



COMPUTING NEA

Product Library and Online Shop System

BY DANIEL OGUNRIN

Contents

| | |
|--|-----|
| ANALYSIS..... | 3 |
| 3.1.1 IDENTIFYING THE PROBLEM | 3 |
| COMPUTATIONAL THINKING | 3 |
| 3.1.2 STAKEHOLDERS..... | 5 |
| End-user Information..... | 6 |
| 3.1.3 RESEARCH THE PROBLEM | 6 |
| QUESTIONNAIRE | 6 |
| Questionnaire – The Responses | 8 |
| Current Systems..... | 13 |
| IPOS..... | 15 |
| INTERVIEW | 19 |
| Proposed Solution..... | 20 |
| User/Client Requirements | 21 |
| Design requirements (Hardware and software)..... | 22 |
| Functional Requirements..... | 23 |
| Non-Functional Requirements..... | 25 |
| Limitations | 26 |
| Success Criteria | 27 |
| DESIGN | 29 |
| 2.1 – Specification..... | 29 |
| 2.1.1 – Inputs | 30 |
| 2.1.2 – Processes..... | 30 |
| 2.1.3 – Storage | 30 |
| 2.1.4 – Outputs | 30 |
| 2.2 – Decomposition | 34 |
| 2.2.1 – Database | 34 |
| 2.2.4 | 43 |
| – System Design | 43 |
| Creating the assets..... | 74 |
| 4 - Final Testing | 159 |
| 4.1 - Test Plan..... | 178 |
| 4.2 - Usability Testing..... | 181 |
| 5 – Evaluation | 190 |
| 5.1 –Success Criteria | 190 |
| 5.1.1 – Evaluation of Success Criteria..... | 190 |

| | |
|--|-----|
| 5.1.2 – Unmet and Partially Met Criteria..... | 192 |
| 5.2 –Validation | 199 |
| 5.2.1 – Evaluation of validation | 199 |
| 5.2.2 – Unmet or Partially Met Validation..... | 200 |
| 5.5 – Maintenance | 202 |
| 5.6 – Limitations | 203 |
| 5.7 – Improvements..... | 204 |
| 5.8 – Conclusion..... | 205 |
| 5.9 – Client Sign-off..... | 206 |

ANALYSIS

3.1.1 IDENTIFYING THE PROBLEM

DESCRIPTION OF THE PROBLEM

Apps have become integrated in our everyday lives. We use them to wake up, read the news in the morning, follow recipes, and to connect with friends. When we are bored, we play games and read books – all on our favourite apps. Another reason people use retail apps are to buy the latest trending sneakers, latest street fashion and latest accessories to look trendy and cool. However, there lies a big problem that normal retail apps do not solve - the lack of information about release/drop dates for the most popular sneakers and streetwear items and where to buy them from for the best price. This is the problem my app will try to solve.

Many people are upset when they miss out on the latest fashion items that they want because they do not have suitable and efficient apps to buy them on or they just do not know when and where to get it from. I have personally dealt with the frustration of trying to buy a latest item and it selling out straight away or just missing out on the item because the date of release was not made clear to me. The solution to this problem is an efficient fashion website that only advertises the most popular items and also shares information and notifies its users on release dates and restocks.

This problem is solvable using a computational approach because it can be broken down into a lot of smaller subtasks and allows focus on important parts of the task. It also limits the time people take searching for retail websites and information to buy their items. In order to solve this problem, I will conduct research into what makes a successful app and look at existing systems and then develop the app.

My system will also differ from existing systems because it will give normal buyers a prediction of the price of how much the resell price of an item would be if they can't get it for the retail price. It will also give them a price tracker if they think the resell price is too high and it will notify the user when the price of the shoe/item is going down. For resellers, it will give them a profit prediction on the shoes and items. With providing vital information and features for these two main groups of people, it enables everyone to be able to use the app for their benefit.

COMPUTATIONAL THINKING

To achieve my product, I will need to utilise various aspects of computational thinking. Computational thinking is the step that comes before programming. It is defined as the process of breaking down a problem into simple enough steps in a way that a computer could even understand. We have already identified the problem - the next step would be to split the problem into a different number of stages. This is vital when developing computer systems but is also useful when attempting to understand complex problems with many parts and allows for viable solutions to be developed. Using computational thinking, I will be able to use skills developed from other subjects (for example using skills learned from Geography to observe patterns and trends in data), allowing me to create and improve the system more efficiently.

Computational thinking is not to be confused with programming; however, one is dependent on the other. Computational thinking is an essential step that comes before programming and defines what is going to be programmed. So, before I start to program my project, I will use computational thinking to produce a strategy on how to approach different areas of my app.

There are four key areas of computational thinking: decomposition, pattern recognition, abstraction, algorithm design

- **Decomposition:** Breaking a complex problem into smaller and more manageable parts.
- **Pattern recognition:** observing patterns, trends, and similarities in data
- **Abstraction:** Only focusing on information that is important to your tasks
- **Algorithm design:** developing a step-by-step solution to the current problem

Each one of these techniques is particularly important for computational thinking and therefore for my project.

- **DECOMPOSITION:** Decomposition is a problem-solving technique that is essential to computer science. The aim of decomposition is to reduce the complexity of a problem by breaking it down into a series of smaller, simpler problems that can be completed one at a time. When the solutions to all the smaller problems are put together, a solution for the larger problem is achieved. This will be useful for my project as I will be required to take on and design different parts of my shoe and fashion app and put them together to finally achieve the final product. Some examples of the subtasks I would have to handle include:
 - Create a system where people can log in and create an account
 - Store all accounts in a database
 - Designing of the user interface
 - Notification system
 - A system that identifies the most popular fashion items (this could be determined by a rating system) and only advertises those items strictly, also need a place for advertisers to advertise their items.
 - A place to write to and store the links and information about each popular fashion item
- **PATTERN RECOGNITION:** Pattern recognition is the identification of similarities, characteristics and regularities that some of the decomposed problems share. By recognising patterns between the decomposed problems, it helps me to solve them more efficiently. Instead of me having to repeat the same process for problems that are alike I can solve them all at once, saving much more time. We can use the same problem-solving solutions whenever similar patterns exist and the more patterns we find, the easier and quicker the overall task of problem solving will be. This is why pattern recognition is an essential cornerstone of Computer Science. In my project I expect to use pattern recognition when coding the functions for both the log in page and the register page and how they will be saved to the database as I expect their code to be similar. I will also utilise pattern recognition when implemented a top nav bar on every page.

- **ABSTRACTION:** Abstraction is the process of filtering out – ignoring - the characteristics of patterns that we don't need in order to concentrate on those that we do. By using abstraction, I can simplify my tasks by filtering out unnecessary information, removing specific detail and any patterns that will not help us solve our problem in order to focus on key aspects of the problem. Abstraction allows us to create a general idea of what the problem is and how to solve it. I expect to use abstraction in my code when creating functions that are of priority and leaving lower priority functions till last if I have time.
- **ALGORITHM DESIGN:** The last stage in computational thinking is algorithmic design. An algorithm is a plan, a logical step-by-step process for solving a problem. Before an algorithm is designed it is essential to check that the problem is completely understood. These instructions can be developed into the actual programming of the project. As this is the final stage of developing the system, I will need to follow the previous areas first before moving on to the actual programming. This allows the system to be made in the most efficient way possible. For example in my designs I will create algorithm designs using pseudocode and flowcharts to describe processes that will happen in my code such as creating an account a favouriting an item.

3.1.2 STAKEHOLDERS

A stakeholder is defined as an individual or group that has an interest in any decision or activity of a business/organisation/company. The internal stakeholders are usually the company's owners and employees. The external stakeholders are typically the company's customers, investors, media and suppliers. In my project we are going to mainly focusing on the external stakeholders.

The clients and demographic for this system would be sneaker collectors, resellers, teenagers, and clients who are into fashion. There is a big target audience for my app and the reasoning for them to use my system ranges from just buying fashion items for personal gain, to buying the items to resell for a larger profit. For my system to be created I need to have input from several sources so that my app can be steered in the right direction

The majority of people buy streetwear fashion and sneakers purely for personal gain, to look nice and wear the clothes that they love. This group of people will be a big stakeholder for my app as there are a lot of people that fit into this category. The problem that these people face is that a lot of the clothes and shoes that they love are either quickly sold-out because they don't know where to get them from and fast, and also because they don't know when it's coming out. A lot of the time resellers have studied the information of the release dates of these items and will have an advantage over these normal fashion buyers. My app's solution to this problem will give both resellers and regular people a chance to access the drops at the same time and have an equal chance at coping their desired item. The solution would be informing the user of the release dates and times of their desired item, and also the best websites to buy them on through links. The user will be reminded through the week so they're ready for the release of the item and when they item releases, the user will be notified instantly and will have the chance to access the link as fast as they tap it. This will give them access at the same time as every other person and give everyone an equal chance of getting their fashion item. This makes the solution appropriate to their needs. The stakeholders for this demographic are Ibrahim Benton and Lily Houghton. Ibrahim Benton is an 18-year-old teenager in sixth form who loves fashion and always posts his best outfits on his Instagram

for his thousands of followers. Lily is a 23-year-old influencer on Instagram with over 100,000 followers – she loves streetwear and displays many different outfits that contain clothing items that are sought after for her follower. Both individuals could make use of my app and useful for their pictures.

Another group of people that my app will be suitable for are resellers. These are a group that use the sneaker and fashion market into a job and sell shoes at a bigger profit than what they bought it for. This is another group that will especially be a big stakeholder for my system. As my app will only display the most popular clothes and sneakers, these are also the items that will give the biggest return in terms of profit. A lot of resellers are already in cool groups that already provide them with information on upcoming releases – however this usually comes at a monthly subscription cost ranging from £10 -£40 a month just for information and advice on sneakers and fashion. Not a lot of people would be willing to pay this, especially with other bills that they already be dealing with. My app will solve this problem as this information will be provided for free and enable people who can't afford to pay monthly subscriptions for a reselling group, without any actual guarantee that you will get the shoe in the first place from these paid groups. The resellers in paid groups can also utilise this app as it can provide information that their group might not (not all groups provide the same quality and amount of info/advice, hence the price range). The stakeholders for this reselling demographic are Richard Chinedu and Jonathan Bowman. Richard Chinedu is a 17-year-old student who resells sneakers as a side hustle and only started reselling this year. Jonathan Bowman is 30 years of age and collects all of the most cool and retro sneakers for his collection. Both of these people would make good use of my app.

End-user Information

My targeted client for this game will be a sixth form student who resells sneakers as a side hustle. I have chosen Richard Chinedu as my end-user, as I believe she has the ability to be critical to certain notification apps, and thus will provide me with good advice going forward. As mentioned above, he is not that experienced in the area as he has only started this year but he already gained a lot of knowledge of the reselling game and thus has an insight as to how to develop ideas and take them forward and make them more enjoyable for my target market, which he is a part of.

3.1.3 RESEARCH THE PROBLEM

QUESTIONNAIRE

To get more clues about the features I want to implement and to gain a better understanding of who exactly my target market is, I've conducted this questionnaire survey between a diverse age group. This will help me know who exactly would be interested in my project and it will also give me product specific requirements which will help me in developing my project.

The survey was produced online using the google forms format and then send the link to people that I want to survey. The following are the questions that I would like to ask people within my demographic as I believe they will help to provide with some clarity on what some of my specific user requirements should be. I have explained my reasoning for asking each question underneath each question.

1. Do you like to buy sneakers?

I am asking this question as the opinions of those who do enjoy buying sneakers and those who don't will also differ a lot. Therefore, this question is necessary to see what appeals to those in the

demographic that already enjoy buying sneakers to make the app appeal to them as they are easier to target, and also to see the views of those who aren't interested in sneaker buying to see if there could be anything I could do to try to appeal to them as well.

2. What type of sneakers do you like to buy?

I am asking this question to have an understanding on what type of shoes that the people in my demographic are already playing and to see what makes them like buying the sneakers to see if there is any general consensus of what type of sneakers young people buy.

3. What category describes you best?

I am asking this question to understand what type of people buy shoes and the reason for it. I want to know what everyone's attitude is towards fashion and sneakers so I can appeal to each group fairly and also to emphasise how big fashion is in people's lives.

