained huge popularity but questions around it arise. During the pandemic period Prashar and Mohali (2020) asked a question: Is the internet an appropriate place to purchase all types of products?. Online grocery shopping was not quite attractive compared to other products. However, in the past years it became a rising trend and now approximately 40% of the grocery market is transferred online (Prashar and Mohali 2020). Thereby, it is imperative to understand what influences consumers to shop online. Many products have a long history of being purchased online but grocery products have not. Recently there has been a shift in consumer perception which provides a new opportunity for online grocery shopping. While many consumers welcome the online shopping trend, they do not take it as an option to completely replace their traditional in-store habits. Contrastingly, 34% of supermarket shoppers have altered their shopping habits from in-store to online, while 60% continued to use the supermarket for their food and grocery shopping during the COVID-19 pandemic (Prashar and Mohali 2020).

2.1.3 Obstacles to online shopping

Obstacles to shopping online in three eras			
↓ Obstacles/Era →	2000s	2010s	COVID-19
network reliability	✓		
lack of security	✓	✓	✓
lack of customer service	✓	✓	✓
Disappointment in expectation vs reality	✓	✓	✓
lack of slot availability		✓	✓

Table 2.2: Obstacles to shop online

Numerous obstacles exist affecting consumer behavior. In the early 2000s, the obstacle to online shopping as shown in Table 2.2, was network reliability (Liang and Huang 1998; Limayem et al. 2000; Cockburn and Wilson 1996).

Lack of security and customer service were observed to affect all eras. According to a survey conducted by Ahuja et al. (2003) on students and non-students, who did not purchase online, to identify the reasons behind their unwillingness, many important findings were observed. The overwhelming concerns in both samples were privacy and security. Consumers considered it a burden when they face difficulty with their log-in details, having limited product information and when customer services where difficult to reach via phone (Katawetawarks and Wang 2011).

Another important obstacle that was presented across all three eras, was the disappointment of consumers when the ordered product was not as expected. According to Prashar and Mohali (2020) in-store purchasing makes it easier to purchase a product by touching or seeing it, however

that is impossible when done online. Consumers are only left with an image and some information about the product to make their decision.

Additionally, another main deterrent of online grocery shopping is the "fear of spoilage of perishable goods and lack of trust". Lack of slot availability was found to be a significant issue when purchasing online during the 2010s and COVID-19. Due to the high demand of online orders especially in the COVID-19 period, available slots for delivery were limited (Li et al. 2020).

Moreover, studies indicate that grocery shopping, along with related decisions for households, fall within the female members of the house. Equivalent preference now falls within younger or middle aged individuals of the society. Older individuals usually avoid online grocery shopping due to their lack of technology intelligence, however, that is not always the case. Therefore, sometimes age or even power of decision-making within the household can be an obstacle to online grocery shopping (Prashar and Mohali 2020).

2.1.4 Website design & features

One of the most important aspects for consumers that sometimes deter them from making an online purchase is the checkout process, which was consistent across all three eras.

Starting from the early 2000s, according to Aldridge et al. (1997) online platforms needed to reassure their customers and the information they provide, including banking details and addresses, were protected and safe. Online stores may use integrated mechanisms to build trust in safeguarding consumer's personal information and avoid misuse of credit card payments. According to Limayem et al. (2000) the easier the checkout process, with incorporation of security measures such as encryption, Secure Sockets Layer (SSL), and secure payment systems, should be mandatory and requires implementation.

Further at the 2010s, Lim et al. (2016) expanding on previous warnings, consumers purchase products and services from suppliers they trust, or brands that are familiar with (Lim et al. 2016). Therefore, trusting sellers or brands online is imperative and either a factor that incentives or deters consumers from shopping from online retailers (Lim et al. 2016). Therefore, certificates from organisations like eTrust, increases the chances of online consumers making a purchase from websites as it increases the reliability, confidence and trust consumers feel towards the supplier which in conjunction will also benefit the website with increasing sales (Korgaonkar and Karson 2007).

Online shopping has brought the traditional shops and shopping experience into the pockets of customers where they can purchase products with just a single click (Prashar and Mohali 2020). Finally, with the arrival of the COVID-19, online platforms need to handle the aforementioned problem with checkout process information. Since online grocery shops are still growing, consumers are critical over their evaluation. Customers read online reviews by previous customers to assess if they should trust a website or not and find as much information to make a rational decision. Therefore, this model is a guidance for online grocery shop owners as well as marketers to be aware of customer intentions and assessments in order to direct and adjust their activities towards attracting more customers.

2.1.5 Customer service

Customer service is imperative for the success of any platform or physical store. Good customer service in online stores has been critical since the early 2000s (Limayem et al. 2000).

Katawetawarks and Wang (2011) found that 72% of online consumers from a study they conducted, stated that customer service is a major factor in online shopping satisfaction. When customer services are unavailable or unreachable, then customers become skeptical and perceive that companies try to hide an unattractive aspect of their platforms or simply do not have the

intention to solve their issues. As customers are unable to touch or try the products prior to their purchase, online stores should offer some additional options as a trade off, one of this being customer services.

COVID-19 created an opportunity for online platforms to solve their existing customer service problems (Elliott, C 2021). Better technology, more ways to communicate and penalization, where added in customer service sections. The results of a survey conducted by (Deloitte Insights 2021) shows that 58% of respondents could provide a brand name that improved their customer services after the outbreak.

2.1.6 Summary

The background research conducted for this paper shed light to the motivations of people to either shop online or in-person. The importance of having enough information and limiting or dissolving information asymmetries, which mainly benefit shop owners, was pivotal. Additionally, there was an observed trend of changes in habits by customers due to the current pandemic. An increasing amount of customers now shop online rather than in-store due to the fear of the virus. Benefits and disadvantages of online and in-store shopping were also highlighted. This chapter also investigated how web design, content and the website meeting customer needs, are also some of the motivators or barriers to customers shopping online. Finally, the importance of customer service and how its absence can be detrimental to a website was also highlighted.

All the aforementioned details showed that older aged consumers, with limited experience with technology, are in a very disadvantage position even more so with COVID-19. The limitations and problems that this category of consumers is facing is taken into consideration in order for the MySuperTrolley app to have features which will make navigation as easy as possible.

2.2 Pre-existing Related Applications

This chapter will compare and review three of the pre-existing related applications that allow for price comparison of different grocery shops or sell grocery items from multiple stores. Important features implemented, strengths and weaknesses for each application will also be discussed. An excel spread sheet with additional comparison is added to the Appendices at section B.

2.2.1 Trolley (Web App)

Trolley is a free online price comparison mobile responsive web app, previously known as SupermarketWatch, established in 2020 (Trolley 2021). The application currently displays items from more than 10 well-known shops in the UK with most of the retailers to be grocery shops as shown in Figure 2.1. Trolley aims to help consumers save money on their everyday grocery shopping. This application does not have any revenue and every retailer that wants to add their items into this app can contact the developers.

The screenshot shows the Trolley website's item details page for 'Ariel Colour Washing Gel 24 Washes'. At the top, there is a navigation bar with a search bar and various category icons: Baby & Kids, Chilled Food, Drinks, Food Cupboard, Fresh Food & Bakery, Frozen, Health & Beauty, Home & Entertainment, Household, Meat, Fish & Poultry, Occasional, Pets, and Vegan & Free From. Below the navigation bar, there is a product image of a purple Ariel washing gel bottle labeled 'Color & Style' and '24'. To the right of the image, the product title 'Ariel Colour Washing Gel 24 Washes' and '888ml' are displayed. Underneath the title, it says 'Price Comparison'. Below this, there are four price comparison cards for Waitrose, ASDA, Morrisons, and Tesco. Each card includes the shop logo, average rating (4 stars), price (£4.00), price per litre (£4.51, £4.50, £4.50, £4.79 respectively), promotional offer ('2 FOR £7' for ASDA, '3 FOR £10' for Morrisons), and a 'VISIT SHOP' button.

Figure 2.1: Trolley: Item Details page

Features

- Find items either by search or by category section.
- Display average rating of items in a precision of half a star (max. 5 stars) along with the number of people rating this item.
- Compare all available shop's prices for a specific item.
- Contains FAQ & and asking a question sections.

Strengths

- Compare over 10,000 products and 2,000 brands from 11 supermarkets and shops across the UK with everyday items from household essentials to beauty and medical needs.
- Data are updated daily with all the latest prices from each supermarket, including promotional offers.
- Displays title, description, photo and price of the item. These information are most important for consumers to be able to completely understand the item.
- Displays related items with the selected item. Additional items include items that are most likely to be chosen by consumers next, making the process easier for the consumer.

Weaknesses

- There is no basket in this website. It is just for comparing prices. Therefore, consumers cannot make an order from the website.
- No filtered sub-categories of items. Application includes a large number of items. For example, one of the categories is called "Meat, Fish & Poultry" with so many items in it making difficult the selection of product wanted.
- Irrelevant "Today's Biggest Savings" section placed in the home page. Items included in this section are neither the cheapest, nor are in on offer that day.

2.2.2 Buymie (Android & iOS Mobile App)

Buymie is a mobile app for on-demand groceries established in 2015 (Buymie 2021). The app allows you to order items from 3 local stores, namely, Lidl, Tesco and Dunnes as shown in Figure 2.2, and have them delivered by personal shoppers in no more than an hour.

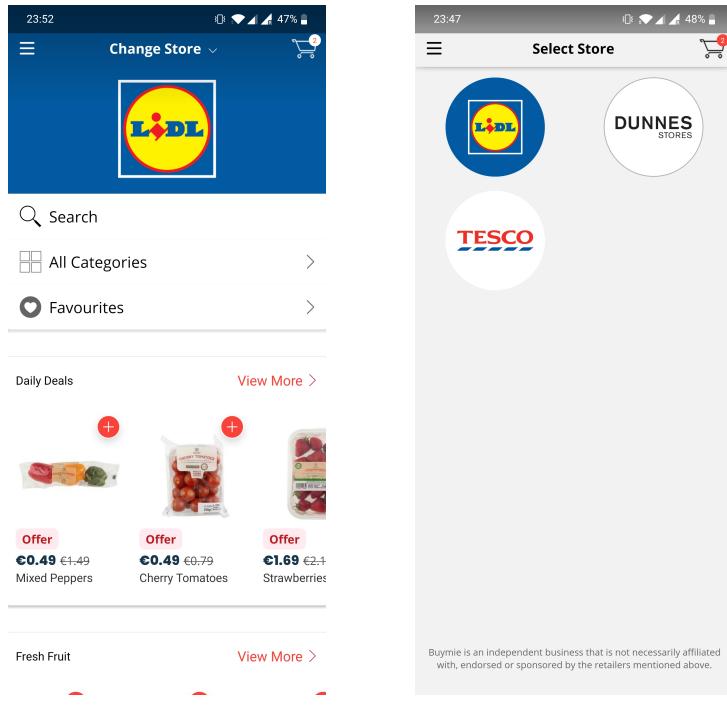


Figure 2.2: Mobile Application with three stores

Features

- Find items either by search or by the category section.
- Add note to the shopper for a specific item.
- Have an account. You can also change the account details.
- FAQ section & ask a question via a live chat that will be answered within 24 hours.
- User can purchase items by adding items to the basket.

Strengths

- The only application where a user can shop online items from Lidl in the UK (Bristol).
- Available for same-day delivery.

- Customer service in case the consumers encounter any problems whether before making an order or after.
- Creating an account provides crucial opportunities to consumers, mainly for people that are not as tech-savvy. Such features are the previous order favourites sections.
- Offer opportunities, such as refer a friend and both of you get €10 for your next order, free delivery on orders above 40 (€3 otherwise) and Buymie Plus (€8.33 per month) which includes unlimited free delivery.

Weaknesses

- Cannot order items from different stores simultaneously. With minimum order value being €30, consumers maybe deterred from purchasing.
- Consumers can only search for items from a specific store. Therefore, cannot compare prices of items from the three stores.
- App delivers only in Ireland and UK. However, valid postcodes were only identified for Bristol locations for purchases from Lidl thereby making the delivery area limited.
- Descriptions are not helpful or in some cases non-existent resulting in misunderstanding of purchasing item from consumers.

2.2.3 Devo (Web App, Android & iOS Mobile App)

Devo is a Web, Android and iOS Mobile application. It offers delivery from various local corner shops to your door as shown in Figure 2.3 in as little as 10 minutes (Devo 2021).

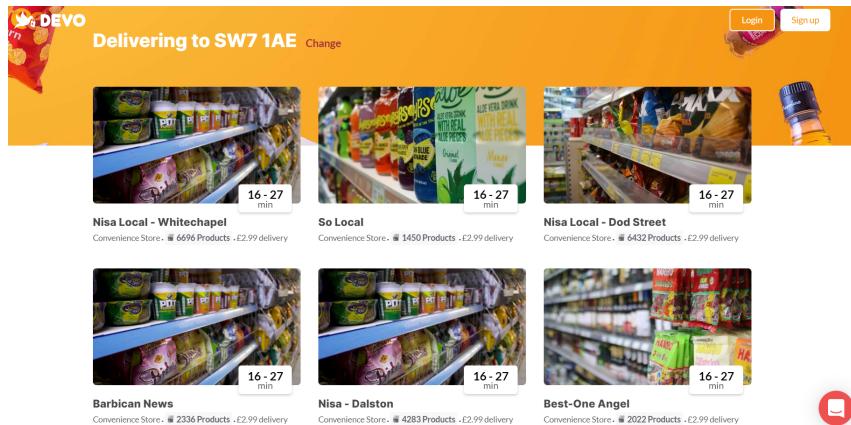


Figure 2.3: Devo: Shows local stores available according to the postcode entered

Features

- Find items either by search or by the category and subcategory section.
- Have an account. You can also change the account details.
- FAQ section, ask a question by sending a message that will be answered within few minutes.
- Users can purchase items by adding items to the basket.

Strengths

- Consumers first need to input their location. This determines if they can deliver your food to your location. If yes, then it displays different grocery shops available nearby, along with a range of time it needs for the food to be delivered in addition with the number of products the shops sells and delivery cost. It also displays the stores' opening hours.
- Allows payment with Google pay (secure payment).

- Displays title, description, photo and price of the item. These information are most important for consumers to be able to completely understand the item.
- Displays related items with the selected item. Additional items include items that are most likely to be chosen by consumers next, making the process easier for the consumer.
- Available for same-day delivery. Consumers can choose ASAP or scheduled time.
- Customer service in case the consumers encounters any problems whether before making an order or after.
- Creating an account provides critical opportunities to consumers, mainly for people that are not as tech-savvy. Such features include the previous order favourites sections.
- Offers opportunities such as refer a friend and you both get 5 pounds off of your next order.

Weaknesses

- Necessary to register in order to make an order.
- Consumers cannot add items from multiple stores simultaneously.
- Consumers can only search for items from a specific stores. Therefore, cannot compare prices of items from other stores.
- Limiter area on delivery. Only in some areas in London.

2.2.4 Summary

The three pre-existing applications provide a variety of functionalities that are very helpful and mandatory nowadays, to satisfy consumer needs for grocery shopping. However, none of the apps can completely satisfy all consumers needs. A combination of: comparison of prices between shops in Trolley, customer service provided in Buymie and secure payment with Google Pay along with same-day delivery provided by Devo, can result in a complete application that can dominate others in the future. The combination of the important features along with the most necessary functionalities and an attempt to minimise weaknesses of these applications are taken into consideration for the implementation and creation of MySuperTrolley.

3 | Requirements Analysis

This chapter will explore the functional and non-functional requirements of the project. These requirements were determined at the very initial parts of the project. However, after the implementation started and more functionalities were implemented, they were then later revisited and updated accordingly.

3.1 Requirement Process

In order to make a final list of functional and non-functional requirements, the following procedure needed to take place. Firstly, all ideas were written down as bullet points and categorised under important features that required implementation. Then, some user personas were initially created in order to describe how different users will interact with the application. Afterwards, those scenarios were converted into user stories for both grocery and admin/manager users. Later, a first draft of requirements was created followed by a survey conducted for participants to categorise the requirements from least to most important. When all those parts took place and taking into consideration comments from the survey responses, the final functional and non-functional requirements were finalised.

3.2 IDEAS

This is the first step of the creation of requirements. Ideas were written down and categorised into six different important functionalities of the application. Those functionalities are: what do consumers see the first time they visit MySuperTrolley, important opportunities for grocery users and for admin/manager users, as well as delivery and payment. Multiple ideas were listed capturing most of the issues found from the background research on online shopping behaviour which can be found in Appendices at section C.

3.3 User Personas

Given the ideas of MySuperTrolley, some scenarios were developed, in order to capture how admin/manager user and different type of grocery users, from different age ranges and different technology knowledge and experience, will interact with the application. The following scenarios have a main goal, for grocery users, the ordering of items from more than one shop in a single basket, which will be implemented in MySuperTrolley. Some of the scenarios are detailed below, however a PowerPoint with all user personas is also attached in Appendices at section D.

- **Kyle** is a 43-year-old employee in a big firm in London. He lives with his wife and two children in a flat in central London. He was using the underground to go to work. Kyle was buying groceries for the whole family, mainly from Tesco and some specific items from Iceland and Lidl for his children near his workplace. However, he is currently working from home and is not able to find an Iceland store near their flat. The next available delivery slot for him to buy some of the groceries online is in five days which gives him the flexibility

to add more items in his basket before the delivery date. Thereby, Kyle has the option to add items that may have run out even a day before his delivery slot. However, as Kyle buys a lot of products from all three stores he needs a website where he can order all weekly items from Tesco, Iceland and Lidl and have them delivered at their flat.