4. How would you describe your fashion style?

This question was asked to enable me to understand the type of clothes that appeals to people in my demographic and to emphasise that trendy and streetwear clothing styles are what appeal to most of my target audience. The main purpose of the app is to only show the most popular clothing items and shoes because that's what a lot of people like to buy, so this question will prove it

5. Would you find using an app that notifies you on release/drops that cater to your fashion style helpful?

This question was asked to find out if the purpose of my app will solve the problems that people in my demographic face. If answered positively it will certify that my app could be potentially successful.

6. What current applications do you use to notify you on releases/drops?

As I will be needing to investigate and research existing systems and take note of some features within them and some features that they don't have, I will need to know what current notifying systems people in my demographic already use. For the features the apps don't have, I will try to implement into my own to make mine the most efficient and to meet the requirements of the user.

7. Do you find it frustrating when you're unable to get a nice shoe/item because it sells out too quickly?

I asked this question to see how people feel about an item that they want to purchase selling out very quickly. I know that it has a big effect on me and is very frustrating, so I want to find out if people felt the same way – if they do, I will know what user requirements that they want from my app and I will cater to their needs by implementing a system that makes it fair for everyone to get a chance at buying the item that they want.

8. Do you find resell prices too high?

This is another question to highlight the frustration of casual buyers when they have to buy their favourite item from a reseller who is giving them a crazy price to purchase. This information will help me with some features I want to implement to solve this.

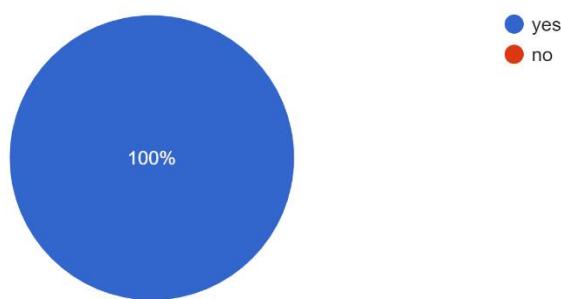
9. How much do you spend on clothes and sneakers yearly?

This question will show me how much money people spend on clothes and sneakers and shows me that they could utilise my app a lot if they spend a lot. It also emphasises how important fashion items are in people's lives.

Questionnaire – The Responses

do you like to buy sneakers ?

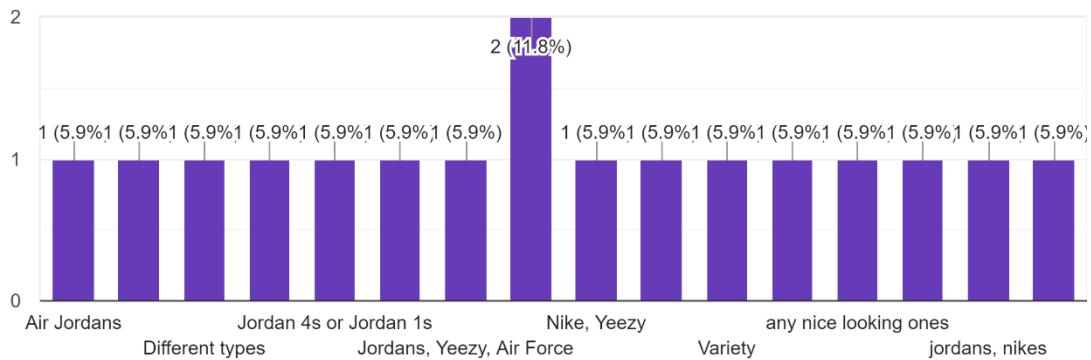
17 responses



Whilst the main target of the app are teenagers and people who like to buy sneakers for whatever reason, I still believed it was important to question people to clarify and check if there would be a variety of answers. This answers to this question cement my previous thoughts that people do enjoy buying shoes and this makes this app potentially useful for them. I also wanted to see if there were some people that didn't like to buy shoes as this app isn't exclusive to sneakers, but to popular fashion items as well (for example Trap star tracksuits)

what type of sneakers do you like to buy

17 responses



From this question I can conclude that a lot of people like to buy Jordan's, Nike and yeezy shoes and these brands are clearly the most popular, which gives me vital information as I can base my sneaker section of my app on these types of shoes.

What category describes you best?

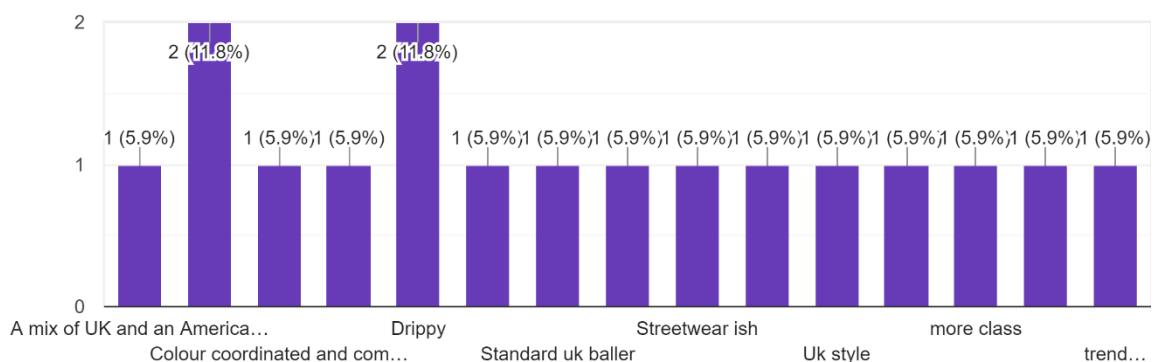
13 responses



No people that were questioned picked the option 'Not really into the latest popular fashion and clothes. This is clear evidence to me that everyone is interested in the latest popular clothing items. As expected, a lot of people are only casual dressers and buy fashion items for personal gain only so this is clear to me that they will be my biggest stakeholders. Resellers also come close though, so I have to make sure the app gives each group a chance as sometimes these two groups can clash, but ultimately, I will cater to the casual dressers more. There are also some sneaker collectors and people who are interested and love the latest fashion and shoes and this displays that there are a variety of different stakeholders and types of people who could be interested in my app.

how would you describe your fashion style

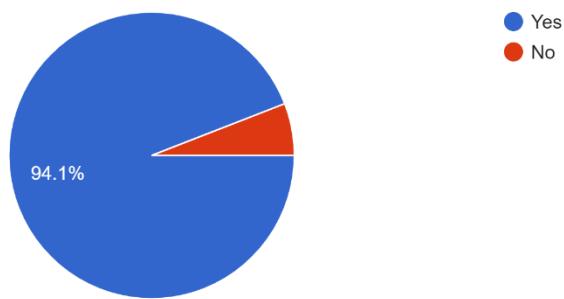
17 responses



There is a variety of answers here. I can infer from the answer trendy and 'drippy' that a lot of people do buy and wear very popular fashion items. The answer streetwear is also a very popular fashion style. A few people just said comfortable clothes which is very understandable, a lot of popular clothing items are very comfortable

would you find using an app that notifies you on release/drops that cater to your fashion style helpful?

17 responses



This is a question that was my asked to see if the audience and what percentage of them were interested in a system similar to the one that I am making. This would give me a clear indication of if the system I am creating would be successful or not and if my system would actually cater to my audience. An overwhelming majority of people (94.1%) responded with the option yes, giving me a good indication for my user requirements. However, a small percentage of people said no (5.9%). I will also take the opinion of these people into consideration and I would try to make improve my system in some way shape or form to turn that no into a yes.

what current applications do you use to notify you on releases/drops

17 responses

dropelist

SNKRS

snkrs

Sole Supplier

Nike

Dropelist and some supplier

dropelist, sneaktorious

Sole supplier

Standard tings

SNKRS

SNKRS

Basics like SNKRS

None

Adidas confirmed, Nike sneakers

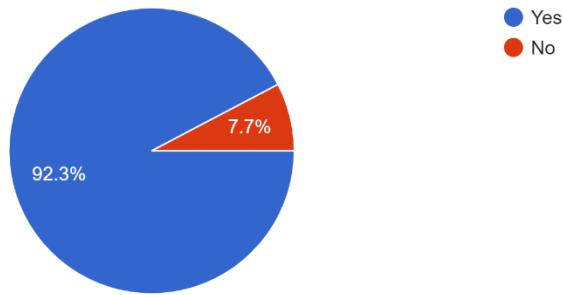
Drop list

A lot of different apps

The famous apps SNKRS and Adidas Confirmed were mentioned a lot here as expected as systems that already have the notification feature on their apps. However, as these apps are mainly retail apps, their apps focus would be more on getting sales rather than notifying users and making it fair for all kinds of users to access drops. The apps that are more similar to the system that I intend to create are the apps such as Sole Supplier and Drop list – such apps will be included in my product research. From these answers, I can conclude that a lot of people use similar apps with similar features to notify them on releases and that there aren't many other apps out there for the sole purpose of notifying users and providing them information on different releases, giving me more of an incentive to create my system. One person also said that they don't use any apps, which suggests my system would be a new concept for them and will help them even more than the other people.

Do you find it frustrating when you're unable to get a nice shoe/item because it sells out too quickly?

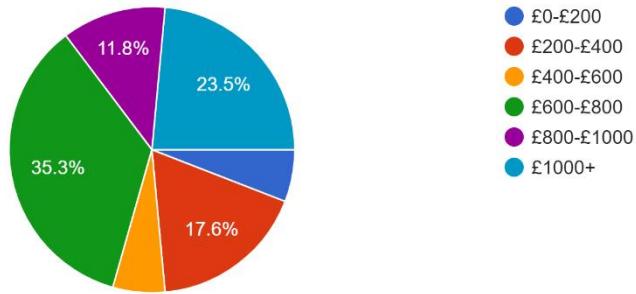
13 responses



From these answers I can conclude that the majority of people (92.3%) find it very frustrating when an item that they want to sell out too quickly. This highlights the problem that I am trying to resolve and proves that it is a real issue in people's everyday lives. The remaining 7.7% do not find it frustrating, meaning my system won't be as useful to them but they could steal utilise some features on the app if they want to.

how much money would you say you spend on clothes and sneakers yearly

17 responses



There is a large variety of answers for this question, this could highlight the different age groups and kinds of people that I sent my questionnaire to because of the different ranges of money spent on clothes and shoes yearly. I can conclude from these answers that a lot of people could utilise my app frequently because most people spend more than £600 pounds on clothes and sneakers yearly, and a large portion even more than £1000 which is a lot of money. Fashion is evidently a big thing in a lot of people's lives.

Current Systems

In this section, I will be taking a very detailed look into similar projects to the one my client would like me to create. From each example, I will go into detail with the different mechanics which the games contain and explain which ones I believe are good and bad in my opinion. I will then form a conclusion with each, stating what I would like to take and implement within this current project and explain how it will hopefully benefit the end users. By creating diagrams and annotated screen shots of the games, it would then create an abstract view of how they are all broken down, making it easier to understand what my client's plans are with me creating this project.

1. The Sole Supplier – probably the most popular of the existing systems that are similar to mine, Sole Supplier is an app that was founded in 2013 with the aims of providing accurate trainer news and release information for the UK and European market. They have a community of partners (e.g., brands and enthusiasts) that provide them with new and exclusive content daily that they can give out to people who check their app or website. The Sole Supplier was founded out of the CEO George Sullivan's personal frustration with the lack of information around the UK's trainer release dates, where to buy them and queuing up every weekend outside of stores. Acknowledging these problems, he decided to create a platform to make the process easier for other sneaker enthusiasts like himself.

Advantages and Best Features

- As soon as you enter the app or website, there is either a full home page with the all the latest drops or a pop-up page telling you the hottest sneaker releases right now. This type of system that doesn't waste any time and is straight to the point is something that I would like to implement into my system as well.
- There are three very important sections when you go to the footwear subpage on the home screen: "Best-selling trainers", "Upcoming trainer releases" and "Latest trainer releases". These 3 categories of footwear are probably the most vital when it comes to making the app the most successful and profitable. This is because the best-selling trainer section is where the majority of resellers, a big target audience for my app, would go to look for the trainers that they could invest in and make them profits. The upcoming trainer releases and latest trainer releases sections are significant for sneaker enthusiasts that are looking for release dates and restock dates for the shoes that they really want
- There is a reminders section that requires the user to sign up to the app to receive the latest information on shoes via email or notification.
- As well as a footwear section, there is a clothing section. This clear distinction is needed for people because it separates the sneaker enthusiasts from the fashion and clothing enthusiasts, but of course people can utilise both. It makes the app more efficient so that sneakers and clothing aren't mixed up into one.
- There is a news section which tells you all the rumoured releases that are expected to drop in the next year but do not have a set release date yet. It also gives first looks at shoes for sneaker or clothing designs that are yet to come out. The news section can also tell you if some release dates have been moved back and much more.
- Finally, there is a raffles section that gives users a link to enable them to enter raffles from different retailers, either from the UK or European. It tells you information for when the

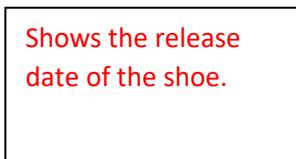
raffles close and when they opened. Raffles are important because it gives everyone a fair chance of being able to buy a sneaker because the winners of these raffles are completely automated and randomised. The most popular raffle app is SNKRS, which has been previously mentioned in the questionnaire section

Disadvantages

- The app seems more like a retail app with the amount of different clothes, sneakers and features it includes. It gives you a section for menswear and women's wear and all the different brands that are not so popular to the demographic that I am trying to target. Brands like Converse and Levi's are not so desirable among the young audience; therefore, it seems unnecessary to include this in a system that aims to provide you with the latest and most popular sneaker and fashion items.
- There is also Women's beauty news section on the app. This is also an unnecessary feature because it differentiates from the initial purpose of the app – to update people on the latest sneaker releases and provide them with accurate information
- Also, other unnecessary features include categories like socks and swim wear because these are not included in streetwear, which I want my app to be based around



Jordan 1 High OG University Blue

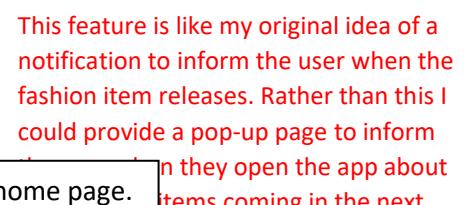
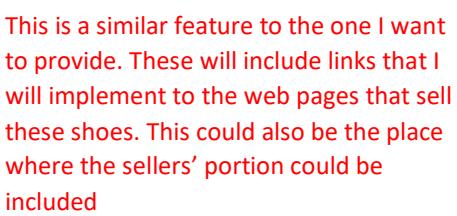
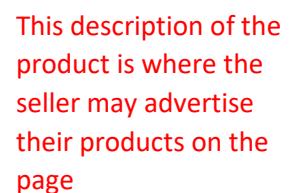


In Stock • Release Date : 6 Mar 8am • 555088-134
£135.00

The Jordan 1 High OG University Blue is the latest in a stream of blue Air Jordan colorways to have recently been unveiled, and it looks set to be a future classic. The AJ1 was first unveiled way back in 1985 as Michael Jordan's first

Where to buy

From: £135.00



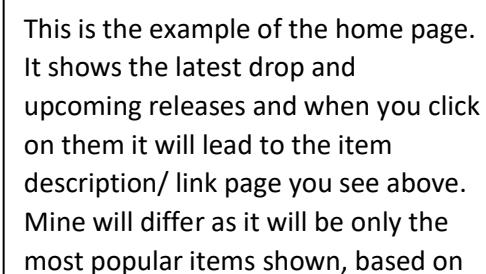
LATEST DROPS

| | | |
|------------------------------|-----------------------------|---------------------------------|
| Jordan 1 Mid University Gold | Jordan 1 Low SE Turf Orange | Nike Air Force 1 Pixel Sail Tan |
| £100.00 • In Stock | £105.00 • In Stock | £95.00 • In Stock |

| | | |
|------------------------------|----------------------------------|----------------------|
| Nike Air Max 95 Coconut Milk | Nike Dunk Low Disrupt Pale Ivory | Jordan MA2 Vast Grey |
| £155.00 • In Stock | £105.00 • In Stock | £115.00 • In Stock |

[View more drops.](#)

UPCOMING RELEASES



IPOS

Since I will be creating a similar system to this, it would be highly useful to monitor some of the processes which occur during this system.

| Requirement | Input | Process | Output | Implementation |
|--|--|---|--|----------------|
| User login | The user needs to create a unique username and password | Checking if the password and username are unique, and that they meet the correct validations | The user has created a new username and password that they can use to log in to my app and receive reminders for shoes | |
| Navigating to the website of the company that is selling the shoes displayed | The user needs to click on the link of the website of the shoe that he wants to view or potentially purchase | The URL that is clicked and the device gets the IP address of the URL and the browser is taken to the website via the TCP and HTTP. (HTML for website) | The website for the product that the user desires is rendered. | |
| Searching for your desired product | The user needs to input the name of the desired product that he wants to find | The name of the product is inputted into the search bar | All product matching or similar to the name searched are loaded up and shown to the user. | |

| | | | | |
|---|---|---|---|--|
| Getting restock and new product notifications and reminders | The user needs to sign up and turn on notifications for the app, to get the reminders and notifications for the desired products across the app | When a new sought-after product gets restocked or comes out for the first time, the app will send an automated notification to all their users. It will also send notification as a reminder just before the shoe comes out | A pop up will come showing that the product is about to come out or has just come out | |
|---|---|---|---|--|

2. SNKRS- This is one of the most popular apps that also came up when I conducted my questionnaire. Nike launched the Nike SNKRS app to serve as the brand's primary online distribution channel for their more highly-coveted sneakers. Since day one, the app became the go-to platform for purchasing limited sneakers directly from Nike and Jordan Brand. To gain full access to the app, users must first create a Nike+ account, then they'll be greeted by a list of upcoming/past sneaker releases on the home screen. Tapping any of the sneakers will display more images and background information regarding each release.

Advantages:

- App is home to almost all hyped releases
- Clean, easy-to-use interface

- Sneaker reservation opportunities

Disadvantages:

- Randomized raffle process
- Some products cost more in-app than in-store
- More product exclusions for store coupons

The screenshot shows the Nike SNKRS mobile application interface. At the top, there is a navigation bar with the Nike swoosh logo and the word "SNKRS". To the right are icons for a shopping cart and a menu. Below the navigation bar, there are three tabs: "Feed" (which is underlined), "In Stock", and "Upcoming".

The first card, titled "STREET SNKRS", features a man standing outdoors wearing a red zip-up hoodie and matching red pants. The text "STREET SNKRS" is prominently displayed in large white letters, with "Lewisham — 12.01" below it. A "Learn More" button is located at the bottom of this card.

The second card, titled "KICKCHECK", features a woman sitting on a chair, looking down at her feet. The text "KICKCHECK" is displayed in large white letters at the bottom of the card. A "Learn More" button is located at the bottom of this card.

Below these cards, there is a small image of a modern interior space with a ceiling featuring recessed lighting and a large potted plant.

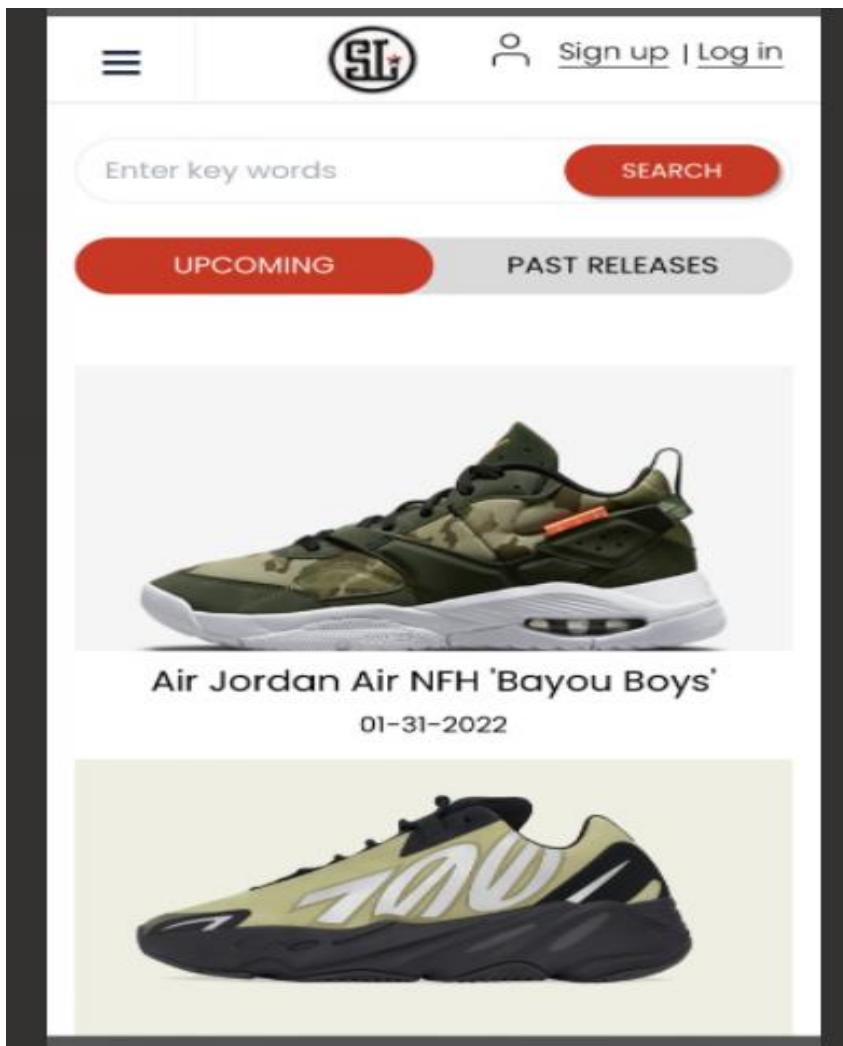
3. SoleLinks- Sole links is a free app available for Apple and Android users that sends notifications straight to your phone when certain shoes release or restock. Pressing Sole links notifications sends users directly to the site which saves valuable time and increases your chances of copping during those hectic releases. In addition to the alerts, the app also includes sneaker release information and a list of retailers that will be selling the shoes online that is extremely helpful when for manual users.

Advantages:

- App sends notifications as direct links
- Good for releases and restocks
- Provides links to US and International retailers

Disadvantages:

- App can freeze during certain releases
- Some app notifications don't come through



INTERVIEW

I decided to interview my end-user to have a more in depth understanding of the type of app that I should be creating to suit their needs. This was necessary as whilst the questionnaire could provide some insight, they were not specific enough and the questions were more about the notification apps that people already currently used and their current experiences with trying to get shoes as opposed to being about the type of app that they would like to suit their needs but does not currently exist. I decided to interview a 17-year-old male who already has history in the reselling region of sneakers and trendy fashion items. The reason for this was because a lot of people are just casual dressers and buy these clothes only to look trendy and cool. While there is nothing wrong with this, I decided to choose someone that would give a unique perspective as there are not a lot of resellers out there. Questions in bold text will be asked by me and below them will show the end-user's responses

Would you like the program to be on the computer or as an app for a phone?

An app because it would be more suitable for me because I am always on my phone so it would be more efficient. Also, as a phone is more portable it is easier to access the app anywhere and everywhere so I can always keep an eye out for possible money-making shoe opportunities. Although, a website could also be made into a widget on your home screen, giving it the same functionality as an app and will still enable portability for me. Also, both require an internet connection so if the program is a website, it will not matter too much

What kind of features would you like to see on the app?

I would most like to see a list of the best websites that people can buy the shoe from when they click on the item, it will be immensely helpful when trying to obtain a shoe on a release day.

What do you think of a unique list that the favourited items of the user are saved to and are then the items in which the user will be notified about?

I think this would be a very helpful feature as it will allow me to have a personalised feel to the app and also keep it concise as only the items that I really want will show up here, disregarding items that I have no interests in. However, sometimes I would like to receive notifications about different items just in case I missed an item to favourite, or when I just want to hear shoe news.

Would you utilise a contact feature and why?

Without a doubt, I love leaving feedback on a website especially if I had a pleasant experience on there, but also if I am having problems, I would like to contact the company so they can fix these problems promptly

How would you like the app to be designed?

I would prefer the app to have basic colours but also be very exciting at the same time. If you could figure out a way to do this it would be great. This is because I would want something that draws me in and is appealing but at the same time not too much that it becomes an eyesore. I will expect you to find the perfect balance

What would you like to see when you enter the website/app?