- **John and Lucy** are a married couple in their 70s. They are enjoying their retired life living off of their pension in a house outside of London. Every Saturday they drove to different grocery shops to satisfy their weekly grocery needs. Due to COVID-19 the government guidelines advised for older people to stay home and try to limit their transportation in order to reduce the chances of them getting sick or hospitalised. John and Lucy are also not tech-savvy and their son usually helped them to order groceries from Tesco. However, Tesco alone does not satisfy their needs. John wants to buy specific products for his plants, which he grows in the garden, where he previously bought from Morrisons. Therefore, John wants to shop items for other websites as well. They would like a website where they can buy items from different shops in one basket. Also, since they are not very experienced with technology and their weekly shopping items are almost identical, a previous order section where they could repeat their purchase would be very helpful to them. In addition, items that they frequently purchase but not on a weekly basis could be added to a favourite section so as to not browse the whole website for items they want. This will help them find such items easier for future orders.
- **Anna** is a 35-year-old doctor working in a hospital in Glasgow, but she lives in a small village 45 minutes away from Glasgow centre. She is working many hours every day at different shifts and she frequently comes home very tired after working. Due to COVID-19 she is afraid of going to the supermarkets to buy her groceries because there is a possibility for her to infect other shoppers there. Thereby, she needs an app where she can order groceries by booking available slots that will fit with her work schedule. She also wants to be able to make sure that her order can be delivered to her location before adding items into her basket.
- **Jed** is a manager at MySuperTrolley. He wants to be able to have access and edit or delete items when there is a glitch on the website, so that he will prevent purchases of items that are unavailable or have other issues.

3.4 User Stories

Following the creation of the concept of user scenarios as described in the previous sub-section, user stories for both grocery users and admin/manager user were developed. Some of them are listed below, however, a full list of user stories is added in the Appendices at section E.

3.4.1 Grocery User

What consumers see first when visits MySuperTrolley

- *As a grocery user, I want to view cities of available areas of delivery, so that I will make sure that the order can be delivered to my location.*
- *As a grocery user, I want to be able to add my postcode, so that I will see if the order can be delivered in my location.*
- *As a grocery user, I want to view the different shops that sell items in the website, so that I will know from the start if i want to continue or not.*

Options available for grocery user

- *As a grocery user, I want to be able to register/login, so that I can have extra opportunity functionalities.*

- *As a grocery user, I want to have access to my previous orders, so that I can make my future orders easier.*
- *As a grocery user, I want to add/remove items from favourites, so that the selection of items in future orders will be easier.*
- *As a grocery user, I want to be able to see all grocery shops that sell the item I selected with price and offer if any, so that I can compare the different items before making a decision.*
- *As a grocery user, I want to view the offers category, so that I will see if any of the items satisfies my needs and add it to my basket.*
- *As a grocery user, I want to find items from specific categories, so that I will make my selection of an item faster.*
- *As a grocery user, I want to filter categories into subcategories, so that It will make my selection of an item faster.*
- *As a grocery user, I want to view title, description and photo of a product selected, so that I will fully understand an item.*
- *As a grocery user, I want to add items to the basket from different grocery shops, so that I can satisfy my grocery needs from one website.*
- *As a grocery user, I want to be able to see a preview of the order before payment, so that I make sure all the items I selected are in my basket.*
- *As a grocery user, I want my basket to separate the items based on the selected shop, so that I know how much money I spend in each shop.*
- *As a grocery user, I want to view the available slots along with the delivery cost, so that I can select the most convenient slot for me.*

3.4.2 Admin/Manager User

- *As a manager, I want to be able to manually edit/delete some products, so that I can prevent customers from buying some items in case of a glitch.*

3.5 Survey

After the creation of user stories, a first draft of requirements was developed. Then a survey was conducted with the list of requirements that can be implemented in MySuperTrolley. This was achieved by asking people to rate requirements on their importance to them as a future user of MySuperTrolley, in a five-point scale from not important at all, to most important. Firstly, the ethics checklist given in Appendices in section A was required to be filled by the survey takers. People were then asked if they usually shopped online as well as if they usually shopped groceries online in a three-point scale with yes, no or sometimes as options. With a sample of 41 responses, 62.5% usually shop online, 35% sometimes shop online while 2.5% of the survey takers, do not usually shopping online. Out of all the responses, 19.5% of people usually shop groceries online, 31.7% sometimes shop online and 48.8% do not usually shop online. Participants were then asked if they would like any extra features implemented in MySuperTrolley app which were not stated in the survey and if they have any extra comments to mention.

With 41 responses, some of the following answers are displayed below, however the link for all the questions and answers of the survey is given in the Appendices at section F. When participants were asked if the user should be able to enter their postcode to check if their location is available for delivery, 85.4% thought of this feature as "Very Important", 9.8% as "Important" and only 4.9% as "Not Important". However, the following question "Users receive notification for availability of postcode", as shown in Figure 3.1, had a variety of responses. Some responses stated that this feature was highly important to be included in MySuperTrolley app, however, others responded that to get notification for entered postcode is not as important compared to the previous question.

Overall, for users to receive notification for availability of postcode is a necessary feature as 61% of survey takers, consider it as "Very important" and "Important". This was taken into account when implementing the application.

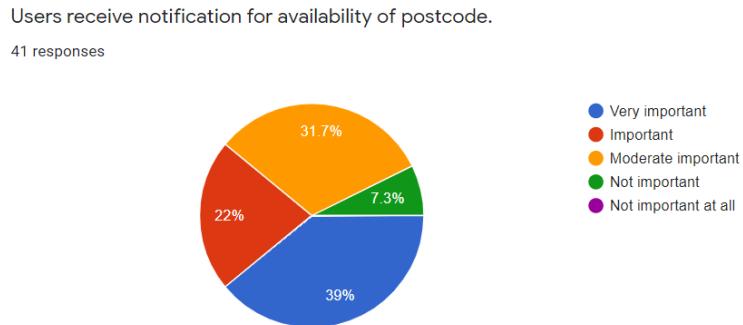


Figure 3.1: Requirements Survey Question 4

When survey takers were asked to rate a feature that allows users to add items from different grocery shops in a single basket, the majority of participants, with 70.8% found it "Very important" and "Important" while only 7.3% found it "not important" as shown in Figure 3.2.

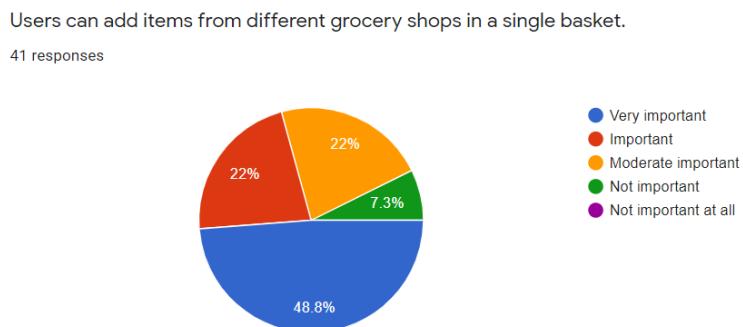


Figure 3.2: Requirements Survey Question 25

Additionally, when participants were asked about authenticating with Facebook/Google, as shown in Figure 3.3, only 46.3% found it useful, indicating that this feature is not a priority and should be skipped as an implemented feature in the application.

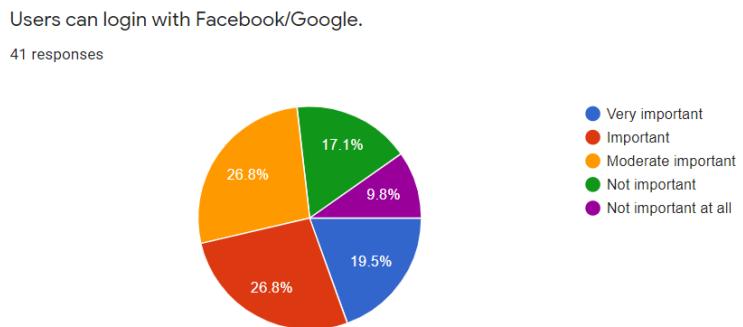


Figure 3.3: Requirements Survey Question 10

3.6 Final Requirements

Once all the previous steps of the requirement process were completed, a set of functional requirements could be established. These requirements would be used as reference for all tasks requiring implementation in MySuperTrolley according to their relevant importance. The MoSCoW method was used for the importance of those tasks, which is a method of prioritisation in which requirements are labelled as "must have", "should have", "could have", "won't have" (Agilebusiness.org 2019). Along with functional requirements, a set of non-functional requirements were established which are needed as part of the system design. Whilst functional requirements describe what the system should do, non-functional requirements, even if they do not directly address a functional request, are necessary for the system to function efficiently.

3.6.1 Functional Requirements

Must have – Describes a requirement that must be satisfied in the final solution to be considered a success.

- Show countries/cities available for delivery.
- Show grocery shops that have items in the website.
- Users can enter a postcode to see if their location is available for delivery.
- Users can have access to previous orders.
- Users can add/remove items from favourites.
- When clicking on a product, the page will show all shops that sell a specific item along with price and offer (if any).
- Show details of an item (title, description, photo).
- Users can find items from categories sections.
- Users can use a filter for subcategories.
- Users can add items from different grocery shops in a single basket.
- Users can see the preview of the order before confirming the order.
- Users can view available slots for delivery (range of time, day, cost).
- Users can book a slot for delivery.
- Users can register/login.
- Users can view items they added to their basket separated by grocery shop along with their total price.
- Users can make an order with items from multiple shops in a single basket.

Should have – Represents a high-priority item that should be included in the solution if it is possible.

- Users can access offers section.
- Users receive notification for the availability of postcode.
- Users receive notification for confirmation of the order.
- Users can add multiple number of the same product in the basket (using item quantity).
- Users can ask a question if they have any inquiries.
- Live chat for customer service & support.
- Users can see FAQ with answers.
- Users can select next-day, or another selected day for their delivery, in order to see which shops can deliver items to their location based on their selection (e.g some shops may provide next-day delivery because of their proximity to the customer. If a customer selects next day delivery, only shops that offer next-day delivery will be displayed. If the customer changes their delivery slot more shops will appear).

- Users can see the related products section for a selected item.
- Users can use a filter for the number of products per page, high to low price or vice versa and previous customer reviews.
- Users can update their personal details.
- Users can search for items.
- Users can edit their basket (users can also do that after their order is confirmed up to a specific time period before their order is shipped and paid for).
- Manager can manually add/remove some items (e.g in case of a glitch).

Could have – Describes a requirement which is considered desirable but not necessary.

- Display the username when the user is authenticated.
- Users receive a verification email when they register for the first time.
- Users can pay with apple/google pay.
- Users can log in with Facebook/Google.
- Users can find products through the most-shopped groceries section (section of most purchased items).

Won't have – Represents a requirement that were agreed will not be implemented in a given release but may be considered for the future.

- Users can receive an alert when an item they have in their favourites section goes on sale.
- Users can have the option to be called or receive a text which indicates that their order is on the way.
- While searching for products, it would be useful to indicate the products that the users have previously bought.
- Users can be able to see their final order before paying, and be able to transfer an item from one shop to another, using drag and drop. For example, if they order 9 things from Tesco and 1 from Lidl it might be better to add that 1 item from Lidl to their Tesco basket so they get everything from a single store. However, a simple drag and drop will be a good feature.
- The app should have different user groups, for example, elderly, family, student, etc. where these groups should have different priorities when it comes to delivery slots.

3.6.2 Non-functional Requirements

Performance

- The website's load time should not be more than two seconds for users.
- The application should not slow down when the number of users using the website increases.
- Real-time updating of items from different shops.

Security

- Credit card details must be stored in line with legal requirements.
- Passwords must be encrypted.

Usability

- The application must be easy to learn, require no formal training and have an intuitive flow between pages.
- The application must be responsive and should function correctly on both desktop, tablets and mobile devices without any problems or restriction of features (responsive web app).

4 | Design

This chapter will discuss the software engineering processes and practises used throughout the project. Agile development methodology, including version control and iterative development are going to be discussed. Additionally, system architecture, ER-diagram, activity diagram along with various prototypes will also be introduced. Finally, this chapter will explain the Tools and Technologies selected for the creation of MySuperTrolley in replacement of other similar ones.

4.1 Agile Scrum Development Methodology

The project focuses on the agile development methodology, which divides the development process into iterative smaller steps with the introduction of testing, flexibility and changes throughout the life cycle of projects (Lucidchart 2017a). In order to fulfill the different sections of agile development cycle, SCRUM framework is introduced as shown in Figure 4.1. Generally, SCRUM includes a development team with a Product Owner and a SCRUM master (Lucidchart 2017b). The Product Owner is the source of information about the product while the SCRUM master is responsible for a smooth sprint process and to divide tasks between team-members. While completing this project and becoming my own SCRUM master, the responsibilities by each of these roles were taken into consideration and implemented.

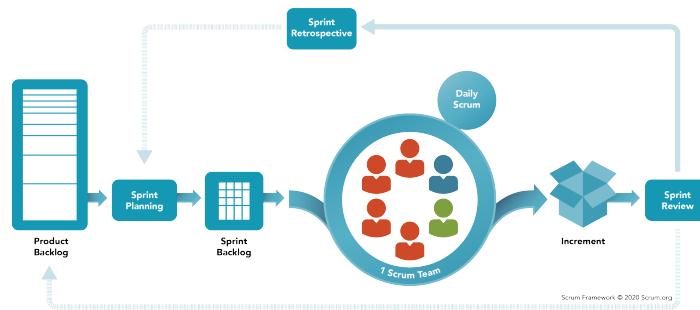


Figure 4.1: SCRUM Methodology

4.1.1 Stand-Up Meetings

As aforementioned, to fulfill the Product Owner responsibility, weekly stand-ups were made with my supervisor. Weekly stand-ups were a way for my supervisor to be kept up-to-date with the project and discuss any unclear tasks related to the project. Confluence was the main platform that used to keep the different sections of the project in structured folders. Two of the sections were the agendas and meeting notes. The agenda was prepared ahead of each meeting containing the project's progress. An example of an agenda is added in the Appendices in section G. Meeting notes taken during each meeting were written digitally in Confluence and then sent to my supervisor after the end of each meeting.

4.1.2 Version Control

A very important part of the project was the usage of a version control, throughout all the process. Bitbucket was used which is a Git repository management solution ensuring a place to manage repositories, backing-up your progress, avoiding the loss of any code in cases of local development issues (Atlassian 2021). The use of branching was mostly applied in the deployment process to ensure that if any unexpected issues came up, a full functional system would still be present on the master branch. Furthermore, even though Continuous Integration (CI) was considered out of scope, as discussed with my supervisor, the use of pipeline was mandatory for the deployment of MySuperTrolley and was set up to run unit tests as well as ensuring the code is correct, when a code was pushed into BitBucket.

4.1.3 Iterative Development

To fulfill the responsibilities of the SCRUM master, by ensuring a smooth process in the iterative development, JIRA was used as an issue tracking system. When the requirements were finalised, issues with meta-data, as shown in Figure 4.2, were added in the backlog of JIRA and were assigned into sprints.

The screenshot shows a JIRA issue page for 'MST-83'. At the top, there's a header with 'Done' and a green checkmark icon. Below the header, there are tabs for 'Development' (selected), 'Create branch', 'Priority' (High), 'Original estimate' (2h), 'Time tracking' (1d logged), 'Labels' (Back-end, Database, Front-end), 'Sprint' (None +1), and 'Reporter' (Nikolas Finiotis). A 'Comments' tab is active, showing two entries:

- Nikolas Finiotis** February 2, 2021, 9:19 AM: Changed the way of storing items in database for easier implementation of order preview page. All the items and brand ids will be stored in the database as string of JSON. The JSON should have keys the brand ids and values list of item ids.
- Nikolas Finiotis** January 31, 2021, 10:40 PM: Edited Due to difficulties of spring boot, I changed the way of storing the item details and I removed the many to many relationships and I added an extra column in the order table that stores the item id and brand id as json converted into a string.

At the bottom, there are buttons for 'Edit', 'Delete', and a link icon. A footer at the bottom right shows creation and update dates: 'Created January 31, 2021, 10:37 PM', 'Updated February 2, 2021, 9:19 AM', and 'Resolved February 2, 2021, 9:19 AM'. There's also a 'Configure' button.

Figure 4.2: Issue meta data

4.2 System Architecture

The project's system architecture was designed using a three-tier architecture pattern, namely, presentation, application and data layer. The reason of selecting such a type of architecture pattern was to separately focus on each individual layer without directly affecting the usage of the other layers.

Presentation layer: this layer represents the user interface of the web application, as shown in Figure 4.3, in front-end client side. This layer consists of three main components, namely, router, components and services. Components, are all the different pages implemented in the web application. Router is responsible for rendering the matching of components when the user clicks to navigate to a different component. Moreover, the service component handles the API requests, which is the connection for transferring information from the presentation layer to the application layer. Those requests are accomplished by the AXIOS HTTP Library.