I would like to be loaded into a home page with lots of information on the hottest drop and the most recent drop and being able to click on it to go straight into action. This is because it will reduce time spent navigation through the app to find the most wanted item around that time, and it would make more sense of this item just to pop up on the home screen to increase my chances of getting that product.

Finally, which information would you like to see each product store in the product database?

The information that would be useful to me such as release dates and exact times, any possible delays in releases and also a brief description and picture of the shoe.

Proposed Solution

From all of my research and analysis, I have proposed website similar to the Sole Suppliers app that will give release and restock information on the most desired shoes in the world right now. Different

from the sole supplier, I will also include streetwear products as brands such as trapstar and Cortez have been rising in demand lately, and are also selling for a good amount on non-retail stores such as Ebay. In my app I will include a footwear feature and a streetwear feature, allowing the user to view different categories under those names and will also allow easy navigation through the website. When an item is clicked on, it will display its information and links to the website wear it is currently selling and the price it is selling at. The home page will include a lot of information on the most popular drop right now so the user doesn't have to waste time navigating through the app if he has only one desired goal. I will also have a favouriting system where the user can favourite items as long as they have logged in and these items will then be added to a reminders and favourites list and these will be the items that the user will be notified on. Any item a user wishes to see reminders on, it will pop up immediately when entering the app or a reminder will be sent via the email they have signed up with. Finally, I will have a contact section for general feedback from users.

User/Client Requirements

After doing my problem research I can begin to form the basis of the type of game that I would like to create for my project. I will go through the Client Requirements which goes through the requirements of my stakeholders as well as from my questionnaire and interviewing that I felt would be useful to include within my game. Then design requirements which are any necessary design elements that I found would likely be necessary to include, here I will use a lot of my research found from my current systems research. Next, I will go through functional requirements which are the requirements of how the system should react to certain inputs to provide a certain output. Finally, I will go through some non-functional requirements which are more about the characteristics of the product and the user-experience.

| Requirement no. | Requirement |
|-----------------|---|
| R1 | The game should be suited to any type of demographic, such as resellers and casual buyers |
| R2 | The website should be easy to navigate around |
| R3 | The app should have multiple useful features, such as favouriting desired items |
| R4 | The website should be accessible on all platforms. |
| R5 | The first page of the website |

| | |
|----|---|
| | should give you essential information immediately on the hottest drop that week, with pictures and more for it to look nice |
| R6 | The notifications given on chosen items should be very quick so it gives more of a chance of coping the item |
| R7 | Users should be able to contact us when they run into an issue |

Design requirements (Hardware and software)

| Requirement no. | Requirement | Justification |
|-----------------|--|--|
| R8 | Have a specific theme throughout the application | To ensure the app is consistent throughout and looks interesting to use. The theme should be appealing to the age group which I am targeting which is teenagers. However, the colour scheme to go with the theme shouldn't be too bright and there should only be a few set colours. This is because the app shouldn't be crowded with colours as it may be too overwhelming for the user. |

| | | |
|-----|---|---|
| R9 | All users should have their own profile to keep track of their progress data | Allowing users to have their own account to log on to ensures that their data is specific to them if they use a different device. Their favourites list and desired reminders should always be maintained |
| R10 | Internet connection (Software) | This is because the database will be online so and internet connection will be required to sync the data |
| R11 | A platform for a HTML, CSS and Java Editor. FTP client to upload files to web server (Software) | This is to write up the code for the website and to design the aesthetic of the page. Also, java script is used for validations etc. |
| R12 | Computer/Laptop (Hardware) | This hardware is needed for all the software to run on |
| R13 | Touch Screen (Hardware) | The features will require a touch screen to be used if your accessing the website on your phone |
| R14 | 2 GB RAM (Hardware) | This is the minimum RAM required to run this program |

Functional Requirements

| Requirement no. | Requirement | Justification |
|-----------------|---------------------------------|--|
| R15 | User will be able to login with | This means that each user's data will be stored for them |

| | | |
|-----|--|---|
| | a username and password | and it makes the app personal and will remember their settings and data |
| R16 | User can search for shoes and clothing items in a product database | The search engine for searching shoes should be smooth and efficient and all the right products similar to the word searched should come up. |
| R17 | User can create account to receive restocks and reminders | Users should be able to create an account if they are a new user so their favourites list can be personal to their own account. It should validate the passwords to ensure the user enters the correct password which they have chosen. |
| R18 | To be able to develop the game I will need to have Visual Studio downloaded onto my computer to be able to code Using HTML, Java and CSS | To be able to code I will need to download this software to my device to start my process. It runs on various platforms and it is very good for JavaScript and is the top pick for web developers, which is the reason I am choosing it as I am creating a website. |
| R19 | Program should have a menu to access the login page, reminders, features and homepage. | This is for quick search and navigation through different features of the site |
| R20 | Use drop down menu with | The shoes should display |

| | | |
|-----|---|---|
| | categories under a feature. Display all the shoes and clothing within the sub-category | depending on which sub category you pick (e.g. Latest Drops under Footwear). This is most efficient when using sub categories and drop-down menus |
| R21 | Users are able to log out and log in easily | If the user wants to switch to another account or log out for any other reason, it should be quick to further the smoothness of the website. |
| R22 | There should be multiple links for each product (depending on how many sites the product is sold on). Users can click a link successfully to go and buy the shoe that they want | This is because a single shoe can have multiple sites which it could be sold on. Giving users all these sites using hyperlinks and icons would be efficient, whilst maintain the aesthetic of the website |
| R23 | User can reset password. | This is in the case of a forgotten password, so the user doesn't have to create a new account. |

Non-Functional Requirements

| Requirement no. | Requirement | Justification |
|-----------------|---|---|
| R24 | When the website loads up, the first screen should be the home page with a menu of icons with links to each of the features in the site | This is so the user can navigate around the app easily and can find the feature they want quickly. The icons help the visual appeal of the app. |
| R25 | User should be able to mark | This will make it easier and faster |

| | | |
|-----|--|---|
| | sneakers as “favourite” and these foods will appear in a favourites list and will be the shoes that the user gets restock/release notifications about. | for the user to find products that they are interested in and it will give them the ability to personalise their app to fit them and receive reminders about their favourite item. |
| R26 | User should be able to tap on a product and a pop up of the information about that product will appear | This is to help the user view and retrieve necessary information that they require about the product |
| R27 | The program should be easy to use and navigate for someone of any age | This is essential because easy navigation and smoothness in a website is vital to your experience on the website, so people don't get confused or have problems when trying to find the things that they're searching for |

Limitations

It is necessary to identify some of the limitations that could limit me in developing my project, including the limitations of the hardware and software which I will use to develop my solution.

Sorting and filtering when searching for shoes could be a possible limitation when trying to develop my website, could limit how good my website could be as knowing how to code that part would be too complex for me at this stage as I have not yet learnt that. Another limitation I could face is trying to build the database for the products, usernames and passwords.

The laptop hardware could also be a possible limitation as the laptop I'm using could be susceptible to lagging and a lack of storage. My laptop could also not allow me to download certain apps because of the software requirements.

A further limitation is the fact that I have not used HTML, Java and CSS in the past a significant amount. However, there are many resources available online, and so I believe I am capable of learning this language appropriately.

Success Criteria

| Criteria no. | Success Criteria | Justification |
|--------------|--|---|
| SC1 | User should be able to successfully login with their correct username and password and allow multiple attempts to login. | This is to check that the user can login correctly with their username and password with both being correct to allow them to be logged in. |
| SC2 | When a user favourites the item, that is the item the user should be continually receiving notifications on the shoe, such as release date. | This is so the user can always be alert and up to date with the release dates and information of the shoe |
| SC3 | The app should have many visual icons in the menu and in adding notes and should have the option for the user to change the font size and colour | This will make the app accessible to everyone and will allow any demographic to use the website and also give the user an enjoyable experience when on the website. |
| SC4 | User should be able to search the product database and most relevant products matching the search appear. | This is to make the searching function more effective and ensuring that the most relevant searches appear first |

| | | |
|-----|--|--|
| | | on the list making it quicker for the user. |
| SC5 | Display all the shoes within the sub-category | Less memory used if its all on one page and will have no need to be linked to another page of the website. |
| SC6 | Program should have a menu to access the login page, reminders, features and homepage. | Makes the website more efficient to use and allows for more easier navigation |
| SC7 | When a product is clicked on, information about that product should be displayed and links to where you can buy the product should be displayed as well. | Makes it efficient which is important if someone wants to buy a sought-after shoe quickly straight after release before it sells out |

1.9 – Client Sign-off

I, as the client, am happy with the analysis.

Name: Richard Chinedu



Signed: _____

Date: 01/09/2021

DESIGN

2.1 – Specification

To aid with the decomposition of the project, in this section I will create a list of all the necessary objectives and requirements in terms of the inputs, processes, storage and outputs needed to create the system effectively and to the client's brief. Then, I will list the aesthetic and secondary objectives which are features that are not necessities, but will help in the usability and attractiveness of the app. Finally, I will list internal objectives which are personal objectives that I will need to meet in order to effectively create my app.

2.1.1 – Inputs

- User can enter username and password
- User can create a new profile and enter their relevant details
- User needs to be able to click on a link successfully
- User can input item that they're searching for into the search tab
 - Turning on reminder notifications next to a certain product

2.1.2 – Processes

- Username and password are compared against database
- App will login if the input matches the database, and will prompt a retry if input does not match
- Inputted product will display along with its information and more
- Based on user's search, the database of foods is queried to give search result
- The clicked link would be in the process of loading to the website
- When the user has turned on a notification for a shoe and the shoe is set to release, a notification pops up is sent to their device
- When the pop up is clicked on, the user is taken directly to the page of the shoe

2.1.3 – Storage

- Shoes and Products are stored in database and when searched for they will be displayed
- User's information is stored in a database
- Website links are also stored

2.1.4 – Outputs

- App will log user into app if login is correct
- If login failed, warning messages come up prompting user to retry
- User is taken to the website of the link clicked and the product is displayed ready for them to buy
- A user news profile is successfully created ready for them to log in
- User is taken to the page where he can access the shoe after he clicks the notification pop up
- All products similar to the user search displays for the user to find the product that he was trying to look for

- Internal Objectives

Become familiar with Visual Base Studio

- This is what I will be using to create my website and I will need to learn how to create an interface and how to make different features interact with each other. I can do this by creating basic practice apps following tutorials.
- Develop knowledge of Java programming language and syntax
- I will have to ensure I am confident in using the HTML and Java language to both program the features of my app and create an interface. I will also practice this by following relevant tutorials.
- Become confident in my knowledge of SQL or firebase to create Database and database relationships
- As a main feature of my app is the calculations, I will talk to the end-users and ensure I program my app to carry out the correct calculations in the correct way using correct units. I will also need to ensure the app gives good advice based on calculations by expanding my knowledge of diabetes.
- Develop my skills of programming with a database
- Since my app uses a database, I will need to ensure that I am confident in reading and writing data to and from a database along with performing effective queries and creating relevant entity relationships. I will develop this by practicing using databases in a program.

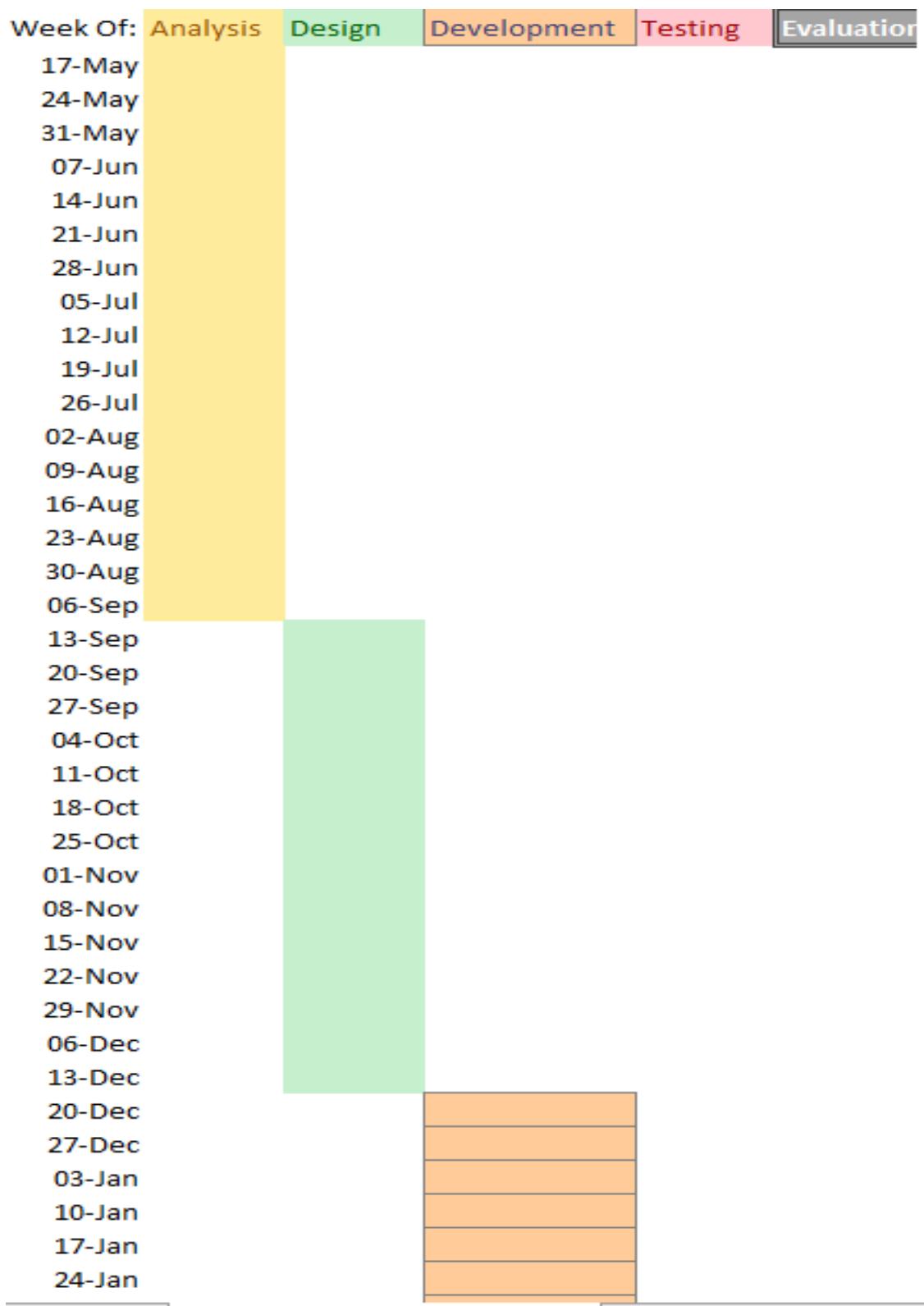
End-User Objectives

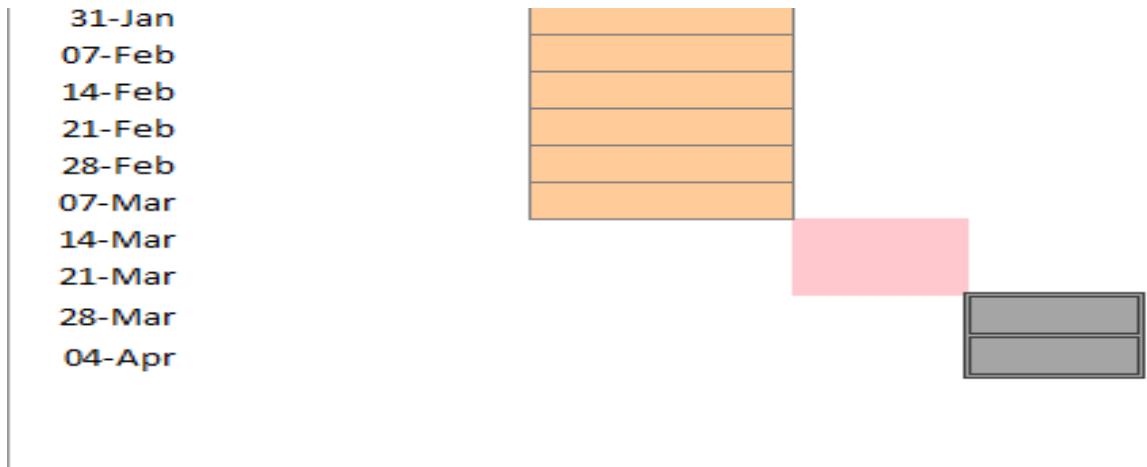
These are the objectives I have received from my end user, Richard Chinedu:

- The app must have quick and efficient links to websites when I click on the shoe from a notification or directly from the app
- The app must run quickly and load up quickly without fail, especially in a hot shoe drop when a lot of apps usually crash due to the high number of users trying to access it at the same time
- Navigating to find shoes should be easy
- Being able to contact the website
- Should be able to favourite items and then receive constant reminders of those shoes after I have signed up
- The app must also have a nice aesthetic
- The most popular shoes and products only with the most profit should be shown to avoid wasting time and making the best out of my experience on the app.
- Release date and time should be clearly shown next to the item, along with the websites that it would release on

Scheduling

This project plan is just a small base mark of when I should be finishing each part of my project. This chart shows how I will use my time effectively to make use of the time available and finish my project on time. It will ensure that each part of my project is done in a certain and rational order and the deadlines that I set myself will help me keep on top of the creation of my app.



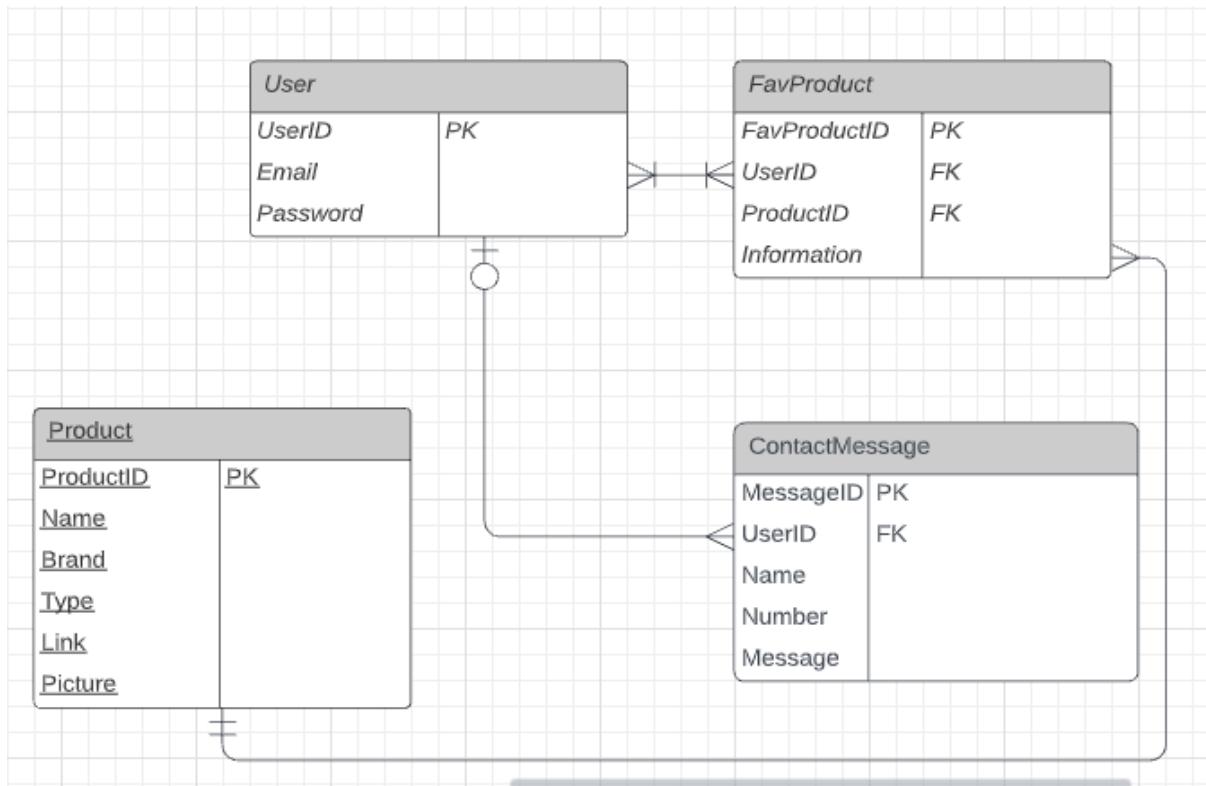


2.2 – Decomposition

Decomposition involves breaking down the problem into smaller more manageable problems to make the task easier to tackle. Each problem can then be solved individually, and then combined to form the final solution. This ensures that each element works individually and can be developed against the requirements for that subtask. For my app, I will start with the development of the login database system and the design of the website and its navigation as it is necessary that this section works for the development of other subtasks, such as the development of the reminder system. A search function for the products could be developed after but is not the most important feature of my system as the user can easily navigate throughout the website and find the products that they might need without the need for inputting a keyword into the search engine.

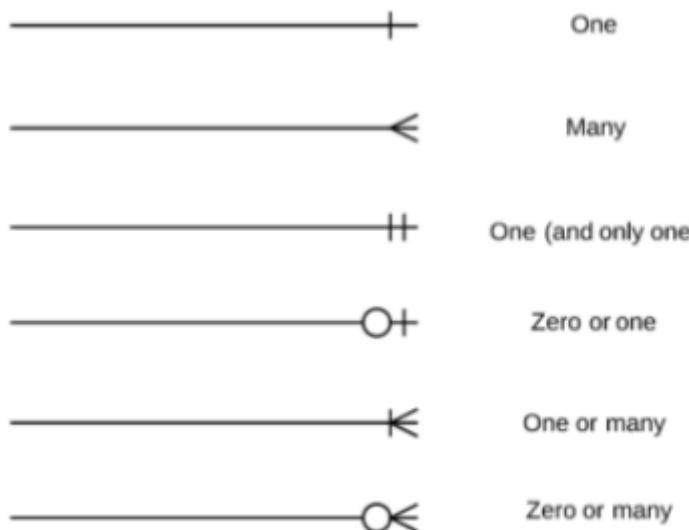
2.2.1 – Database

2.2.1.1 – Entity Relationship Diagram



An entity relationship is used to show how the tables within the database are related and

how, including which fields are the primary and foreign keys in each table (using PK and FK respectively to represent them). It helps to show a visual representation of the database to ensure that all tables will be linked in the correct way and no relationships are missed. My diagram shows how the users will be allowed to favourite multiple products so that they can receive information and reminders on those products. The user will also be allowed to leave many messages as feedback or if they had any problems with the website and wishes to contact the owner (me), while also showing that someone can leave a message without necessarily having to be a user. It also shows that one product can have many people that have favourited that product via the relation between product and favourite product. The image below defines the relational lines on my visual representation of the database that will be in my system above.



Within the database, each table will have to be validated to ensure that the data being entered is correct and reasonable for the table and precautions will be made to avoid duplicate data and any issues that may come with deleting data.

2.2.1.2 – Database Normalisation

Database normalisation is the process of organising a relational database to ensure that there is minimal data redundancy and improved data integrity. There are multiple normal forms used in database design, but the first three are the most used. Applying these normal forms to a database will help to eliminate errors in the creation of the database which will then help to eliminate errors when using the database in the programmed system.

The first normal form (1NF) says that each column and each table should contain one value and no repeating groups. This means that there are no fields left blank and no fields have multiple values in. This is because if the value in the field is required for a calculation, there must be a single value in the field for this to work. If a database violates the first normal form, the solution would be to create a new table that uses the primary key of the first table as a foreign key. This is necessary to creating my database because it helps to avoid problems with calculations and queries when programming my solution.

Before going to the second normal form, the database must first be in 1NF. 2NF says that and non-key field should be dependent on the entire primary key – this is only a problem when using composite primary keys. In the database, all information in a record should be related to the primary key and be about the same thing. It also includes not duplicating data across the record and you must not be able to determine any of the fields from the primary key. This ensures that all data stored is correct for the primary key and relevant as to not complicate the database.

Before going to the third normal form (3NF), the database must first be in 1NF and 2NF. 3NF says that no non-key field is dependent on any other non-key field. This is to make sure that the table does not include any irrelevant fields that could be determined by another field in the table. This is avoided by creating a new table with the non-key fields that are dependent on each other and using the primary key from this table in the original table as a foreign key.