Application layer: this layer acts as the brain of the application, as it consists all the logic between the connections of the three layers. This layer contains four important components, namely,

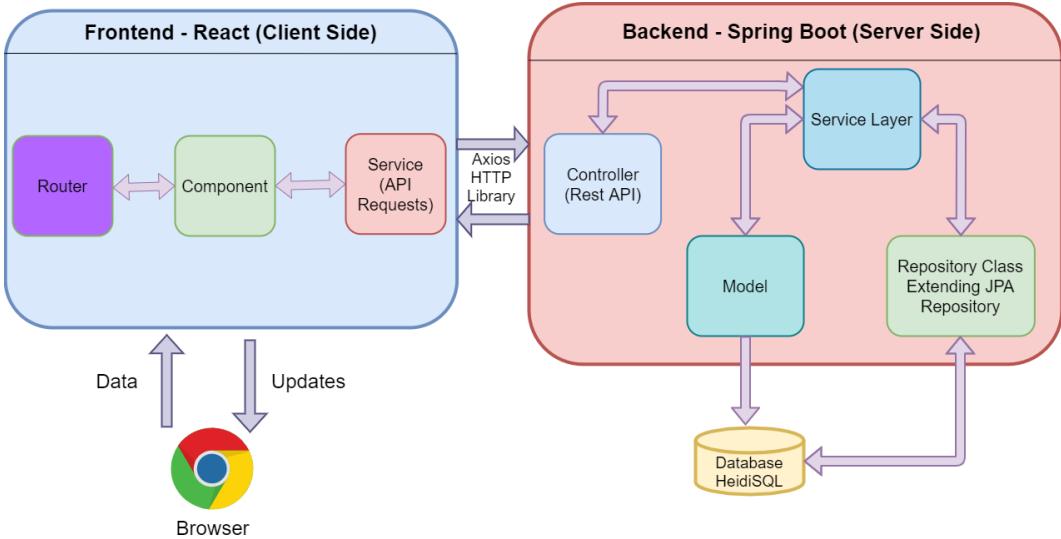


Figure 4.3: React - Spring Boot MVC System Architecture

controller, model, service, repository, as shown in Figure 4.3, in back-end server side. The controller is responsible for the REST API. All the requests made by the service component from the presentation layer are handled by the controller component. The model contains all the different entities and relationships between them. When a request is made the controller component calls the service layer, which then calls the repository class, which extends the JPA repository in order to communicate with the database to give the correct instructions for a successfully request.

Data layer: this layer contains all the storage information for the web application. The database tables and columns are connected with the model component of the application layer. However, the repository component is responsible to handle the additions or selections from the database.

4.3 ER Diagram

Entity Relationship (ER) diagram, is used in database design in order to structure data. The important information visualised in this diagram, are the major entities in the project scope and the inter-connected relationships among these entities (Visual Paradigm 2019). The major entities of MySuperTrolley, are the users, addresses, items and brands. The items table is connected with category, sub-category and sub-sub-category tables. The address table is connected with the city and the country table. All aforementioned entities with their relationships are added in the Appendices in section H. Additionally, the user entity is going to be briefly discussed and explained in this section as an example, as shown in Figure 4.4.

The users table is the most important table, as it is responsible for the authentication of details required for a user to either sign-up or log-in to the application. There are two types of roles a user can have, either grocery user or admin. Therefore, a one-to-many relationship connecting the users table to the roles table is created. Furthermore, there is a many-to-many relationship between the user and address entered by the user. Users can order from different locations each time, and an address can occupy many users due to apartments or houses being occupied by more than one individual. That is why a new table was created called 'address_users' where the primary keys from address and user table are foreign keys in the newly created table. This table was created in order to save user addresses for their orders as they can reuse it for future payments. Additionally, a one-to-one relationship is created between the user and the favourites

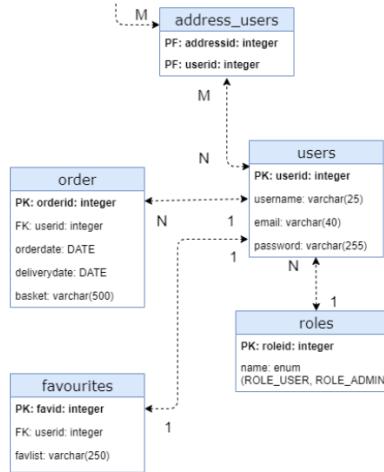


Figure 4.4: Users entity with inter-connections

table so that all the items that a user considers favourites can be saved and reused, once the user is authenticated to the application. This is done for users to find their favourite items easier in the future. Finally, every time the user makes an order, the order details need to be stored in the database. That is why there is a one-to-many relationship between the user and the order table.

4.4 Activity Diagram

Activity Diagrams (ADs) are graphical representations of workflows of consecutive activities and actions taken given a choice (GeeksforGeeks 2017). In the application, ADs aim to model the workflows and data flows as they intersect with different related activities.

The following diagram represents the activity flow, for both grocery and manager users, as shown in the Appendices in section I. In this section the grocery user workflow is going to be discussed.

A user that visits the application for the first time has one option, to enter a postcode in order to check the availability of their location for delivery. If the location is not valid the user can re-enter a postcode. The user receives a response, either the postcode is valid or not. If the location is not valid the user can proceed further to the website to compare prices. If the location is valid, the user can then see the available stores, days and times they can book for a delivery slot.

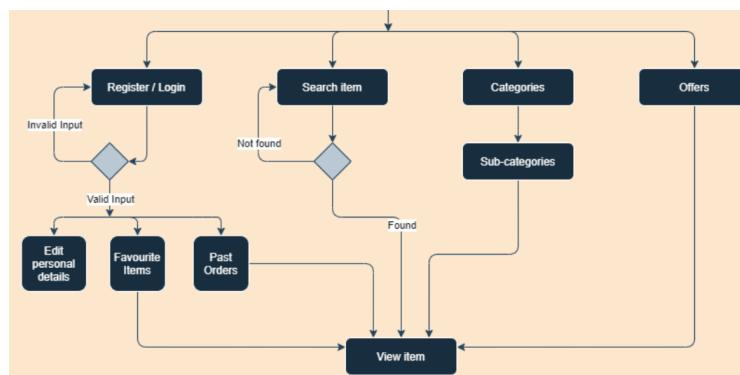


Figure 4.5: Activity Diagram Example

The user can then select between the following four options, as shown in the Figure 4.5. The user can register/log-in, to be able to edit their personal details, see past orders and favourites items. The user can also search for items. Additionally, users can find items through categories or through sub-categories and also view different offers.

All four options can lead to the viewing of an item. Thereafter, users can add items to favourites or into their basket where they have the opportunity to view or edit their basket.

Finally, the user can view a preview of their order. At this point the user can also change the delivery date. This is done to give the option for users to see if they can complete their order with shops selected based on their location. Afterwards, users can confirm their order and pay for it. At any time the user can chat through a live chat, see the FAQs and ask a question.

4.5 Prototypes

4.5.1 Paper Prototypes

Paper prototyping is a process of creating paper representations of digital products to realize concepts and test designs (The Interaction Design Foundation 2021). Paper prototypes are sketches to present the main functionalities that are considered important for implementation. Various paper prototypes were created for different parts of the application. Some include the initial page with the available slots and shops, login/register, categories and sub-categories, item details, order preview and past orders. Each of the prototypes were briefly discussed to present the idea behind the sketch and why it is presented in such way, as shown in appendices in section J. One example is shown in Figure 4.6

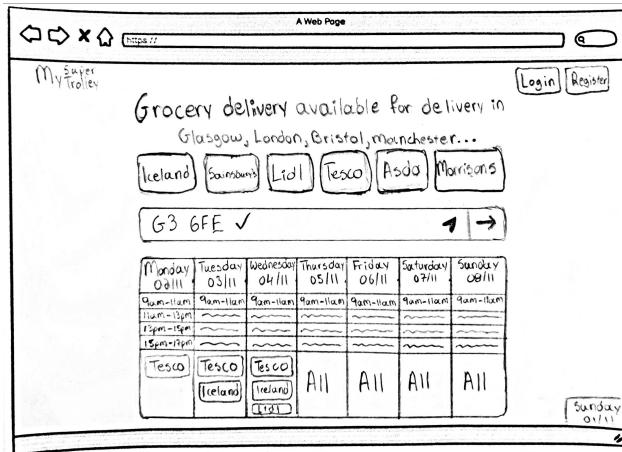


Figure 4.6: Valid Postcodes Prototype

The main idea behind Figure 4.6 is that it is useful for the grocery shopper to know if the items can be delivered to their location. Therefore, users can see which cities are available for delivery and enter their postcode to check availability and thereby book a slot based on their needs. For example, by taking into consideration the user's postcode, the application can display nearby shops for next-day delivery or other shops available for them to purchase from, on another selected day.

4.5.2 A/B Testing Survey

Some of the features of MySuperTrolley could be designed differently. By combining various already existing grocery shop applications, some paper prototypes were sketched. In order to decide which paper prototype would be further converted as a Wireframe, a survey was conducted where participants were asked to decide between paper prototypes for the same feature each time. The link for all the questions and answers of the survey is given in the Appendices at section K. For example, the two approaches for the category and sub-category page are shown in Figure 4.7 and 4.8. From 14 participants, 85.7% selected option A and 14.3% preferred option B thereby option A was selected and those prototypes were converted into Wireframes.

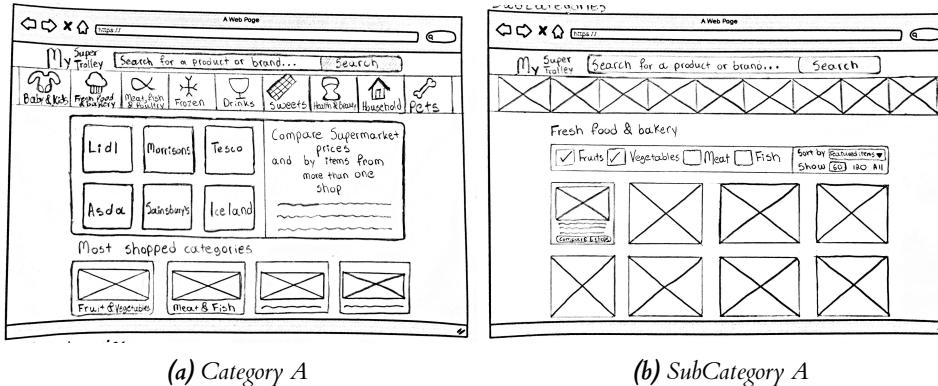


Figure 4.7: Option A

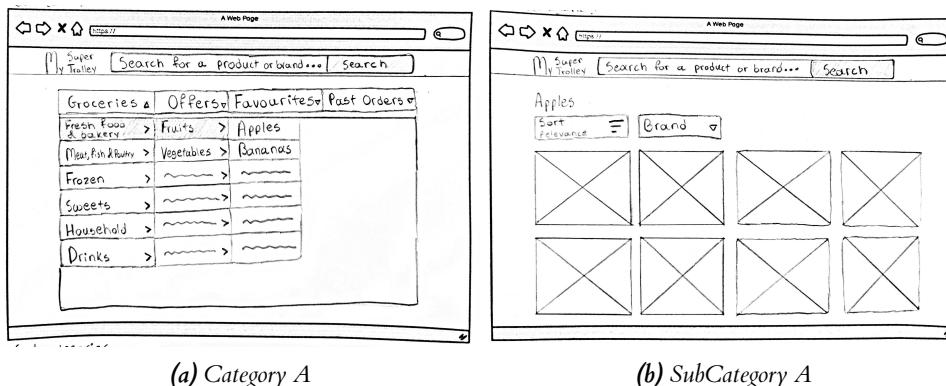


Figure 4.8: Option B

4.5.3 Wireframes

Wireframes are crucial for the design process. Wireframes are layouts of a web page that demonstrate what interface elements will exist on key pages (Experience UX 2019). Multiple Wireframes, such as log-in/register, available slots and shops, categories, item details, order history and order preview were implemented and the link to access them is added in the Appendices in section L.

An example of the items page is shown below, in Figure 4.9. In the Figure, the title, photo and description of the item is presented along with the details of the different shops that sell this exact item. Moreover, the related products section is shown to help the user to add items needed to

their basket easier. For example, the product shown is a Head & Shoulders shampoo. In the related products, different sizes of shampoo and conditioner are placed.

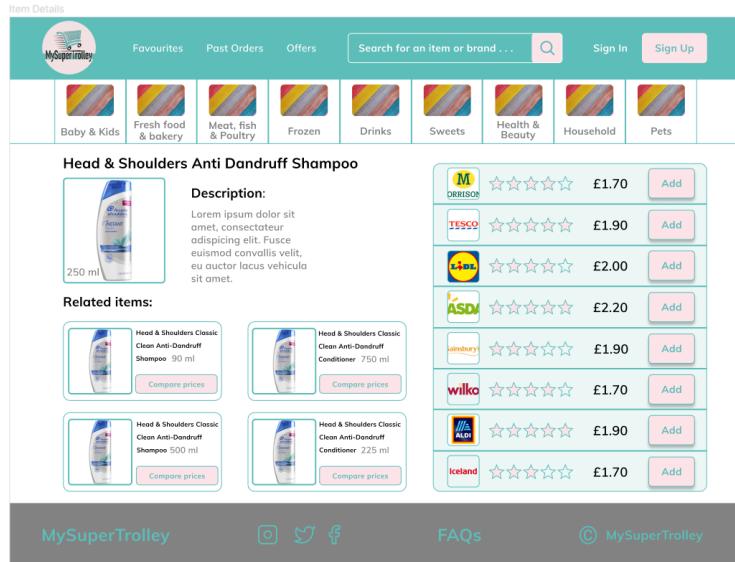


Figure 4.9: Item details Wireframe

4.6 Tools and Technologies

There are many tools and technologies available that could be used to build a web application for online grocery shopping. After an extensive research where the advantages and disadvantages, along with the restrictions of some technologies that may impact the implementation of MySuperTrolley were identified. These resulted in the selection of Maven Spring Boot MVC for the back-end, React with AXIOS HTTP library as a front-end as well as MySQL for the database.

4.6.1 Backend - Maven Spring Boot MVC

Spring MVC offers decoupled ways of solving web frameworks through a model view controller (MVC) design pattern. This separates your application into a triad, making it easier to build lines of code without having to start over each time. It provides a decoupled framework that makes it easy to test applications.

Both Spring Boot and Django are widely used frameworks among developers. Both frameworks provide similar features JavaSterling (2020). Nowadays, both Python and Java are useful to learn, however, Java is more stable, secure and scalable than Python. For this multi-tasking project, the use of Spring Boot should be preferred over Django, as Spring Boot can handle multiple requests at a time. Due to the fact that MySuperTrolley should be a secure application and lots of data will be required to be rendered from the database to the application while also be able to handle multiple requests at a time, Spring boot is selected. Spring Boot is also selected because it can be used with react in the front-end.

4.6.2 Frontend - React

Angular: A huge developer community exists behind AngularJS and it excels particularly in supercharging HTML code due to its ability to aid the construction of dynamic user interfaces (Cuelogic Technologies Pvt. Ltd. 2020).

React: Facebook and Instagram rely on the React JavaScript framework due to its ability to build large applications that are dynamic and can be vastly scaled (Cuelogic Technologies Pvt. Ltd. 2020). React is at its best when rendering complex user interfaces and is rapidly becoming the fastest growing JavaScript framework of the modern age.

React is commonly used when the app contains multiple events, the application development team has expertise in HTML, CSS, and JavaScript as well as when creating shareable components in your app project.

React was preferred over Angular because of isolated debugging, which helps to achieve better app's stability. What's more, the component-driven architecture of React allows for components to be re-used, and thus save development time and cut costs. Additionally, React is quick and efficient.

To make the app available for mobile users as well, it was decided to build a progressive web application. React.js was picked for this purpose as an effective, powerful, and flexible tool.

4.6.3 Database - MySQL

SQL databases provide great benefits for transnational data Chan, M (2021). The data integrity is paramount and the structure of the data doesn't change frequently. It's also best for fast analytical queries. Whereas NoSQL databases provide much more flexibility and scalability, resulting in rapid development and iteration.

Since MySuperTrolley application requires transaction data, and fast query requests, SQL database is selected for the implementation. MySQL is selected using HeidiSQL which is a useful and reliable tool designed for web developers using the popular MySQL server, Microsoft SQL databases and PostgreSQL (Ansgar Becker 2015).

HeidiSQL provides the opportunity to browse and edit data, create and edit tables, views, procedures, triggers and scheduled events. HeidiSQL can also export structures and data either to clipboard or SQL files, much more than other servers. There is encrypted communication between client and server which makes it lightweight. Additionally, it allows you to write queries with customized syntax-highlighting and code-completion as well as synchronise data between local and remote databases.

5 | Implementation

This chapter will review and describe the implementation of key features in MySuperTrolley. Additionally, the chapter will outline the use of various tools and libraries for implementing the specific features along with discussing any issues occurred during implementation. Finally, the deployment process of the application will be included in this section.

5.1 Postcode Validation

From the background research section, one of the encountered issues was that users became aware of the unavailability of delivery in their area very late in their journey. Most of the times users found this at the checkout page when adding their delivery details. This resulted in frustration, where users may later avoid visiting the application therefore losing future customers.

In an attempt to avoid this issue for MySuperTrolley, the very first page the consumer visits in the app, displays the cities available for delivery and shops that sell products within the application, for the users to become aware that the app delivers to their area. The consumers are therefore prompted to enter their delivery postcode to check for availability as shown in Figure 5.1.

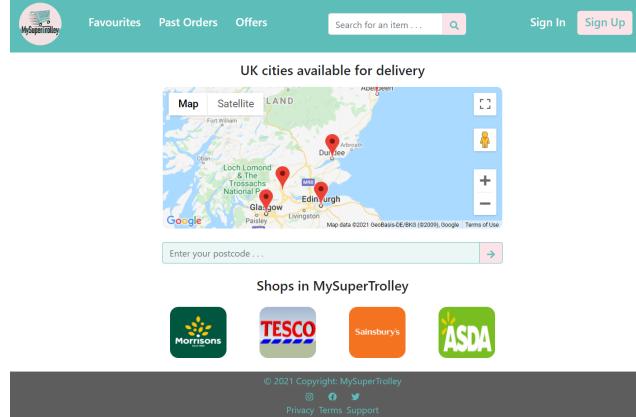


Figure 5.1: Landing Page of MySuperTrolley

The functionality for checking the postcode was made in the presentation layer using '**uk-postcode-validator**' library. By importing the `isValid("N7 7AJ")` function, the app first checks if the postcode entered exists in UK. Afterwards, for each city available for delivery, namely, Glasgow, Edinburgh, Stirling, Dundee and Aberdeen, a list with the first part of the postcode was created. For example, Glasgow has postcodes that start from G1 to G80. Since all the postcode lengths of those cities vary from 5-7, the first 3/4/5 characters were checked. If this check returns true, the postcode is available for delivery as shown in Figure 5.2a and the user is then prompted to select a slot for delivery. If the location is not available for delivery the application informs the user, as shown in Figure 5.2b. In both cases, the application allows the user to continue in order to compare prices between shops, or collect from the stores.