Data Dictionary

User

| UserID | This will be the primary key of this table used to uniquely identify each user | Text | 96 |
|--------|---|------|----------------------|
| Email | Unique email address in the format with an @ which will be linked to the password | Text | dogunrin@outlook.com |

| | | | |
|----------|--|------|----------|
| Password | Each user needs a password at least 8 characters long made of at least one capital letter, one lowercase letter and one number used for logging in | Text | Pa55word |
|----------|--|------|----------|

ContactMessage

| | | | |
|-----------|--|------|-------------------|
| | | | |
| MessageID | This will be the primary key of this table used to uniquely identify each message | Text | 96 |
| UserID | Foreign key used to link this table to the user table and identify users in this table | Text | 67 |
| Name | User will need to enter their name if they went to send contact the owner | Text | Daniel Ogunrin |
| Number | User will need to enter their number if they went to send contact the owner | Text | 07533820762 |
| Message | This will be the message where the user can send the desired message | Text | I love this site! |

ProductID

| | | | |
|-----------|---|------|----|
| | | | |
| ProductID | This will be the primary key of this table used to uniquely identify each product | Text | 35 |

| | | | |
|---------|--|----------|---------------------------------|
| Name | Stores name of the item | Text | Yeezy 350 V2 "Dazzling Blue" |
| Brand | Stores brand name | Text | Nike |
| Type | Stores type of product, streetwear, or footwear | Text | Footwear |
| Link | Stores link of product to best place to buy from | URL text | |
| Picture | Stores image of how the product looks | JPG | image |

FavProduct

| | | | |
|--------------|---|------|------------------------------------|
| | | | |
| FavProductID | This will be the primary key of this table used to uniquely identify each favourite product | Text | 39 |
| UserID | Foreign key used to link this table to the user table and identify users in this table | Text | 97 |
| ProductID | Foreign key used to link this table to the products table and identify products in this table | Text | 45 |
| Information | Stores information of product including release date | Text | March 11 th 8:00 AM BST |

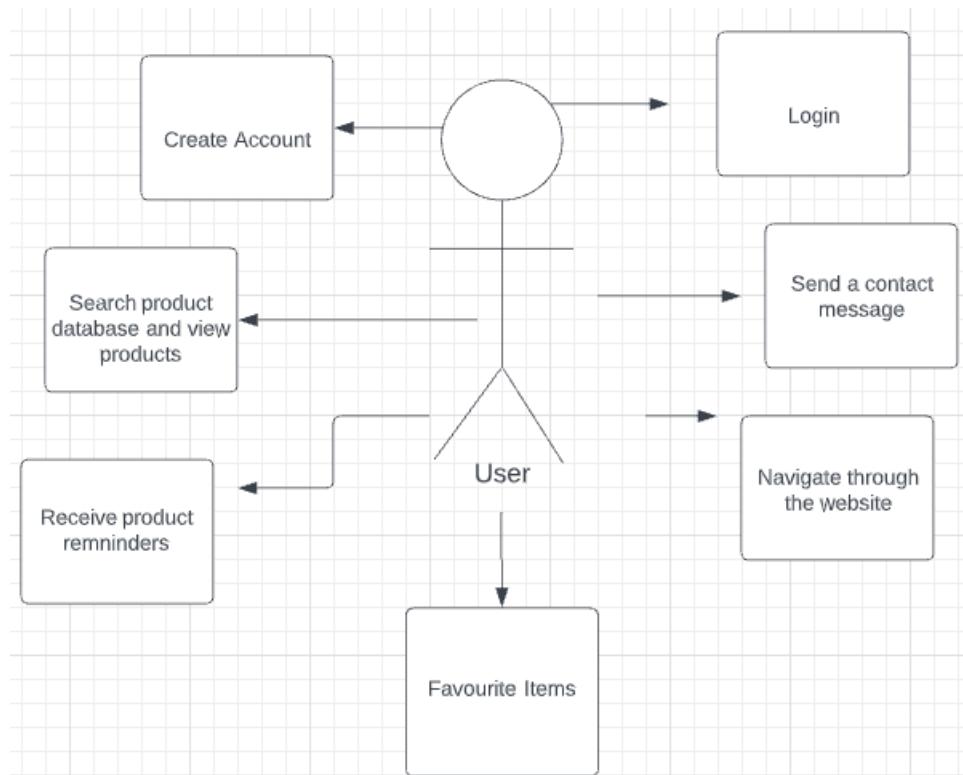
2.2.2

– Proposed System DFD

For my proposed system, I have made a level 1 DFD to show the systems processes, any data stored and how this will interact with outside entities such as the user. Each arrow shows how the processes will flow and the descriptions on the arrows show the inputs needed for the process and

the result of that process. This will be very useful in my creating my program and in my decomposition as it shows which processes need to be fully working and be able produce the desired outcome for the next process to work properly. It also helps me create my database as it shows what needs to be stored which I can use to ensure I have all the relevant tables in my database. It shows me at which point the data needs to be stored and therefore at what point the database will need to be accessed. The diagram shows me what the user may input into the system and what they will be able to see from these inputs and how this might vary between the patient and the doctor.

2.2.3 – Use Case Diagram



A use case diagram is used to show what each user can do with the system without focusing on how to do it. In the diagram, the user is represented by the labelled stickman image user and what can be done by this user is represented using arrows and squares. The arrows show which activities can be done by each user to visually represent how the app can be used by different users. This is helpful because it shows the basic decomposition of the program and what the website needs to be able to do in terms of the basic functions for each user. It also helps the user to know what functions and how many functions that he can do on my system. My diagram shows that the user has a lot more features on the website to use than me as the administrator, so there was no need to draw a diagram for the owner of the website.

The tables below specify the details of each use-case including which actor it applies to, a description, the preconditions (if necessary) and the normal course that the case will follow. It also shows any alternative courses and any exceptions (e.g., when something fails).

| Login | |
|-------|------|
| Actor | User |

| | |
|----------------------|--|
| Description | This use case describes a user logging in to the app |
| Preconditions | Database is online |
| Normal Course | <ol style="list-style-type: none"> 1. System displays home page 2. User selects login option 3. User enters email 4. User enters password 5. User clicks login button 6. Website loads home page |
| Exceptions | <p>E1. Login failed (occurs at step 5)</p> <ol style="list-style-type: none"> 1. System displays message that username/password is incorrect 2. Prompts user to try again |

| Favouriting Items | |
|--------------------------|---|
| Actor | Patient |
| Description | This use case describes a user favouriting an item which is added to their reminders list |
| Preconditions | Database is online and user is logged in |
| Normal Course | <ol style="list-style-type: none"> 1. User favourites desired item 2. Item is added to the users personalised favourites list 3. User can view more information on those items and will be notified when the item releases |
| Exceptions | <p>E1. User is not logged in</p> <ol style="list-style-type: none"> 1. website warns users that they need to be signed in to favourite items. Prompts user to log in. |

| Searching product database and viewing products | |
|--|---|
| Actor | User |
| Description | This use case describes a user searching for their desired item and viewing them |
| Preconditions | Database is online |
| Normal Course | <ol style="list-style-type: none"> 1. User searches for product by name/brand/keyword 2. User selects product from search list 3. Products that are similar to user search are displayed to the user |

| | |
|---------------------------|--|
| Alternative course | A1. User can use the top navigation bar to navigate through the website and find the products they desire themselves as there are not many products on the website to begin with, as only the most popular products are on the site to begin with. |
| Exceptions | E1. No search results are given (occurs at step 1) 1. Message is displayed saying that no search results are found |

| Send contact message | |
|-----------------------------|--|
| Actor | User, anyone |
| Description | This use case describes someone sending a message to contact the company/owner of the website |
| Preconditions | Database is online |
| Normal Course | 1. User enters name 2. User enters email 3. User enters number 4. User enters the message they wish for me to receive 5. Thank you for contacting us message is shown after the user clicks send |
| Exceptions | E1. Fails to enter one of these fields and clicks the send button 1. Message is displayed saying this is a required field |

| Receiving reminders as notifications | |
|---|--|
| Actor | User |
| Description | This use case describes a user receiving notifications on their desired item |
| Preconditions | User is logged in |
| Normal Course | 1. User favourites an item 2. Item is added to the reminder list 3. When item is going to be released, an automated reminder/notification/pop-up is sent and displayed to the user |

| | |
|-------------------|--|
| | 4. User clicks on item and is taken to the where to buy screen with the links |
| Exceptions | E1. A product is delayed or cancelled 1. The next time the user logs into the app a message is shown informing him that they product that they wished to receive reminders on has been delayed or cancelled |

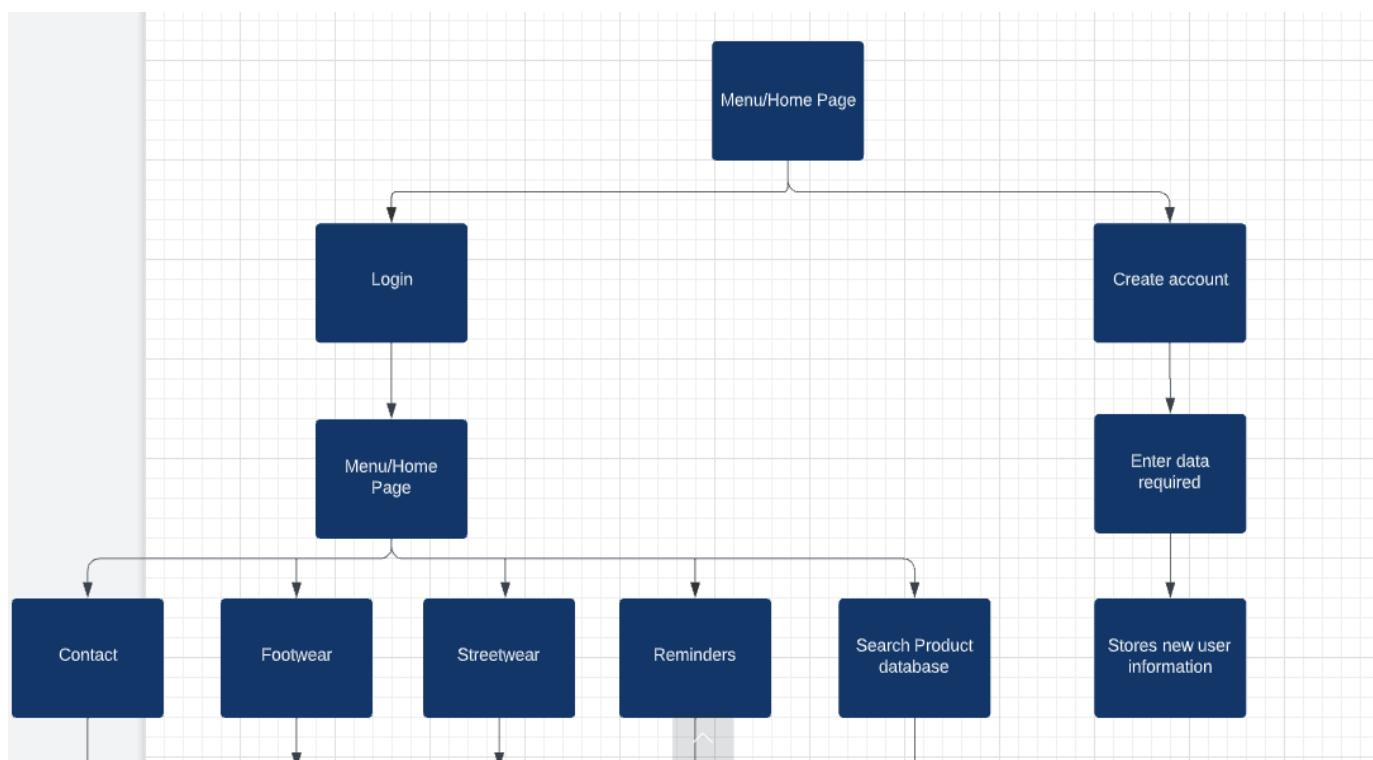
| Navigating through the website | |
|---------------------------------------|--|
| Actor | User, anyone |
| Description | This use case describes a user easily navigating through the website |
| Preconditions | Database is online |
| Normal Course | 1. Home screen is loaded up for the user 2. Top Navigation bar and a scrollbar to the right is displayed for easy navigation for the user 3. When the user wants to click on a feature for example under footwear, the user can hover over the footwear section and a drop down will display with each category 4. User clicks on category and is taken to that page. |

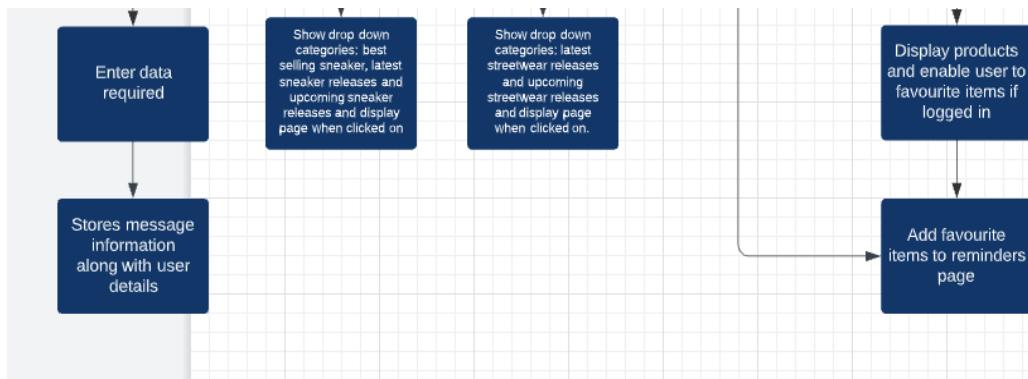
| Create Account | |
|-----------------------|--|
| Actor | User |
| Description | This use case describes a user creating an account for the app |
| Preconditions | Database is online |
| Normal Course | 1. System displays login page with create account link 2. User selects create account option 3. User enters all required personal details 4. User clicks create account button 5. Website saves data to database 6. Log in screen is loaded |

| | |
|-------------------|--|
| Exceptions | <p>E1. Data is missing (occurs at step 3)</p> <ol style="list-style-type: none"> 1. System displays message that shows which data is missing 2. Prompts user to try again <p>E2. Data is invalid, and account cannot be created (occurs at step 4)</p> <ol style="list-style-type: none"> 1. System displays message that shows which data is missing 2. Prompts user to try again |
|-------------------|--|

2.2.4

– System Design





A system design diagram is used to show the basic function of the website and how these flows to the user. It will be helpful in the development of the solution as it shows which activities flow into which other activities and in some cases, shows which features must be fully working before the next section can be tackled due to the nature of the website and how it flows. Along with my use case diagram, this diagram is extremely useful in showing how the website is used by the different users (patients and doctors) and the features that will be needed to be separated and the features that need to be shared. This diagram will also help me to ensure that I program my website in the correct order and that everything flows together as expected. When developing and testing my website, I can use this diagram as a basis of the order the app should run things and the options that will be available on the different screens to ensure that program flow is correct and logical.

2.2.5 – CRUD

User

• Create

- The user needs to be able to create an account for the website. By doing this they will create a record in the database which will include their personal information that they have entered.
- The user can create a new contact message and press send which will create a new record in the database.

• Read

- In order to use any of the features on the app, once the user has logged in the databases containing their information must be read to the app in order to display relevant information on the screen. The user cannot read any information from the database containing information regarding any other user.
- If the user changes their details on the app, they must be able to read data from the database. They cannot access any other users' details.

• Update

- The user will be able to change their own personal details and update the database. They will not be able to update the information of any other user

- The patient will be able to edit their favourite products and subsequently edit their reminders list and save over them in the database. They will not be able to change any other users' list.

- Delete

- The user will be able to delete items from the reminders list that they have added by clicking on the heart button again to unheart the item. They will not be able to delete the entries of any other users in the product database

2.2.6 - Security

Security in an app is very important. The app also includes personal information of the user which will also need to be protected. The main method of securing the app is by using usernames and passwords. The usernames will be unique to each user and the passwords will be chosen by the user but will have constraints, for example, they must be longer than 8 characters, and include a number, a capital letter and a special character. This will make the passwords more secure and less susceptible to hackers. Requiring a username and password also makes it more secure as both have to be correct for access to be granted.

When a user of the website is typing a contact message, their number and name and email will need to be protected from the outside as well, which will mean the database needs to be secure

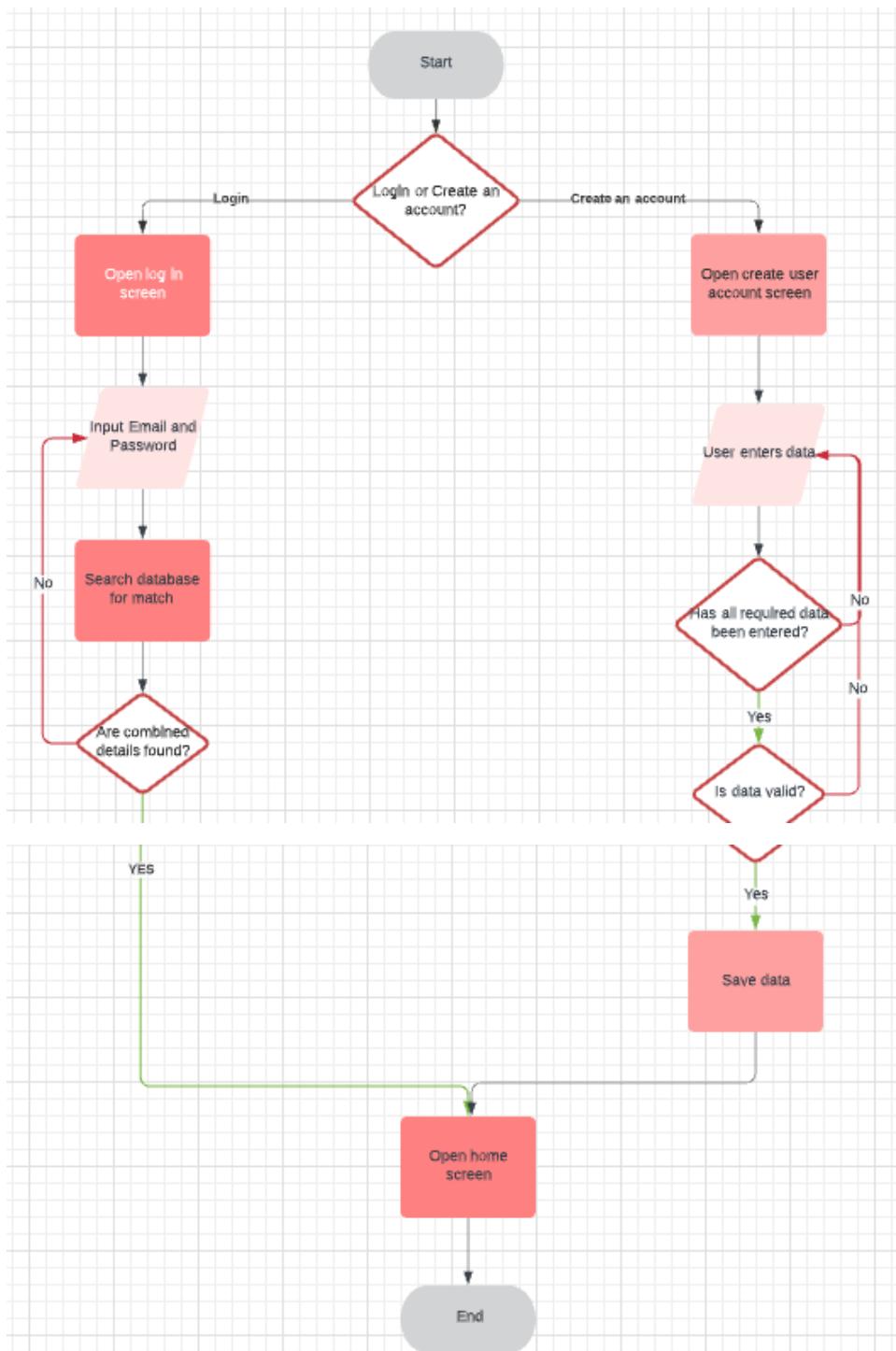
If either user decides to change their personal details, they will also have to re-enter their password. This is to make sure that no one else can change their details on their behalf for example change their username so that the user cannot log back into the account.

Algorithms

Flowcharts and Pseudocode

As part of my design, I have decided to decompose the task of programming this project into a series of separate tasks, as seen below. I have taken each task, designed a flowchart explaining the process of completing this task, along with, where deemed necessary, pseudocode.

Logging in or Creating an Account

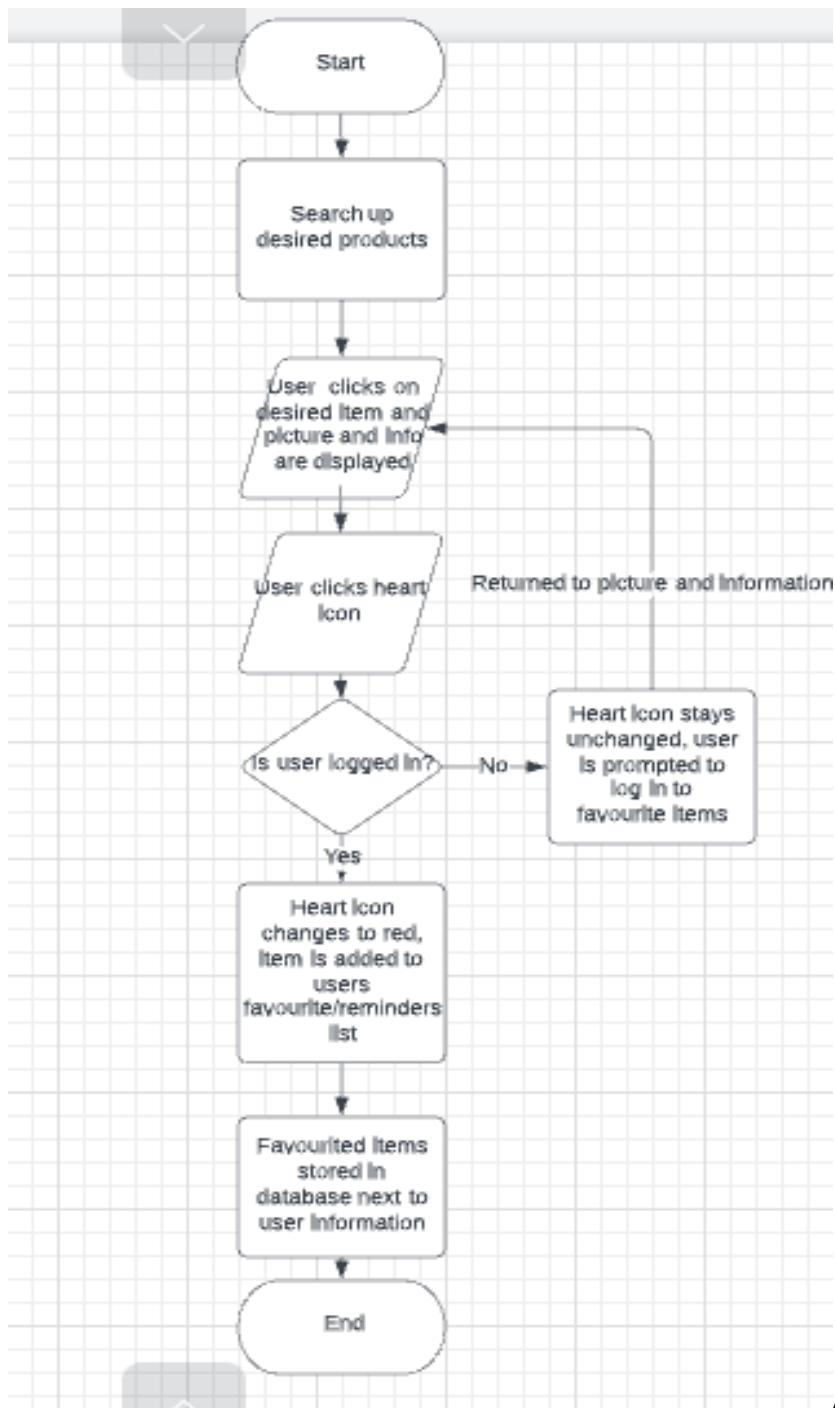


To login, the users will click the login button from the home screen and the website will take them to the relevant screen. The user will then be prompted to enter their email and password and click the login button. The database will then be searched and the email and password of each record in the database will be compared to the input email and password. The data in the database must be read in pairs as the email and password must both be for the same user in order to be correct. If the email or password is incorrect/not entered, a message box will appear informing the user of the failed login and they will be allowed to reattempt to log in. Once the email and password are both correct and have been found in the database, the home screen will load and the data for that individual will be loaded.