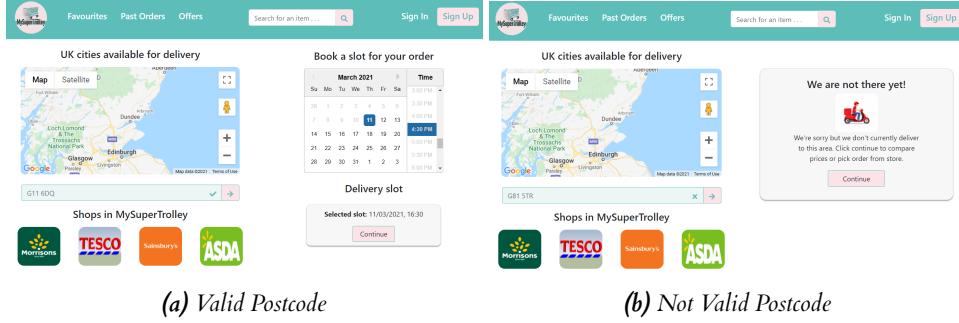


Figure 5.2: Postcode Validation

5.2 Price Comparison

Another important issue discovered during background research, was that the item details were sometimes insufficient for the consumer to completely understand the item or the delivered item did not meet the consumers expectations. Therefore, as shown in Figure 5.3, the title, description and photo of the item is presented.

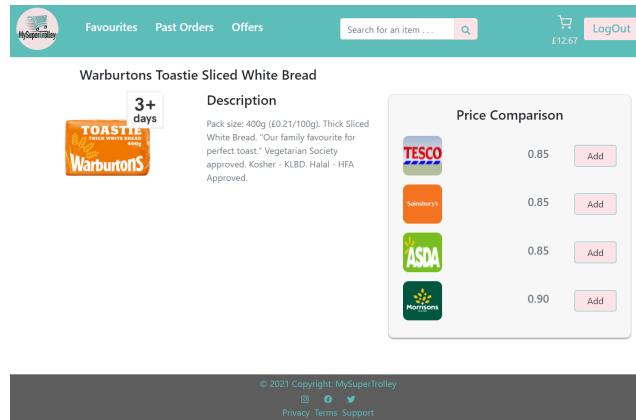


Figure 5.3: Item Details Page in MySuperTrolley

Additionally, since the main aim of MySuperTrolley is for the user to be able to add items from more than one shop in a single order, there is a price comparison section for the user to select the cheapest shop that sells the exact item they are looking for. All this information is taken from the database. From the Appendices in section H it can be observed that the item table has a composite primary key with the itemID and the brandID uniquely identifying each row in the database. Therefore, in order to display all this information for the item presented in the figure, the item details component in React calls the item service in order to make a GET HTTP request using '**AXIOS**' library with the itemID as an input parameter. Then, the item controller of Spring Boot handles the request by calling the corresponding service. The service then calls the repository which is responsible to get the information needed from the database. We want to select all the shops that sell this item and sort it by price, so that the shops displayed are from lower to highest price. The request to the database from the item repository to accomplish this information is given below:

```
@Query(value = "SELECT * FROM item WHERE itemID = ?1 ORDER BY price",
nativeQuery = true)
List<Item> findItemsByItemID(int item_id);
```

5.3 Features for people with limited technology skills

From section 2, it was found that high experience with technology can result in more orders purchased online. Furthermore, during COVID-19, people who were older and with limited technological knowledge, were found to be the most disadvantaged group. Therefore, MySuperTrolley provides two functionalities that will make the ordering process of consumers with limited technological skills faster.

The first feature is the Favourites page as shown in Figure 5.4. This feature is implemented in order for users to select their "favourites", meaning items that they will usually purchase. For example, users can add 30 items, and only select items from this page for their orders. The "Favourites" feature is implemented by creating a table in the database, as shown in Appendices in Figure H. This table just contains the favID, userID and a comma separated string favlist with the itemID of the items they consider them as favourites.

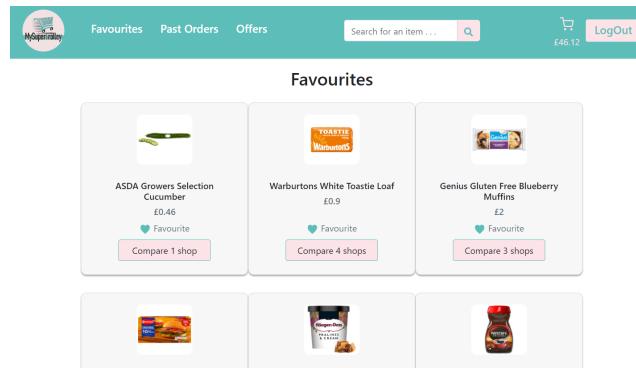


Figure 5.4: Favourites Page in MySuperTrolley

Figure 5.5 shows the code of the retrieval of the favList of the user. A Post request is made using **AXIOS** with the id of the user given as an input parameter in order to retrieve the existing favourites list of items of the user as a comma separated string. If the first element is a comma, it is removed in order to not get an empty item id as a first item in the favourites page. Then the string is split in order to get the list of all the favourites item ids.

```
if (JSON.parse(localStorage.getItem("user")) !== null) { // retrieve fav items for authenticated user
    FavouredService.findFavbyUserID(JSON.parse(localStorage.getItem("user")).id).then(r => {
        if (r.data.length === 0) {
            this.setState({favList: []});
        } else {
            var favList = r.data[0]['fav_list'];
            if (favList[0] === ",") { // remove , if it is the first element
                favList = favList.substring(1, favList.length);
            }
            var numbers = favList.split(",") // split item ids
            this.setState({fav_numbers: numbers});
        }
    });
}
```

Figure 5.5: Retrieve favourite items for current user

Figure 5.6 shows the code of how the component handles the addition or removal of an item in the favList. When the user clicks on either the full or empty heart icon, representing if the item is a favourite or not, the 'addFav' function with the id of the item is given as an input parameter is called. Firstly a check that the user is authenticated is made, otherwise it redirects the user to the login page for authentication. After that, the id is checked on whether or not it already exists in the favList, in order to either remove it or to add it. Finally, the list is then converted back into a comma separated string and a Post request is made to save the favList back to the database.

```
addFav(id) {
  // If the user is not authenticated redirect them to login page
  if (!authService.isLoggedIn()) {
    this.props.history.push("login");
  } else {
    if (this.state.fav_numbers.includes(id)) { // remove item from fav
      const index = this.state.fav_numbers.indexOf(id)
      this.state.fav_numbers.splice(index, 1)
    } else { // add item to fav
      this.state.fav_numbers.push(id)
    }
    if (this.state.fav_numbers.length === 1) {
      FavouritesService.saveFavList(JSON.parse(localStorage.getItem("user")).id, this.state.fav_numbers[0])
    } else {
      FavouritesService.saveFavList(JSON.parse(localStorage.getItem("user")).id, this.state.fav_numbers.join(","))
    }
    this.showSubCatItems(this.state.subcat_ids)
  }
}
```

Figure 5.6: Add/Remove favourite item

The second feature is the Past Orders page, as shown in Figure 5.7. This feature is separated into scheduled/previous orders with the ability to edit/view order accordingly. The main reason for this feature is that the user can order the same exact order with the same items in the future. Users can also click and add all the items in every order, they can also remove items from their basket or add further items in their basket to complete their order. All the information is saved into the orders table, by making a Post request when the users placed the order. When the delivery date is in the future, the order is placed in the Scheduled orders otherwise in the Previous orders. However, the edit/view button is not implemented and it is considered as a future work for MySuperTrolley.

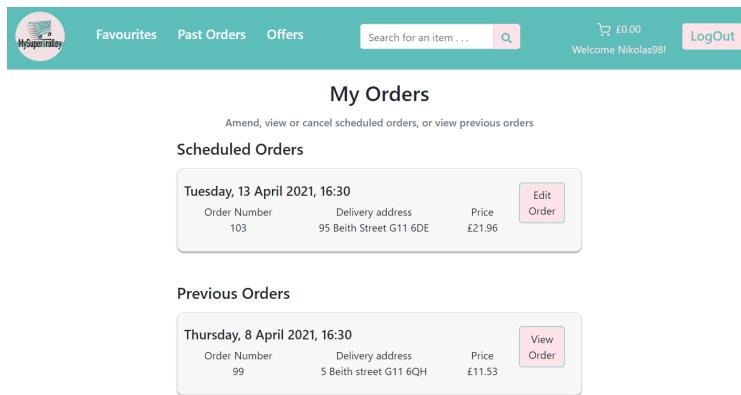


Figure 5.7: Past Orders Page in MySuperTrolley

5.4 Item Quantity

When making an order, consumers often select more than one quantity of the same item, for example three cucumbers. The quantity is implemented with the user having the ability to select the same item, up to 10 times, using the **react-numeric-input** library. There are three different lists that are used for handling the addition of items in the basket, namely basketList, quantitiesList

and prices having the items ids, quantity and price of an item respectively. When the state of the quantity is changed, a function 'addItemToList' is called with the quantity, item id, brand id and price given as input parameters. The function first checks if the user is already authenticated, otherwise it redirects the user to the login page. Then the index of the item selected in the basket of current user is found. There are three different options. If the quantity of item is zero, the item is removed from the three lists. If this is not the case, and the index found is null, meaning the item is not yet in the basketList, the corresponding details of the item is added in the three lists, otherwise, the quantity of the already existing item in the basket is updated. The code for the implementation of the addition of items using quantities as well as a presentation of the page in mobile version is shown in Figure 5.8 below:

```
// if the user is not authenticated redirect them to login page
if (!AuthService.isLoggedIn()) {
  this.props.history.push("/login");
}

// get the index of the item id in the brand id list
var iid_index = this.state.basketList[bid].indexOf(iid);

if (quantity === 0) { // remove the price, quantity and item id
  this.state.basketList[bid].splice(iid_index, 1);
  this.state.quantitiesList[bid].splice(iid_index, 1)
  this.state.prices[bid].splice(iid_index, 1)
  this.showShopItems()
} else {
  if (iid_index === -1) { // not yet in the lists
    this.state.basketList[bid].push(iid);
    this.state.quantitiesList[bid].push(quantity);
    this.state.prices[bid].push(price);
    this.showShopItems()
  } else { // update quantity of specific item
    var temp_quant = this.state.quantitiesList;
    temp_quant[bid][iid_index] = quantity
    this.setState({quantitiesList:temp_quant})
  }
}
```

Figure 5.8: Handling item quantity

5.5 Order Preview

Before the user places an order, there is an order preview page where the user can check the important details of their order, before proceeding to the checkout page. Such information consists of the items separated in different cards according with their shop selected, quantity of the item, along with the total prices for each shop, a calendar reminding them their slot selected at the start and a chance to change it. Additionally, the total price of the order along with the extra fixed delivery cost is included, as shown in Figure 5.9. In order to implement this page, a GET request is made to select all the items that were added into the basket, separated by their brand id in different cards and calculate the total price for each card, along with the total price of the order. The shop cards are shown in the page, only if the user added an item from the specific shop into the basket.

A Datepicker imported in react using the **react-datepicker** library used for the calendar to book the slot. All the previous dates and times were disabled, as previous dates cannot be selected for deliveries. Additionally, when a user books a slot, it is saved in the 'this.state.startDate' variable and it is displayed in the card underneath. If the user is a previous visitor, thereby they have previously entered their postcode and have visited the Valid Postcode page, then the booked slot is shown as a default in the order preview page. However the user has the opportunity to change the slot again.

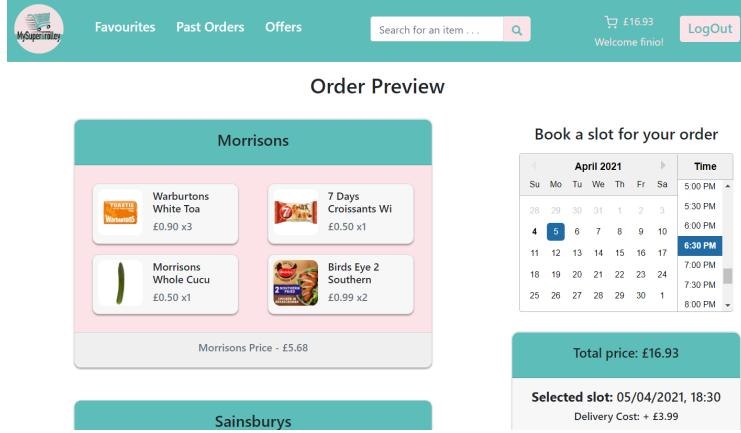


Figure 5.9: Order Preview in MySuperTrolley

5.6 Checkout process

The checkout page is a mandatory feature to complete an order. The delivery and payment details are taken from the user in the form of using bootstrap forms. The address, city, postcode and country are saved when the order is placed along with the total price as shown in Figure 5.10, in order to be presented in the past order page. This is implemented using a POST request, using a json parameter, called basket which saves and then renders the information by a GET request into the past orders page as shown below:

```
{"basket": {"1": [1, 9, 13, 42], "2": [2, 43, 43, 48, 2], "4": [7, 44], "5": [42, 4, 32, 33]},  
"total_price": "33.61", "address": "82 Beith Street", "postcode": "G11 6DQ"}
```

The payment method can be completed with three options, namely, credit/debit card along with Paypal. The most optimal way of payment to be integrated in such applications is Paypal. However since the items in MySuperTrolley are mimicking the functionality of the application and they are not real, the payment is not actually made. This feature is considered in the future work once the website is established and actual items are presented into the application.

Delivery address

First Name	Last Name	
Andrew	Brown	
Email (Optional)		
andrewb@gmail.com		
Address		
Flat 8 14 Earlston Place		
Country	City	Postcode
United Kingdom	Edinburgh	EH7 5SU

Payment

<input type="radio"/> Credit card	<input type="radio"/> Debit card	<input checked="" type="radio"/> Paypal
		
Name on card	Credit card number	
Andrew Brown	X0000000000000000	
Full name as displayed on card		
Expiration	CVV	Total price: £36.80
xx/xx	xxx	<input type="button" value="Place Order"/>

(a) Delivery Details
(b) Payment Details

Figure 5.10: Checkout Page

5.7 Mobile Responsive

MySuperTrolley application has a non-functional requirement, the access of application by different devices such as desktop, tablet and mobile phone. Therefore, the app is implemented to also be mobile responsive, as shown in Figure 5.11.

Figure 5.11, shows four different pages in the mobile version of MySuperTrolley. The first picture shows the landing page, as discussed in 5.1, the second picture shows the category page along with filtering into subcategories for easier selection of items. In order to implement the functionality of the filtering, all the subcategory ids from the selected category are rendered from the database, and an empty list is initialised. When the list is empty, all the items of that category are displayed in the page. Once a user clicks on a subcategory, the subcategory id is added/removed from the list in order to display the subcategory items. As illustrated by the picture, the bakery filter is selected and relevant items are displayed. Picture three, shows the item details page as discussed in 5.2 section, where it shows that Dettol is in offer in Morrisons and Tesco shops and two items are added to the basket from Morrisons. Finally, the last picture shows the items along with their details in the basket page, as well as the total price of the basket.

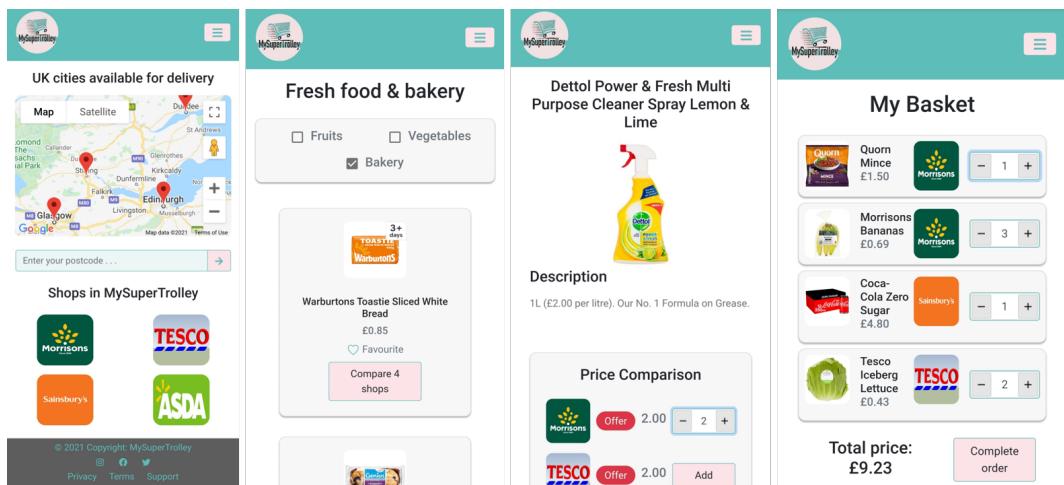


Figure 5.11: Web App Responsive

As aforementioned in the background research section, grocery shopping can be done at any time of day or completed even while doing other work. A mobile user-friendly application can help consumers to make an order when they are not at home without access to a laptop or Desktop.

5.8 Deployment

Deploying the application was crucial to undertake the usability evaluation, so that the participants only need to click on the website link. Otherwise, participants should download and execute files with most of the users not having a Computer Science background.

Initially, deployment started by using the Heroku platform. The deployment in Heroku would be easier if Github was also used. However, the only way to deploy the application in BitBucket was by using the pipeline. A different branch for deployment was created as all the server files needed to be in the root directory and this would cause confusions in the master branch.

Before deploying the application the front-end was executed separately by using 'npm start', to show the static version of the app and the back-end using Tomcat in spring-boot to render

the REST API. For deployment, this was completed only by running the back-end. For the implementation of such functionality, some dependencies were added in the pom.xml file, in order to build the front-end modules and add them in the static file in spring-boot.

After those changes were made, some refactoring was required as spring boot does not understand HTML hyperlinks and uses only React Links. The Routes were changed into Hash Routes for the separation of back-end API requests, due to confusions when the application was deployed.

Another issue was that Heroku only provided a free PostgresSQL for the database. So, when the .sql files were exported from HeidiSQL database, it did not work due to the differences of syntax between Postgress and MySQL thereby ClearDB was selected. After trying to fix some errors I managed to deploy it. Unfortunately, after some navigation through the app, it only showed the static details due to an error which occurred with reaching the maximum number of connections to the database by using the free version.

The only possible way to accomplish deployment was to switch into PostgresSQL database since Heroku had the most advantages of the free version. This meant that the new database needed to be created from scratch using the .sql file but since the syntax was slightly different those changes had to be made manually. Finally, after some refactoring from MySQL to PostgresSQL syntax in the back-end when making the API requests the application was deployed successfully.

5.9 Demo Presentation video

The following link shows a demo with all the features of the application implemented. Try to download the video because the quality online from the clicked link is not very good.

https://drive.google.com/file/d/1esqcxX8_rPXeChi4RXkkubysroNtWrXi/view

6 | Evaluation

This section will evaluate MySuperTrolley. The two main different measurements used for evaluating the project were namely, unit testing and user evaluation. Unit testing provides a test table with the list of all the features tested and whether they have passed the test or not, along with the code coverage. Moreover, this chapter will discuss the full user evaluation process analysing followed by a discussion of the results.

6.1 Unit Testing

In this section, some unit tests were created to ensure that the application performed as expected. In addition, the unit tests were also used to check the correctness of the implemented code for MySuperTrolley. Most of the functionality of the back-end is tested, ensuring that the rendering of the API data is correct. A table with 34 different tests was created, to try and test the correct and wrong data, rendered by the call of different methods in the application. A table showing the different features tests along with the result of the test passed or not passed, is created. A sample of the table is shown in the Table 6.1 while the whole table is given in the Appendices at section M.

Unit Testing		
Test Cases	Passed	Not Passed
Brand Controller		
Retrieve all brand shops	✓	
Items Controller		
Retrieve item using the item id	✓	
Retrieve item using the brand id	✓	
Retrieve item using the item id and brand id	✓	
Retrieve distinct items by category id	✓	
Retrieve items that are in offer	✓	
Retrieve number of shops sell each item	✓	
Category Controller		
Retrieve all categories	✓	

Table 6.1: Sample of Test Cases

Seven different test files were created, using the **MockMvc** framework, to avoid any actual insertion or altering of data in the database by mimicking the functionality of the different controller methods to test them. An example of tests' code is shown in Figure 6.1. When all the tests were finished, a new test suite file, responsible for running all seven test java files in one, called 'MySuperTrolleyTests' is created. This is made using the **junit.runners.Suite** and **junit.platform.runner.JUnitPlatform** libraries.

```

@Autowired
MockMvc mockMvc;

@Test
public void getItemsByItemIDBrandID() throws Exception {
    // Given
    int expected_item_id = 15;
    int expected_brand_id = 4;
    String expected_title = "Sainsburys Large Pineapple";
    int expected_size = 1;

    // When
    final ResultActions result = this.mockMvc.perform(
        get( urlTemplate: "/api/items/{itemid}/{brandid}", {expected_item_id}, {expected_brand_id})
        .andExpect(status().isOk()));

    // Then
    result.andExpect(jsonPath( expression: "$[0].title", is(expected_title)));
    result.andExpect(jsonPath( expression: "$[0].item_primary_key.item_id", is(expected_item_id)));
    result.andExpect(jsonPath( expression: "$[0].item_primary_key.brand.brand_id", is(expected_brand_id)));
    result.andExpect(jsonPath( expression: "$.*", hasSize(expected_size)));
}

```

Figure 6.1: Test Case example

The code coverage was performed to check the percentage of the code being tested, using the test cases created. The **IntelliJ IDEA** library is used to calculate the code coverage measuring the percentage of classes, methods and line of code that were tested, as shown in Figure 6.2. Google suggests that 60% of lines of code covered in tests is 'acceptable'. MySuperTrolley has 55%. Due to the time restrictions, not all functionalities were tested as an ideal application in production. However, all the tests were passed, but in case the development of MySuperTrolley continues to be taken in further implementation, more tests will be required to cover every single functionality of the application.

Element	Class, %	Method, %	Line, %
MySuperTrolley	89% (49/55)	45% (125/273)	55% (527/949)

Figure 6.2: Code Coverage

6.2 User Evaluation

The MySuperTrolley application is designed to be used across the UK or even worldwide. Thereby, future possible users evaluating the application was a crucial evaluation metric that required to be taken into consideration during the evaluation process. The evaluation satisfied the School of Computing Science ethics checklist provided in the Appendices at section A.

6.2.1 Methodology

A Google form survey was created and was used as the main form of collecting participant information. The URL link of the deployed application of MySuperTrolley app was also included within the survey. Firstly, the survey required the participants to answer on which device and operating system they were using during the completion of the tasks for the survey. The survey was split into three different sections. The first section required participants to complete 20 tasks, by navigating through the application and answer on how simple the task was to complete. Afterwards, participants were promoted to answer 10 multiple choice questions. The answers from the multiple choice questions were collected to calculate the average System Usability Scale (SUS) score. Finally, participants were asked to answer some open-ended questions about

what they liked/disliked in the application and what extra features they would like to see implemented within the application.

From 37 responses collected after the survey, 48.6% were using their Laptop, 32.4% a Smartphone, 16.2% a PC and only 2.7% a tablet during the completion of the survey, as shown in Figure 6.3. The operating system used was found to be 52.8% Windows, 27.8% iOS and 19.4% Android.



Figure 6.3: Device & Operating system used

6.2.2 Tasks

Some questions asking on the simplicity of completing the task will be briefly discussed below. All survey questions and answers can be found in the appendices at section N.

Task 1 - Enter the postcode of your delivery location to check if MySuperTrolley delivers to your location (If your postcode response was not valid, try to enter a postcode from the cities that are marked in the map shown in the page just to complete the following tasks. e.g G1 4JE).

A total of 94.6% of the survey takers successfully completed and the remaining 5.4% partially completed the task. All participants found the task easy to complete. The task was simple and was designed based on applications that have similar functionalities. However, the two participants that partially completed the task, may be living in locations that MySuperTrolley does not service to according to the postcodes previously mentioned, did not see the map displaying the areas that the app services or did not see the example postcode given in the parenthesis of the task question.

Task 9 - From the category navbar find the item "Ben & Jerry's Ice Cream Cookie Dough" and add it to your basket. Select the shop that has the lowest price.

A total of 97.3% of participants successfully completed the task while only 2.7% partially completed the task. 86.5% of participants found it easy to complete, whilst 13.5% found it hard. MySuperTrolley application had eight independent categories. The "ice cream" was placed under the "sweets" category and not under "frozen". However, after five people found the task hard to complete this might indicate that some categories are interlinked and are not as distinctive as predicted.

Task 13 - Access favourites page. Check that the three items you previously added are present.

A total of 83.3% of participants successfully completed task 13, 3.9% of them partially completed the task and the remaining 2.8% did not manage to complete the task at all. 94.4% of the participants found it easy to complete and 5.6% found it hard to complete. There was an issue during the implementation while adding and removing items from the favourites section which was identified when participants were making comments during the evaluation section. Some participants encountered problems when trying to access the favourites page while others did not. The favourites list was saved as a comma separated string, as aforementioned in the implementation section (e.g favList = '37,4,41,2,67'), and then split with ',' to get the item ids. When a participant removed the first element from the string the updated favList = '4,41,2,67'

was split into [,1,41,2,67] and the GET request made to retrieve the items from the item ids in the list was therefore trying to find an item with empty item id. As a result, nothing was retrieved and the participants were seeing an empty page. The problem was immediately fixed after identifying it, by removing the first character if the first character was a comma.

Task 20 – Visit Past Orders page. Check that the order you just made is placed in the scheduled/past orders along with the details of your order (See only delivery address, price, delivery time). The edit/view order button does not work as it is considered as future improvement.

A total of 83.3% of participants successfully completed this task, whereas 16.7% partially completed it, as shown in Figure 6.4. 86.1% found it easy to complete and 13.9% found it hard. Participants that did not manage to see the placed order was because the order of using the POST request was not made successfully. 5/6 of the responses that did not manage to see the order, were using iOS operating system, some of them using iPhones, iPads or Macbooks. When looking at the orders made from the database, there were 6 less orders than the number of participants that went through and the orders that were missing where from participants using an iOS operating system. There was also a participant that was using iOS that marked the task as complete. One possible reason for this discrepancy may be the different versions or updates of the iOS system that the participants were using. Most of the fixes of the newer version of the iOS were around security. A large number of hacks are made through POST requests thereby with the new update the POST request might have not been allowed with the free version of Heroku deployment website.

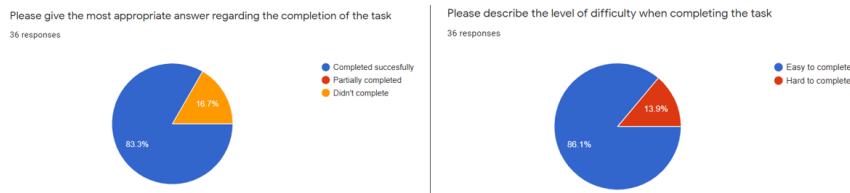


Figure 6.4: Completion & hardness status

Overall most people managed to successfully complete the tasks and find navigating through the app easy for the completion of the tasks.

6.2.3 Multiple Choice Questions – Quantitative Information

For gathering quantitative feedback information when evaluating the usability on MySuperTrolley, ten multiple choice questions were made to calculate the SUS score for each participant. The questions were found from Thomas, N. (2019), included both positive and negative questions where participants were required to answer them using a five-point scale system ranging from Strongly-Disagree to Strongly-Agree. The odd numbered questions were positive and the even numbered questions were negative. To calculate the SUS score for each participant, from all the positive questions 1 point is subtracted from the score of each question and from all the negative questions we subtract the value that the participant gave from 5 for each question. The scores are then added and multiplied by 2.5 to find the total SUS score. When participants had a score 80.3 or higher, it indicated that participants loved the application and would likely recommend it to their friends. When participants got 68-80.2 it meant that they found the website "okay" but the website could use improvement. When participants got a score of 67 or lower it means that the usability feature requires improvement to fix the issues (Thomas, N. 2019). Questions used along with their responses can be found in the Appendices at section N and the total SUS score for the first 10 participants is shown below in Figure 6.2. The SUS score for all 37 participants can be found in Appendices in section O.

Participants	SUS Score calculation										SUS Score
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	
1	4	1	5	1	4	1	4	2	5	1	90
2	4	1	5	1	5	1	4	1	5	1	95
3	5	1	5	1	5	1	5	1	5	1	100
4	5	2	5	1	5	2	4	1	5	1	92.5
5	5	1	5	2	5	2	4	2	5	1	90
6	5	1	5	2	5	1	5	1	5	1	97.5
7	5	2	5	2	4	1	3	2	3	1	80
8	5	1	5	1	5	1	5	1	5	1	100
9	3	2	3	2	4	3	4	2	4	1	70
10	5	1	5	1	5	1	5	1	5	1	100

Table 6.2: SUS Score

The average SUS score from all the participants was found to be 91.4 which indicating that participants were overall satisfied with the application and will most likely recommend it to their friends or family. However, three of the participants had an SUS score below 80.3 indicating that the website still requires some improvements.

6.2.4 Open Ended Questions – Qualitative Information

The third section of the evaluation survey required people to answer some open ended questions. Such questions included: what do people like least/most about MySuperTrolley, which feature of MySuperTrolley was the least/most important to them, and if there are any improvements they would suggest to the design of MuSuperTrolley.

What people like least and were the least important feature for them in MySuperTrolley

People indicated that the feature that they liked the least, were the icons used for the categories, instead of pictures, and that they did not like the colour scheme of the app either. Some other answers included that the categories section disappeared every time they clicked on a category and that changing between categories was not efficient. This was because in the mobile version of the application the filter of some categories would cover the better part of the page and the design was not very pleasant. In contrast, many people noted that every feature implemented was important and nothing was least important to them.

What people like most and were the most important feature for them in MySuperTrolley

The most important features that were noted within the responses were the opportunity to add items to favourites, being able to access the past orders, categorise items with categories and filter for subcategories. Additionally being able to compare prices between shops and purchase items from multiple stores simultaneously as well as easy and simple navigation to all functionalities of the application. Participants noting the favourites and past orders page, stated that this features makes it easier to buy frequently purchased items which would mean that the shopping list would be unnecessary. Another common answer was the ability to compare prices between shops and being able to make an order with items from more than one shop. Participants said that these functionalities reduced the necessity to visit multiple websites for their grocery purchases as the application provides the opportunity to both compare prices between shops and select the item with the cheapest price to add to their basket which may include items from different stores. Favourites and past orders page combined together with the easy navigation of the app were

noted as helpful for people with limited technical skills.

Finally, responses also included that features that were liked by participants were that they could choose categories and filtering for subcategories, the application was very responsive, good that the first thing you check before continuing to the website is the postcode and availability, the offers section and the easy checkout process. All these responses were found to be significant in their presence in MySuperTrolley.

The aims for MySuperTrolley, as described in section 1.2, were all satisfied and covered from comments of the participants, making the goals of MySuperTrolley successfully achieved.

Improvements you would make to this design

An important point that was mentioned by many participants, was the absence of the item quantity. For example if someone wanted to add three croissants in their basket they were not able to. This was taken into consideration and was implemented after the user evaluation, as already described in section 5.4.

Another important issue was the absence of the search functionality. This is an important functionality that is mandatory and extremely useful to be present in grocery websites as it provides the opportunity for users to search for products. However, in order to build a very effective and efficient search engine would require a lot of research and time for implementation and thereby this will be further considered out of the scope of this project.

A further issue that was identified was that the basket total on the top menu was not correctly updating when items were added or removed from the basket if the page was not refreshed. Additionally, participants preferred to see their username displayed to authenticate and confirm that they have signed in their account. Both suggestions were implemented. The cart icon with the total place is now placed in the navbar and automatically updates the total price when items are added or removed from the cart along with a 'Welcome <username>! indicating that the user has logged in the application.

Furthermore, order confirmation was not present and the app redirected survey takers to the login page, without notifying them if the order was made or not. Additionally, the warnings in the log-in and sign-up page were not clearly readable by a user as they showed unreadable errors. Those comments were also taken into consideration and a confirmation of the order was therefore added along with warnings specifying that a users username or password are not correct when there are problems with data authentication.

The requirement validation table for Must Have and Should have requirements is added in the appendices in section P.

6.3 Evaluation Summary

The unit tests clearly tested some of the functionalities of the code, however, more testing that will make the functionality and correctness of any code is always helpful and is also considered a good practise which can improve the application further. Such practices can be done in the future. Overall, out of 37 responses, a significant percentage indicated a positive inclination to the application where participants liked and enjoyed the navigation of the application as well as the features implemented and would also consider using the application if it would be further developed and actualised in the future.

7 | Conclusion

This chapter will summarise all the chapters included in this paper to complete MySuperTrolley project. Additionally, the chapter will discuss some limitations, including issues that restricted the project from going under further implementation along with possible future work that can enhance the current implementations of the application. A final reflection will be outlined regarding the whole process and procedure of the project.

7.1 Summary

MySuperTrolley is a responsive web application that was designed and implemented to provide the opportunity for users to purchase items simultaneously from multiple shops in a single basket, compare product prices across different grocery shops and then select the item with the cheapest price. The application also enables users with various levels of technological experience to easily and simply navigate through the application to successfully satisfy their grocery needs. The application was implemented and deployed to be evaluated by people across the world that would possibly be future MySuperTrolley users. The system is evaluated using two main methods, namely unit testing and user evaluation. Unit testing was used to manually test the correctness of the functionalities implemented. User evaluations were used for new users to perform tasks by navigating through the different features implemented while also being able to share their experience of testing MySuperTrolley by answering some quantitative and qualitative questions. The overall results towards MySuperTrolley was found to be positive as well as indicating that the application is highly usable based on the average SUS score calculated from the collected responses of the quantitative questions. Finally, participants stated that the application, if it is to undergo further implementation and is actually deployed, will be used by a lot of them due to its convenience and for allowing people to purchase items from multiple grocery stores simultaneously, through a single order and have them delivered at their home. The deployed application can be accessed using the following link: <https://mysupertrolley.herokuapp.com/>

7.2 Future work

Limitations existed that prevented the implementation of MySuperTrolley from being able to be deployed into the real-world. A main limitation was that the APIs of different grocery shops were not made publicly available. Websites wanted to focus their efforts to maintaining critical systems, something that was done due to COVID-19 (devportal.tescolabs.com 2021). Therefore, the items presented in MySuperTrolley were manually added in a database, to mimic the functionality of how the application would actually work if the APIs were public information. Therefore, in the future if MySuperTrolley would continue with its implementation, the different grocery stores could be contacted for their approval and to gain access to their APIs.

The aim of the application is to have multiple grocery shops selling items to MySuperTrolley. However, some areas cannot be geographically satisfied with the shops included in MySuperTrolley. Some locations may not have availability for next-day delivery due to geographic restrictions

thereby users should be made aware early on during their process. By having the user being aware, early on, what availability they have and from which shops, user experience improves. The order preview page could also inform the user either to delete items from stores that are very far away from them geographically or cannot deliver to their location on their selected delivery date, so that they can be able to change delivery time or shop selected. The Wireframe prototypes of how it will look like are shown in appendices in section L.

Another feature that was not implemented, was the search functionality. This feature was not implemented due to time restrictions. A possible future internal search engine functionality can be implemented by trying to make searching items within the application as efficient and effective as possible. One way for the implementation of this feature will be to apply tokenisation, stopword removal and stemming to the item titles and the query made by the user. Finally once both the query and the item titles have been modified, the items can be ranked according to the number of words from the query that is found in the item title as a percentage of the total query. For example, the original text would be "Lager Beer Cans". The application includes items with titles "Heineken Premium Lager Beer Cans", "Coke Zero Cans", "Amstel Lager Beer Cans". After the modification of query and item titles something like "[lager, beer, can]" can be added, for query and "[heineken, premium, lager, beer, can]", "[coke, zero, can]", "[amstel, lager, beer, can]" for item titles. Afterwards, a higher weight could be assigned to words that participate both in query and item titles. So for this example, the third item has 3/4 words and those 3 words are multiplied by the extra weight, meaning that the rank of the items will be first the amstel, then heineken and finally the coke when the search is made by any user.

The past order page displays the scheduled and previous orders according to the delivery date. The scheduled orders have an "edit basket" button while previous orders have a "view basket" button. However, further implementation for that was not carried through, due to time restrictions. Potentially, scheduled orders can be edited by users up to 24 hours prior to their scheduled delivery slot. The "view order" button for previous orders, could render the items from that order again and an "add" button could also be added for users who want to add all the items for their next order.

Finally, the payment method was a functionality that was not implemented either. This was because without any real orders or items it would not make sense for it to be fully implemented. Participants would also be reluctant to enter their banking information and therefore this step was skipped. However, if the application would be deployed in the real-world then this functionality would require implementation if the aforementioned points can be taken into account. Payments using paypal, Google/Apple pay will enhance the security of the payment method and more users could visit MySuperTrolley. Login with Fb/Google and adding customer support with either live chat or FAQs could also implemented in the future.

7.3 Reflection

MySuperTrolley project firstly taught me how to conduct a research to gain knowledge about what exists within the real world in the realm of online shopping and more specifically in the grocery shopping. Additionally, the project taught me how to individually organise a large amount of work and split it into smaller tasks to be easily manageable and successfully finish a project I am satisfied with. Using software engineering processes, could make the organisation of work much easier. Being able to build an application from scratch with technologies that I was using for the first time, enhanced my programming experience. The user evaluation taught me that it is a mandatory and extremely helpful when the main usage of the application is for actual people. Lastly, writing a dissertation and explaining the most important topics of the project taught me how to stay on topic and explain the most important and specific points about the project conducted.

Appendices

A Ethics Checklist

**School of Computing Science
University of Glasgow**

Ethics checklist for 3rd year, 4th year, MSci, MRes, and taught MSc projects

This form is only applicable for projects that use other people ('participants') for the collection of information, typically in getting comments about a system or a system design, getting information about how a system could be used, or evaluating a working system.

If no other people have been involved in the collection of information, then you do not need to complete this form.

If your evaluation does not comply with any one or more of the points below, please submit an ethics approval form to the Department Ethics Committee.

If your evaluation does comply with all the points below, please sign this form and submit it with your project.

-
1. Participants were not exposed to any risks greater than those encountered in their normal working life.

Investigators have a responsibility to protect participants from physical and mental harm during the investigation. The risk of harm must be no greater than in ordinary life. Areas of potential risk that require ethical approval include, but are not limited to, investigations that occur outside usual laboratory areas, or that require participant mobility (e.g. walking, running, use of public transport), unusual or repetitive activity or movement, that use sensory deprivation (e.g. ear plugs or blindfolds), bright or flashing lights, loud or disorienting noises, smell, taste, vibration, or force feedback

2. The experimental materials were paper-based, or comprised software running on standard hardware.

Participants should not be exposed to any risks associated with the use of non-standard equipment: anything other than pen-and-paper, standard PCs, mobile phones, and PDAs is considered non-standard.

3. All participants explicitly stated that they agreed to take part, and that their data could be used in the project.

If the results of the evaluation are likely to be used beyond the term of the project (for example, the software is to be deployed, or the data is to be published), then signed consent is necessary. A separate consent form should be signed by each participant.

Otherwise, verbal consent is sufficient, and should be explicitly requested in the introductory script.

4. No incentives were offered to the participants.

The payment of participants must not be used to induce them to risk harm beyond that which they risk without payment in their normal lifestyle.

5. No information about the evaluation or materials was intentionally withheld from the participants.
Withholding information or misleading participants is unacceptable if participants are likely to object or show unease when debriefed.
6. No participant was under the age of 16.
Parental consent is required for participants under the age of 16.
7. No participant has an impairment that may limit their understanding or communication.
Additional consent is required for participants with impairments.
8. Neither I nor my supervisor is in a position of authority or influence over any of the participants.
A position of authority or influence over any participant must not be allowed to pressurise participants to take part in, or remain in, any experiment.
9. All participants were informed that they could withdraw at any time.
All participants have the right to withdraw at any time during the investigation. They should be told this in the introductory script.
10. All participants have been informed of my contact details.
All participants must be able to contact the investigator after the investigation. They should be given the details of both student and module co-ordinator or supervisor as part of the debriefing.
11. The evaluation was discussed with all the participants at the end of the session, and all participants had the opportunity to ask questions.
The student must provide the participants with sufficient information in the debriefing to enable them to understand the nature of the investigation.
12. All the data collected from the participants is stored in an anonymous form.
All participant data (hard-copy and soft-copy) should be stored securely, and in anonymous form.

Project title MySuperTrolley

Student's Name Nikolas Finiotis

Student's Registration Number 2290936f

Student's Signature Nikolas Finiotis

Supervisor's Signature [Signature]

Date 20/10/20

B Comparison of existing similar Applications

Website	Features	Strengths	Weaknesses
Trolley (Web App)	1. Categories 2. Title, photo, description, related products 3. Compares prices with all supermarkets that save this specific product 4. Displays how many people rate a specific product for each shopping with the rating 5. Search a product 6. Categories related to the searched product 7. Buttons to visit each website to buy the products	1. Compare prices of over 10.000 products across supermarkets and shops across the UK. 2. Rating of products before buying 3. Displays title, description of product and related products 4. Categories and search help you find the product you need	1. The website redirects you to the specific app that sells the item (no basket) 2. Cannot pay in this site 3. The categories related to the searched product are not well displayed 4. In the home page, it displays the "Today's Biggest Sale" but the products are the most purchased by customer nor their customer review are the best. So I don't know why this section is placed there. It is not helpful at all.
Buymie (Android & iOS mobile app)	1. Choose between 3 shops to see their products. 2. My Account 3. FAQ 4. Ask a question, help 5. Previous orders 6. Buymie plus 7. Search for a product 8. Categories 9. Title, photo, description 10. Add note for the shopper 11. Add to favourites 12. Cart	1. You can shop products from lidl, tesco, dunnes store 2. The app has cart 3. Refer a friend and get both 10 euro for the next payment 4. Above 40 euro the delivery is free (3 euro otherwise) 5. Buymie Plus (8.33 euros per month). Includes unlimited free delivery, free returns and local shoppers, delivery as soon as one hour 6. Delivery slot available in the same day 7. Categories and search help you find the product you need 8. You can add your weekly purchased items as favourite products. You can easily send them for future orders. 9. Customer Service 10. See previous orders	1. You cannot add to the cart items from different store 2. Delivery only in ireland and in uk. However I didn't manage to find an address where the order can be delivered in the uk apart from bristol. So limited delivery area. In UK it does allow to select delivery slot. 3. Description is not there 4. Sometimes not aware prices of products between the three shops 5. Minimum order 30 euro
DEVO (Web App, Android & iOS mobile app)	1. Enter delivery postcode 2. Choose between several shops to see their products. 3. My Account 4. Previous orders 5. Search for a range of products, cost of delivery and range of minutes needed for the delivery 6. Cart 7. Categories (with sub-categories) 8. Title, photo, related products 9. Google pay payment	1. You first need to put your location. This finds if they can deliver your food to your location. If yes, then it displays different grocery stores available around you a range of time it needs for the food to be delivered to your address along with the number of products the shops sales and delivery cost. It displays the opening hour if the shops is closed at the moment. 2. Displays only shops available to be delivered in the location you entered at the start 3. Payment via google pay (secure payment) 4. The app has cart 5. See previous orders 6. Related products provided can help you to compare with the products you want to add in your basket	1. Necessary to register. 2. You cannot add to cart from two different grocery shops. 3. Shops not publicly known 4. No description 5. It says that you can sign up with fb and google but it doesn't allow you to do that.
Aeda (Web App, Android & iOS mobile app)	1. Choose between one shop. 2. My Account 3. FAQ 4. Ask a question, help 5. Previous orders 6. Search for a product 7. Categories and sub-categories 8. Title, photo, description 9. Add to favourites 10. Display reviews 11. Displays peoples' ratings with reviews 12. Recipe inspiration section 13. "You may also like section", before proceeding to payment 14. Preview order before payment	1. Displays peoples' ratings with reviews 2. Displays reviews with ratings 3. You can add your weekly purchased items as favourite products. You can easily select them for future orders. 4. Preview your order before payment can help you find out if you added anything or if you added anything by mistake 5. Recipe can give you an inspiration not only on how to cook a meal but also can help you if you miss a necessary ingredient 6. You can add a comment can help you see the products with the best customer reviews. 7. you can edit basket until two days before payment. This will help you to add any product in your thought is not necessary to buy but you reconsider. 8. Best product 9. User friendly website	1. First available store for delivery is far away
Sabaharys (Web App, Android & iOS mobile app)	1. Search for a product 2. Offers category (Prices dropped - long term offers) 3. Common categories section 4. Categories with subcategories section 5. Sub-sub categories are displayed after clicking on the category 6. Filtering 7. Read reviews / write reviews 8. Clicking checkout, "before you go section" 9. Forgotten your favourites 10. Book slot 11. Delivery pass 12. Recipes Section 13. Favourites 14. Past orders	1. Past orders 2. Favourites 3. Payment is made few hours before delivery	1. filter your list unnecessary 2. 40€ minimum order for delivery 3. Simpler Front end (more white space) 4. Can not pay with google/apple pay 5. Complex categories

C IDEAS

IDEAS

Web Application responsive

- Easier access by a link either from a computer or by a mobile phone. This covers most of the different ages and type of experiences with technology.

Consumers view the first time they visit the MySuperTrolley website

- Display countries, cities of available areas for delivery. This is done so that they will know their location is available for delivery.
- Display grocery shops that have products on the website (e.g Lidl, Morrisons, Tesco, Iceland, Sainsbury's, Aldi) in order for the consumers to be aware of the grocery shops that are in MySuperTrolley and not spending their time without any particular reason in the app.
- Users are able to enter their delivery location (postcode) before they start. This will happen for the users to see if the order can be delivered to their location.
- Even if the user does not find a valid postcode for delivery, the app can notify the user that MySuperTrolley currently can not deliver groceries to their location but the consumer can still be able to proceed and see the products of the website. This will happen in case the user wants to compare the prices of items from different grocery shops. Also, if the website allows users to proceed without notifying them, at the end, when they complete a basket and they find that the order can not be delivered to their location, it will frustrate the user and maybe will not buy again from this website.
- At the bottom of the page, the app will show some contact information, live chat and FAQ links.

Opportunities for grocery users

- Register / Login (opportunity to do that with Facebook and Google).
- User can edit account details.
- See previous orders. This will help users in future payments to recall some already purchased items or make the same order if they buy specific items per order. This will also help people with less experience with technology to make the process of ordering easier for them.
- Add items to favourites for future orders to make their selection of products easier. This will also help people with less experience with technology to make the process of ordering easier for them.
- Search for products. When the users search for products it will show all different products with a button saying how many shops sales this specific item. When the user clicks on the shops it will show all the different shops that sale this item.
- Offers category.
- Filtering. This will help users to see top sellers, ratings from high to low and the number of products displayed per page.
- Find products from categories, sub-categories and sub-sub-categories (e.g Fresh Food Fruit Bananas).
- Most common categories can have a different section to help users to visit those categories easily. Or there will be just a categories section and then filtering with different sub-categories.
- The selected item will show all shops that sale this product along with its price, offer and ratings.
- The selected item will show the title, photo, description. This will help users understand better the product they are adding to the basket and making sure that the product they select matches what they actually want to buy.
- Displays how many people rate a specific product from each shop along with the rating.
- Related products section for the product selected. This will happen for example, if a user selects a shampoo and related product will be a conditioner.
- Add items from different shops to a single basket. This is one of the most helpful features a website can have. No other website already has that feature. Most of the customers buy different type of food from different grocery shops. This will make their life of the users easier as they will be able to buy all those items from one website.
- The user can edit the basket until one (or some other time similar to that) days before the delivery slot. For example, if the next available delivery is in 5 days and 2 days before the user sees that they also need milk we can add it to his basket.
- Displays "You may also like section", before proceeding to payment.
- Users can preview order before payment. This can help users to find out if they missed anything or if they added any product by mistake.
- FAQ can help users to see if the inquiry they may have is already answered.
- Ask a question, via an online chat (Customer service). Customer service is very useful and mandatory to get the trust of the user.
- Displays the recipe inspiration section. This will make customers select some products that may be forgotten.

IDEAS

Opportunities for manager/admin user

Admin will have access to the portal where they will be able to:

- Enter manually some products.
- Remove items if there is a glitch.

Delivery

- The user can see and book available slots with day, time and cost of delivery.
- Delivery pass. With some monthly fee, the users can have some extra opportunities. For example, free delivery, book better slots.
- The minimum amount of order. Let's say 15€ per shop and overall 40€.
- Above a certain value of the order, there will be free delivery. Let's say 60€.

Payment

- Add credit card and pay, or via google/apple pay.
- The basket would be categorised with all shops that have at least one products selected. For example, if the user selects products from Lidl, Tesco and Iceland, it will show three different prices, one from each shop, but the payment will be in a single basket. The user will just need to click pay three different times but at the same time.
- Payment will be made actually one (or some other time similar to that) days before delivery. Notify the user that the payment is made. This will let them know that they are no longer able to add items to the basket.

D User Personas

User Personas

MySuperTrolley

Emily - 22 years old



Emily is an undergraduate student at the university of Glasgow from Slovakia. She is currently living in Glasgow. According to Covid situation, she is afraid going to the grocery shop to take her weekly grocery shopping as she was usually did in the past years because she is uncomfortable in being near other people. She also shops most of her weekly items from lidl and the more perishable products such as fresh fruit and vegetables from morrisons. She need an app where can put all those items from the two grocery shops in one basket and delivered to her flat.

Kyle - 43 years old



Kyle is an employee in a big firm in London. He lives with his wife and two children in a flat in central London. He was using the underground to go to work. He was buying the groceries for the whole family mainly from tesco and some specific items from iceland and lidl for his children near his workplace. However he is currently working from home and he is not able to find iceland near his flat. Since he buys lots of products to satisfy the needs of the whole family and maybe the next available delivery is in five days, he can also have the opportunity to add items to the basket up to at least two days before delivery as milk may be left out from the fridge. He needs a website where he can order all weekly items from tesco, iceland and lidl and delivered to his flat.

Lucy & John Brown - 70 years old



John and Lucy are married couple enjoying their pension living in a house outside of London. Every Saturday they were driving to different grocery shops to satisfy their weekly grocery needs. Due to the Covid situation the government informed the elderly age people to stay at home and reduce most of their transportsations. They are also not very experienced with the technology and their son helped them to buy from tesco online their groceries. However only tesco does not satisfy their needs because John wants to buy specific things for his plants which grows in the garden. He was buying those things from morrisons. So, he wants to shop items for other websites as well. They would like a website where they can buy items from different shops in one basket. Also, since their are not very good with technology and their weekly items are almost the same, they would like to see their previous order and order the same things every time. In addition, they are not good at finding items through categories and they would like to add all the items they like on favourites. This will help them find these items in future orders easily.

Ryan - 25 years old



Ryan is an employee in a law firm in edinburgh. In the past two years two times someone tries to take money from his card and they hack his account in an online shopping website. He tried to find out what to do but the website didn't have any customer support. Due to these disappointments he is afraid of registering with email address and passwords in websites and generally shopping online. But he is working late almost every day and when he goes to the supermarkets almost all products are gone. He would like an app where he can login with google, pay with apple/google pay so that the payment would be secure. He also needs to be able to chat with a customer service so that they will help him in strange situations.

Anna - 35 years old



Anna is a doctor working in a hospital in Glasgow but she lives in a small village 45 minutes away from Glasgow. She is working many hours per day with different slots. She come home very tired after work. Due to Covid situation she is afraid going to the supermarkets to buy her groceries because she may infect other people there. She needs an app where she can order groceries buy booking available slots that she will be at home. She also wants to be able to make sure before to start adding items to her basket to make sure that the order can be delivered to her location.

E User Stories

User Stories

Structure

```
As a [description of user],
I want [functionality],
so that [benefit].
```

Grocery User

What user sees first when visits MySuperTrolley

- As a grocery user,
I want to view cities of available areas of delivery,
so that I will make sure that the order can be delivered to my location.
- As a grocery user,
I want to be able to add my postcode,
so that I will see if the order can be delivered to my location.
- As a grocery user,
I want to be able to receive notification for the postcode that I entered,
so that I will see if the order can be delivered to my location.
- As a grocery user, I want to view the different shops that sale items on the website,
so that I will know from the start if I want to continue or not.
- As a grocery user,
I want to view contact information and FAQ links,
so that I can find useful information if I have some issues with my order.
- As a grocery user,
I want to be able to have a live chat,
so that someone can help me if I have any issues with my order.

Options available for grocery user

- As a grocery user,
I want to be able to register/login,
so that I can have extra opportunities-functionalities.
- As a grocery user,
I want to be able to edit my personal details,
so that I can be up to date with my details.
- As a grocery user,
I want to be able to login with Facebook or Google,
so that I can log in easier, faster and more secure.
- As a grocery user,
I want to have access to my previous orders,
so that I can make easier my future orders.
- As a grocery user,
I want to view the item favourites page,
so that the selection of items in future order will be easier.
- As a grocery user,
I want to add/remove items to favourites,
so that the selection of items in future order will be easier.
- As a grocery user,
I want to search for different items
so that I can find the items I want to add to my basket.
- As a grocery user,
I want to be able to see all grocery shops that sale the item I selected with price, offer if any,
so that I compare the different items before making decisions.

- As a grocery user,
I want to view the offers category
so that I will see if any of the items satisfy my needs to add them to my basket.
- As a grocery user,
I want to be able to filter products by ratings from high to low and the number of products per page,
so that I make the list of items more precise of what I want to buy.
- As a grocery user,
I want to find items from specific categories,
so that I will make a faster selection of an item.
- As a grocery user,
I want to filter categories into subcategories,
so that will make faster my selection of an item.
- As a grocery user,
I want to view the title, description and photo of a product selected,
so that I will fully understand an item.
- As a grocery user,
I want to view ratings along with how many people rate this item,
so that I can decide whether I should or should not buy that item.
- As a grocery user,
I want to view related items of an item I selected,
so that it can help me buy other items that I forgot, related to this item.
- As a grocery user,
I want to add items to the basket from different grocery shops,
so that I can satisfy my grocery needs from one website.
- As a grocery user,
I want to edit my basket until two days before the delivery,
so that I can add some perishable items that may be run out last minute.
- As a grocery user,
I want to get a notification when the payment was made,
so that I know that am not able to edit my basket anymore.
- As a grocery user,
I want to be able to see a preview of the order before payment,
so that I make sure all the items I selected are in the basket.
- As a grocery user,
I want to be able to see the most common recipe section,
so that can help me view items that I may forget.
- As a grocery user,
I want to view the available slots along with the delivery cost,
so that I can select the most convenient slot for me.
- As a grocery user,
I want to have a delivery pass membership,
so that I get some extra contingency for my future delivery orders.
- As a grocery user,
I want to link my credit/debit card to my profile,
so that I can pay for an order faster, easier and without cash.
- As a grocery user,
I want to be able to pay via google/apple pay,
so that my payment would be more secure.
- As a grocery user,
I want my basket to separate the items based on the selected shop,
so that I know how much money I spend in each shop.
- As a grocery user,
I want to select same-day, two-day delivery, etc.
so that I know from which shops items are available for delivery based on my selection.

Admin/Manager User

1. As a manager,
I want to be able to edit manually some items,
so that I can prevent from buying some items in case of a glitch.
2. As a manager,
I want to be able to delete some reviews of a product,
so that the user will not see offensive review.

F Survey - Requirements

<https://docs.google.com/forms/d/1MfRMnfQLW-oZouFIs7VnPsKSUrstjTzvgisX6cHiKVA/edit?usp=sharing>

G Agenda

23/02 Tuesday



Created by Nikolas Finiotis
Last updated Feb 23, 2021 • 3 min read

Time: 9:30am-10:00am (30 minutes)

What I have done the previous week (20 min)

Implementation

- Started implementing the booking of slot functionality in the valid postcodes page and in the order preview page. I encountered a problem with the timestamp type of saving the delivery date slot into the database. Solved the problem of date. Changed from timestamp into a date type and now the delivery slot is saved into the database.
- Displayed separate the scheduled and previous orders on the past orders page. Display nicely the delivery date.
- Created a checkout page. Displacement of the two sections(billing, payment) in the checkout page along with the total price and a button to place the order.

Unit testing

- Started trying to create tests using Mock MVC trying to find a way to do that by searching and reading some tutorials. Took me so many hours to remove errors in calling mock mvc. I didn't used it before and json path with Result Matcher required some time in order to figure out to solve the problem. **But Finally made it work properly!**
- Finally managed to mock all **get requests** in all the controllers namely, AddressControllerTest.class, BrandControllerTest.class, CategoryControllerTest.class, CityControllerTest.class, CountryControllerTest.class, ItemsControllerTest.class, SubcategoryControllerTest.class.
- Created suite classes and run all the tests in one class called **MySuperTrolleyApplicationTests**.
 - Code Coverage:
 - Class %:** 87% (47/54)
 - Method %:** 43% (113/257)
 - Line %:** 33% (209/620)
- Started to try mocking post requests for the post requests methods in order controller but I encountered some errors. Hopefully, by the next meeting, I will have some good progress on that.

Dissertation

- Updated and fixed some parts from the introduction section in the dissertation, I sent to you via email the updated version of the **Introduction**.
- Wrote the first draft of the **background research** section (9 pages using a lot of content for 6 subchapters, need to refactor, narrow down into a cohesive final draft, approximately 5 pages).

What I am planning to do until the next meeting (5 min)

- Made Sprint 5 Retrospective.
- Add issues for sprint 6 in Jira.
- Implement the **Favourites** page.
- Finalise **Checkout** page.
- When the checkout page is finished and the delivery address is saved in the database, all the details in the past orders page will be automatically retrieved. So will finish the **past orders** page.
- Continue writing unit tests (post requests mocking).
- Finalise **background research** section.
- Start writing the first drafts for the comparison of **already existing similar applications** (features, strengths, weaknesses). **Requirements Gathering** and the **Development Methodology** of the dissertation.
- Continue to create unit tests for post request.
- Research how to approach **Performance Testing**.

Any questions I would like to ask (5 min)

- What comments can you give me about the introduction section?
- As you saw the implementation I made about the checkout page, what extra implementation do you encourage me to do with that. I mean since the items mimic the functionality, is there any specific reason to implement the payment with Paypal or via Google pay, if the payment will not actually be made? Or just when the payment details (fake) are entered and the pay button is made there will be a response that the payment was made?
- Are five pages enough or too much for just the background research chapter?

H ER-diagram

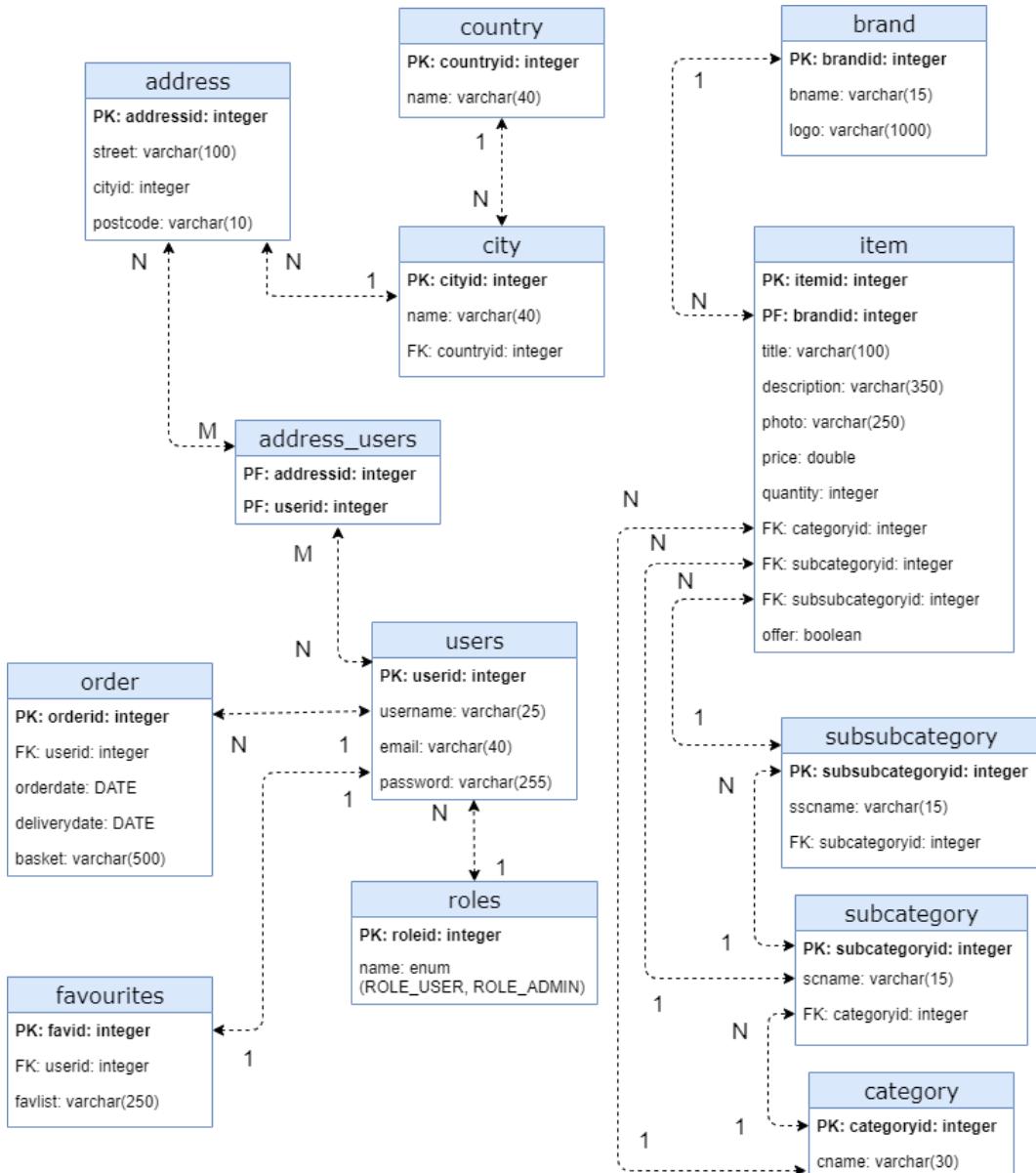


Figure 1: Entity Relationship Diagram

I Activity-Diagram

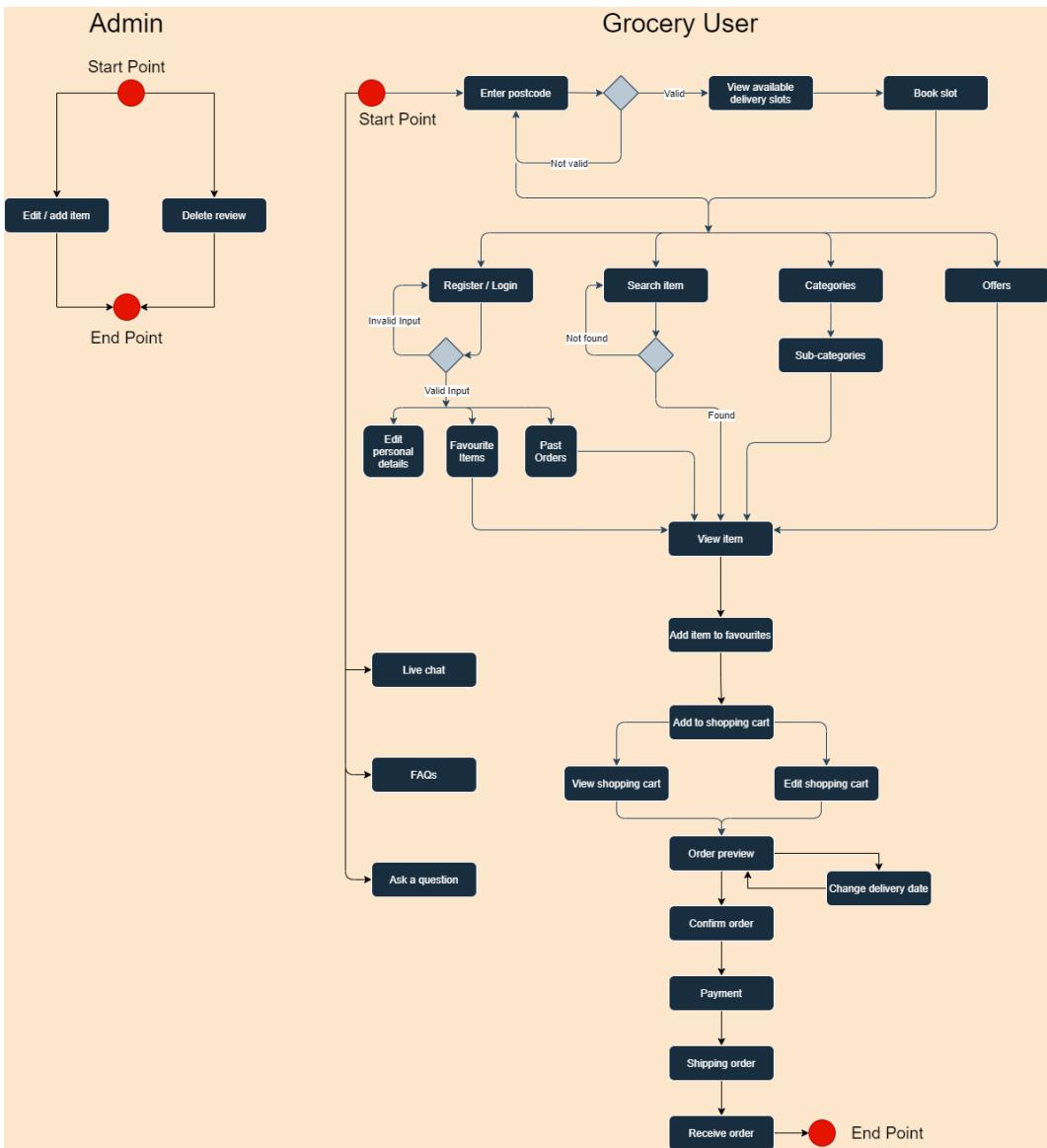


Figure 2: Activity Diagram

J Paper-Prototypes

Enter location, available slots and brands

The main thing behind this page is that the grocery user is good to know if the items can be delivered to their location. So, the user can see which cities are available for delivery and enter their postcode to see the availability and book slot based on the needs of the user. For example, by taking the postcode of the user we can display the shops that are near the user's location for next day delivery, some other shops as well for two days delivery and so on.

Image A

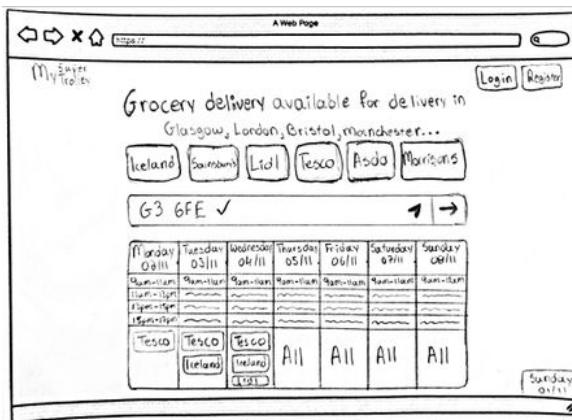
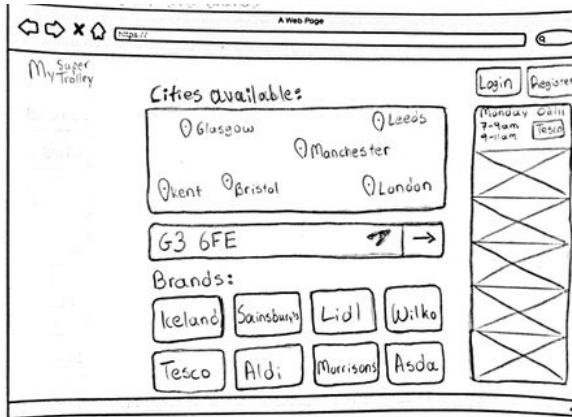


Image B (Selected)



Login/Register

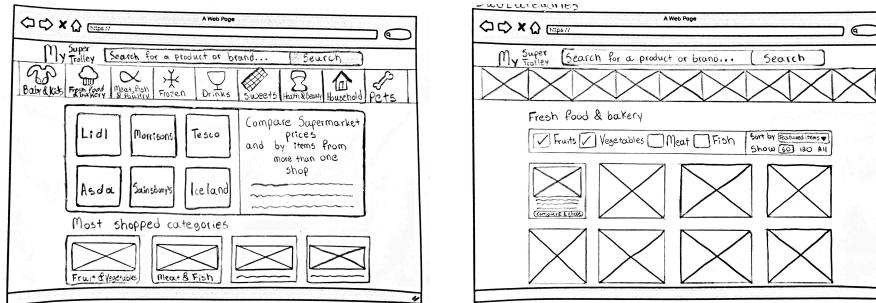
The main thing behind this page is that a user that has an account has some benefits. For example, the user can have access their favourite items and past orders. Users with limited skills in technology can use this opportunity to save into their favourites section, the main items that they shop to make their selection of items easier. Also, some users that shop almost the same items per order can just click on the past orders section and make the same order again with just one click.

The image contains two separate wireframe sketches of user interface components, likely from a design process:

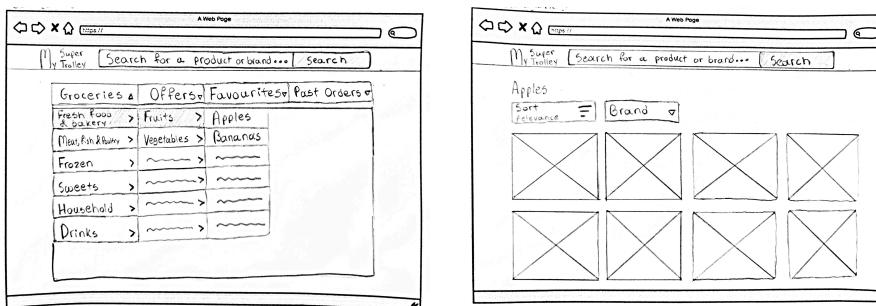
- Login:** A window titled "Login" showing a "Log In" form. It includes fields for "Email address" and "Password", a "Forgot Password?" link, and a "Log In" button. Below the form are links for "Forgot my password" and "Log in with Facebook".
- Registration:** A window titled "Registration" showing a "Create Account" form. It includes fields for "Name" (split into "First name" and "Surname"), "Your Email", "Password", and "Repeat your password". There is a checkbox for "I agree all statements in Terms of Service" and a "SIGN UP" button. At the bottom, there is a link for "Have an account? Log in here".

Categories & Sub-categories

Images A (Selected)



Images B



Item details

The main thing behind this page is that the user must have a great understanding of what they buy. So, in order to achieve that, a title, photo and description must be shown for each item. Also, the page will show all the different shops that sell the specific item along with the price and rating. This is done so that the user can select to buy the item for the most appropriate shop based on their interests.

Image A (Selected)

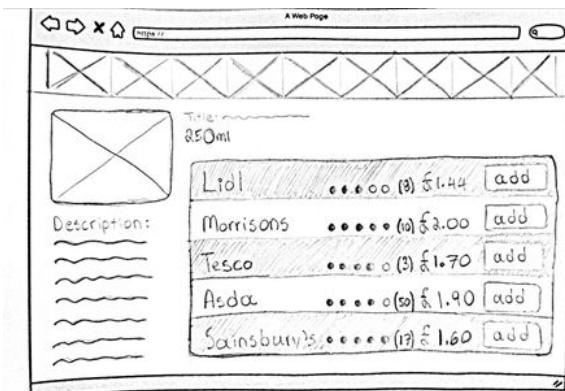
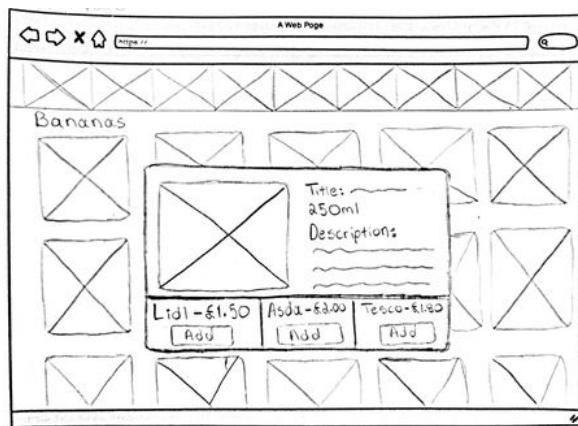


Image B



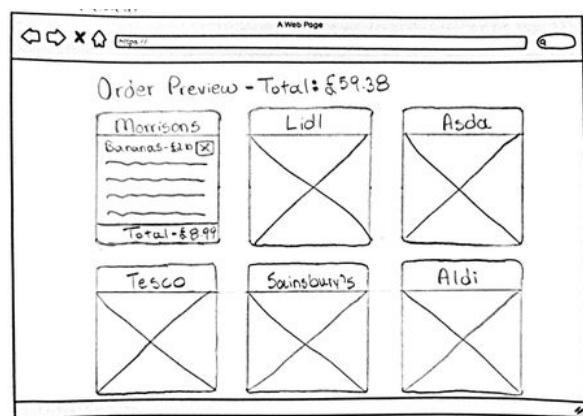
Order Preview

The user before proceeds to checkout must have the option to see again the selected items.

Image A (Selected)

Order Preview			
Aldi	X X		Total £15.23
Asda	X X X X		Total £20.00
Tesco	X		Total 3.55
Lidl	X X		Total £13.65
Morrisons	X X X		Total £8.99
Total - £59.38			

Image B



Past Orders

Image A

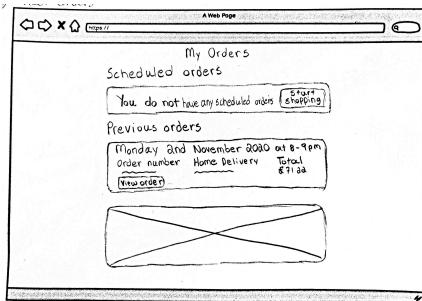


Image C

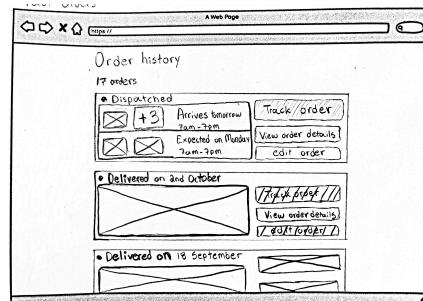
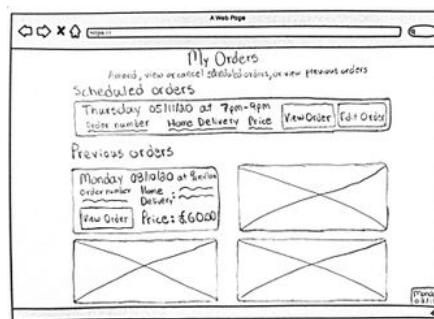


Image B (Selected)



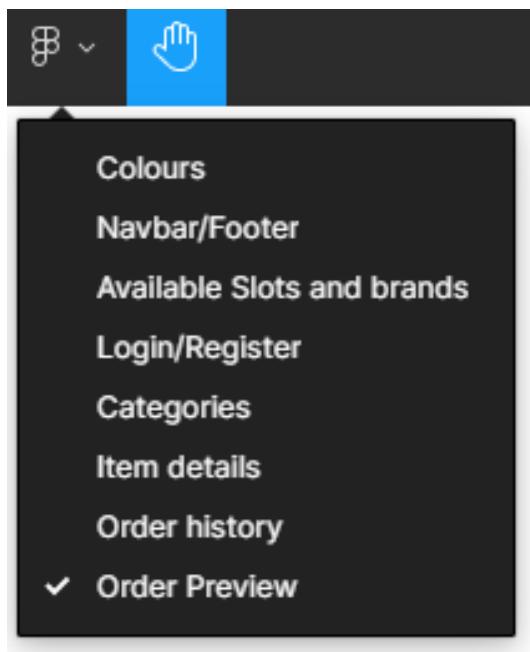
K A-B Testing

https://docs.google.com/forms/d/1qoJvM-EGbwVYaN7lZrzMZ_Uob1Z8Z1pQazhBaAaD5U/edit?usp=sharing

L Wireframes

<https://www.figma.com/file/Joes1PpcAyRKeXInoZqjm6/MySuperTrolley?node-id=77%3A0>

From the top left corner you can select and view the different wireframes as shown in picture below:



M Unit Testing

Unit Testing		
Test Cases	Passed	Not Passed
BrandController		
Retrieve all brand shops	✓	
Retrieve all brand shops - Wrong	✓	
CategoryController		
Retrieve all categories	✓	
Retrieve all categories - Wrong	✓	
CityController		
Retrieve all cities size	✓	
Retrieve all cities size - Wrong	✓	
Retrieve all city names	✓	
Retrieve all city names - Wrong	✓	
CountryController		
Retrieve all countries size and name	✓	
Retrieve all countries size and name - Wrong	✓	
ItemsController		
Retrieve all items size	✓	
Retrieve all items size - Wrong	✓	
Retrieve all distinct items size	✓	
Retrieve all distinct items size - Wrong	✓	
Retrieve item using the item id	✓	
Retrieve item using the item id - Wrong	✓	
Retrieve item using the brand id	✓	
Retrieve item using the brand id - Wrong	✓	
Retrieve item using the item id and brand id	✓	
Retrieve item using the item id and brand id - Wrong	✓	
Retrieve distinct items by category id	✓	
Retrieve distinct items by category id - Wrong	✓	
Retrieve items that are in offer	✓	
Retrieve items that are in offer - Wrong	✓	
Retrieve distinct items by subcategory id	✓	
Retrieve distinct items by subcategory id - Wrong	✓	
Retrieve number of shops sell each item	✓	
Retrieve number of shops sell each item - Wrong	✓	
SubCategoryController		
Retrieve subcategory details (sub id, name, catid) given the category id	✓	
Retrieve subcategory details (sub id, name, catid) given the category id - Wrong	✓	
Retrieve subcategory ids given the category id	✓	
Retrieve subcategory ids given the category id - Wrong	✓	

Table 1: Test Cases

N User Evaluation - Survey

https://docs.google.com/forms/d/13tsJoKT8KfHwLwZZomstAEwQG8-7T2g0-q-91SYEh_8/edit#responses

O SUS Score

P Requirements Validation

Requirements Validation - Must Have		
Requirements	Done	Didn't Do
Show countries/cities available for delivery.	✓	
Show grocery shops that have items in the website.	✓	
Users can enter a postcode to see if their location is available for delivery.	✓	
Users can have access to previous orders.	✓	
Users can add/remove items from favourites.	✓	
When clicking on a product, the page will show all shops that sell a specific item along with price and offer (if any).	✓	
Users can find items from categories sections.	✓	
Show details of an item (title, description, photo).	✓	
Users can use a filter for subcategories.	✓	
Users can add items from different grocery shops in a single basket.	✓	
Users can see the preview of the order before confirming the order.	✓	
Users can view available slots for delivery (range of time, day, cost).	✓	
Users can book a slot for delivery.	✓	
Users can register/login.	✓	
Users can view items they added to their basket separated by grocery shop along with their total price.	✓	
Users can make an order with items from multiple shops in a single basket.	✓	

Table 2: Requirements Validation - Must Have

Requirements Validation - Should Have		
Requirements	Done	Didn't Do
Users can access offers section.	✓	
Users receive notification for the availability of postcode.	✓	
Users receive notification for confirmation of the order.	✓	
Users can add multiple number of the same product in the basket (using item quantity).	✓	
Users can ask a question if they have any inquiries.		✓
Live chat for customer service and support.		✓
Users can see FAQ with answers.		✓
Users can select next-day, or another selected day for their delivery, in order to see which shops can deliver items to their location based on their selection.		✓
Users can see the related products section for a selected item.		✓
Users can use a filter for the number of products per page, high to low price or vice versa and previous customer reviews.		✓
Users can update their personal details.		✓
Users can search for items.		✓
Users can edit their basket (users can also do that after their order is confirmed up to a specific time period before their order is shipped and paid for).		✓
Manager can manually add/remove some items (e.g in case of a glitch).		✓

Table 3: Requirements Validation - Should Have

Bibliography

- Agilebusiness.org (2019), ‘Chapter 10: Moscow prioritisation’, https://www.agilebusiness.org/page/ProjectFramework_10_MoSCoWPrioritisation. Last accessed: 2021-02-28.
- Ahuja, M., Gupta, B. and Raman, P. (2003), ‘An empirical investigation of online consumer purchasing behavior. communications of the acm’.
- Aldridge, A., Forcht, K. and Pierson, J. (1997), “get linked or get lost: Marketing strategy for the internet.”.
- Ansgar Becker (2015), ‘Heidisql – mariadb, mysql, mssql, postgresql and sqlite made easy’, <https://www.heidisql.com/>. Last accessed: 2021-03-11.
- Atlassian (2021), ‘Bitbucket: What is bitbucket? – atlassian documentation’, <https://confluence.atlassian.com/confeval/development-tools-evaluator-resources/bitbucket-bitbucket-what-is-bitbucket>. Last accessed: 2021-03-10.
- Bourlakis, M., Papagiannidis, S. and Fox, H. (2008), ‘E-consumer behaviour. international journal of e-business research’.
- Buymie (2021), ‘Buymie’, <https://buymie.eu>. Last accessed: 2021-02-26.
- Chan, M (2021), ‘Sql vs. nosql – what’s the best option for your database needs? thorn technologies’, <https://www.thorntech.com/sql-vs-nosql/#:~:text=SQL%20databases%20provide%20great%20benefits>. Last accessed: 2021-03-11.
- Cockburn, C. and Wilson, T. (1996), ‘Business use of the world-wide web. international journal of information management’, <https://www.sciencedirect.com/science/article/pii/0268401295000712>. Last accessed: 2021-02-22.
- Comegys, C., Hannula, M. and Väistönen, J. (2009), ‘Effects of consumer trust and risk on online purchase decision-making: A comparison of finnish and united states students, international journal of management’.
- Cuelogic Technologies Pvt. Ltd. (2020), ‘Angular vs. react: A complete comparison guide.’, <https://www.cuelogic.com/blog/what-are-the-differences-between-angular-and-react>. Last accessed: 2021-04-08.
- Deloitte Insights (2021), ‘Changing the playbook’, <https://www2.deloitte.com/us/en/insights/topics/marketing-and-sales-operations/global-marketing-trends/2021/agile-marketing-strategy-digital-world.html>. Last accessed: 2021-03-29.
- Devo (2021), ‘Devo’, <https://devo.co.uk>. Last accessed: 2021-02-26.
- devportal.tescolabs.com (2021), ‘Tesco labs developer portal.’, <https://devportal.tescolabs.com/>. Last accessed: 2021-04-06.

- Elliott, C (2021), ‘This is how the pandemic improved customer service’, <https://www.forbes.com/sites/christopherelliott/2021/01/28/this-is-how-the-pandemic-improved-customer-service/?sh=7f38b15848dd>. Last accessed: 2021-03-29.
- Experience UX (2019), ‘What is wireframing’, <https://www.experienceux.co.uk/faqs/what-is-wireframing/#:~:text=A%20wireframe%20is%20a%20layout>. Last accessed: 2021-03-11.
- Gao, X., Shi, X., Guo, H. and Liu, Y. (2020), ‘To buy or not buy food online: The impact of the covid-19 epidemic on the adoption of e-commerce in china.’.
- GeeksforGeeks (2017), ‘Unified modeling language (uml) | activity diagrams’, <https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/#:~:text=An%20activity%20diagram%20is%20a>. Last accessed: 2021-03-11.
- JavaSterling (2020), ‘Spring boot vs django - difference between spring boot and django’, <https://javasterling.com/spring-boot/spring-boot-vs-django/>. Last accessed: 2021-03-11.
- Katawetawaraks, C. and Wang, C. (2011), ‘Online shopper behavior: Influences of online shopping decision. asian journal of business research’.
- Korgaonkar, P. and Karson, E. (2007), ‘The influence of perceived product risk on consumers’.
- Laguna, L., Fiszman, S., Puerta, P., Chaya, C. and Tárrega, A. (2020), ‘The impact of covid-19 lockdown on food priorities. results from a preliminary study using social media and an online survey with spanish consumers. food quality and preference’.
- Li, J., Hallsworth, A. and Coca-Stefaniak, J. (2020), ‘Changing grocery shopping behaviours among chinese consumers at the outset of the covid-19 outbreak’.
- Liang, T.-P. and Huang, J.-S. (1998), ‘An empirical study on consumer acceptance of products in electronic markets: a transaction cost model. decision support systems’.
- Lim, Y., Osman, A., Salahuddin, S., Romle, A. and Abdullah, S. (2016), ‘Factors influencing online shopping behavior: The mediating role of purchase intention. procedia economics and finance’, <https://www.sciencedirect.com/science/article/pii/S2212567116000502>. Last accessed: 2021-02-22.
- Limayem, M., Khalifa, M. and Frini, A. (2000), ‘What makes consumers buy from internet? a longitudinal study of online shopping. ieee transactions on systems, man, and cybernetics – part a: Systems and humans’.
- Liu, C. and Guo, Y. (2008), ‘Validating the end-user computing satisfaction instrument for online shopping systems, journal of organizational and end user computing’.
- Lucidchart (2017a), ‘Agile methodology and how you’re already using it’, <https://www.lucidchart.com/blog/what-is-agile-methodology>. Last accessed: 2021-03-10.
- Lucidchart (2017b), ‘Scrum for one: Adapting agile scrum methodology for individuals’, <https://www.lucidchart.com/blog/scrum-for-one>. Last accessed: 2021-03-10.
- Office for National Statistics (2019), ‘Internet sales as a percentage of total retail sales (ratio) (%) – office for national statistics’, <https://www.ons.gov.uk/businessindustryandtrade/retailindustry/timeseries/j4mc/drsi>. Last accessed: 2021-02-08.
- Prashar, K. and Mohali (2020), ‘Study of consumer perception towards online grocery shopping’.

- The Interaction Design Foundation (2021), ‘What is paper prototyping?’, <https://www.interaction-design.org/literature/topics/paper-prototyping>. Last accessed: 2021-04-03.
- Thomas, N. (2019), ‘How to use the system usability scale (sus) to evaluate the usability of your website – usability geek.’, <https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/>. Last accessed: 2021-04-05.
- Trolley (2021), ‘Trolley’, <https://www.trolley.co.uk>. Last accessed: 2021-02-25.
- Visual Paradigm (2019), ‘What is entity relationship diagram (erd)?’, <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>. Last accessed: 2021-03-11.
- Wu, W.-Y., Ke, C.-C. and Nguyen, P.-T. (2018), ‘Online shopping behavior in electronic commerce: An integrative model from utilitarian and hedonic perspectives’.