To create an account, the user must first enter all of the relevant and required information in the valid format. If any of the required fields are not filled out, or if anything is filled out invalidly, the user will not be able to create the account. Once all the data is entered properly, and the user clicks the create account button, the data will be added to the relevant database and the account will be created. Since this is equivalent to logging in, they will also be taken straight to the home screen rather than have to log in again as this could be annoying and is unnecessary.

While this flowchart looks relatively large and complex, the process of creating an account and creating the log in system are virtually the same meaning that in the implementation, it may be useful to make use of procedural programming in order to avoid re-writing code. Using this method will reduce the amount of code by making use of existing code and will also make it easier to implement any changes and avoid having to make the edits twice (which would also be more likely to lead to an error). The flowchart also shows that there is a lot of iteration that will be needed, however upon closer inspection these iterative sections are to do with the validation of the data and ensuring that the user is able to re-enter data if they have made a mistake or missed something out. This means that this section should be relatively straight forward to implement as long as the flow of the program is followed and the loops are implemented correctly so that they start and stop when needed.

Favouriting Items



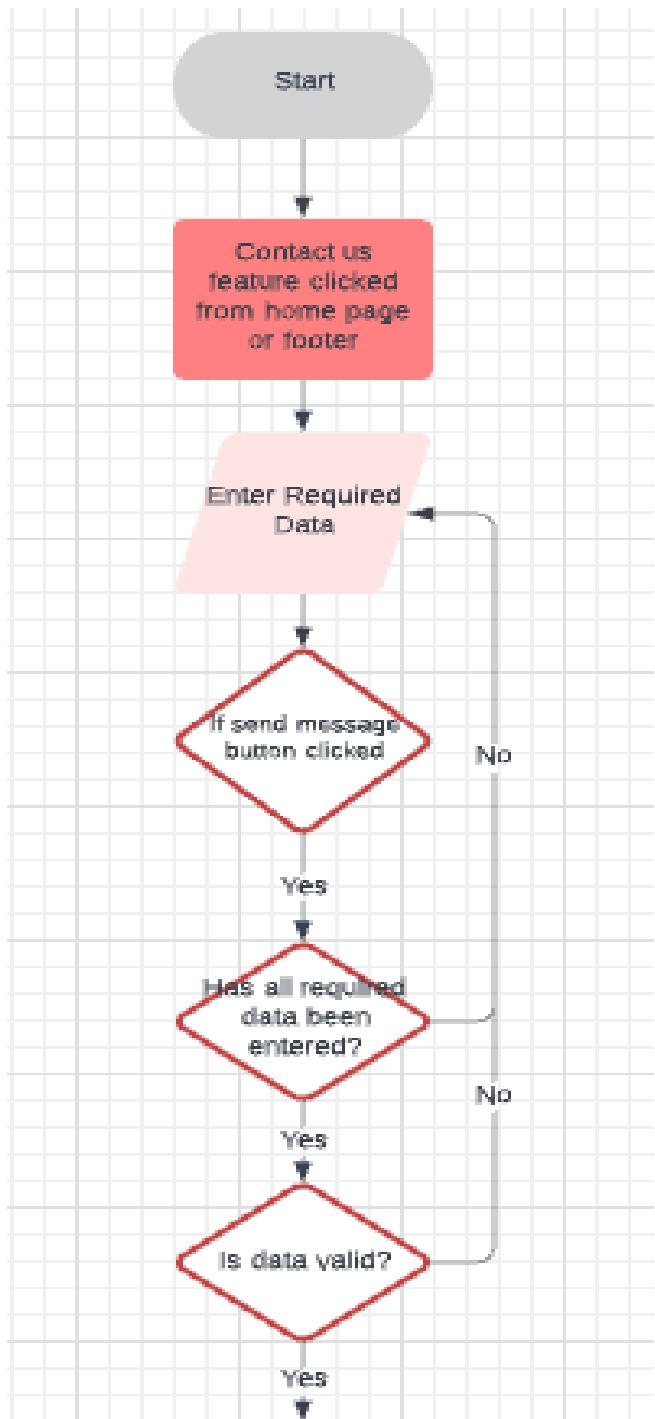
Above is the flowchart

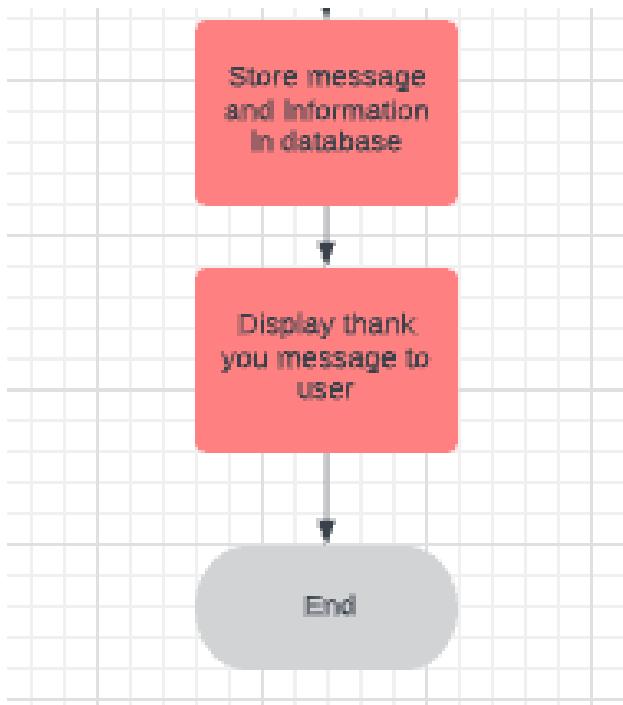
algorithm for when a user wants to favourite an item to receive reminders about. The user will first search up the desired product, either using the top navigation bar or the search function to do so. The products will then be displayed with heart icons next to them if the user wishes to favourite them. When this heart function is clicked, the system will check if the user is logged in or not, if not the system will prompt the user to log in to use this feature and the heart icon will stay unchanged. However, if the user is logged in the heart icon will change from white to red and the item will then be automatically stored in the users' reminders list, which will stay constant even if the user logs out. The algorithm then ends.

This flowchart was essential to create because it is one of the most important features on the app, even though the algorithm does not look very complicated but a lot of code will be required to

complete it. As my website is essentially for the purpose of reminders to the user this will be the most important feature and is vital to get this right.

Sending a contact message



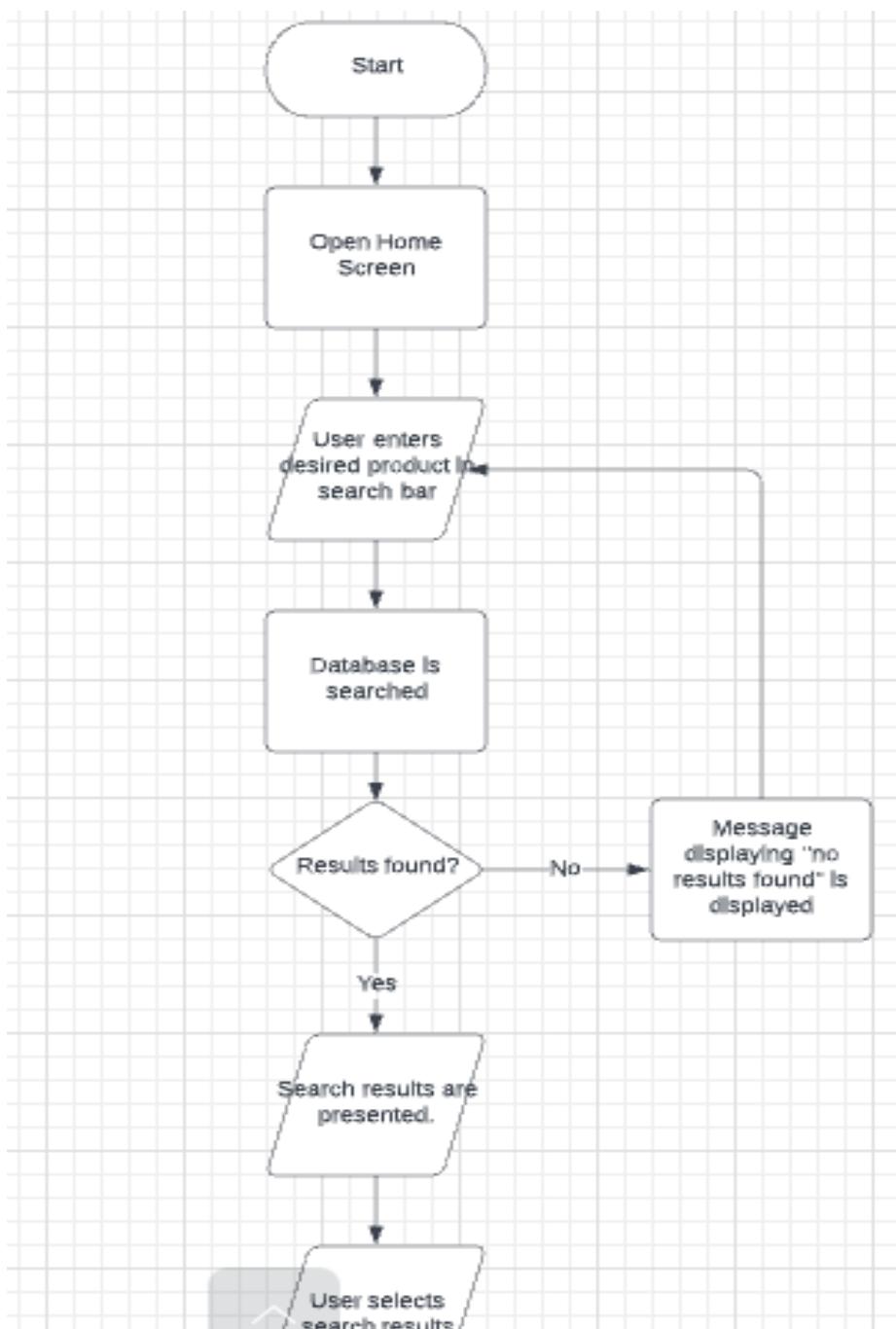


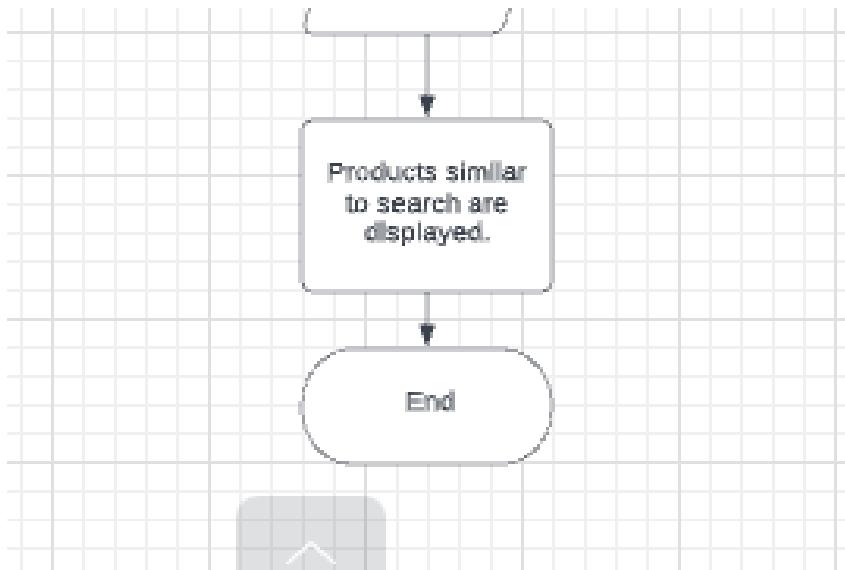
Above you can view the flowchart for when a user wants send a contact message

If a user wants to contact the website for either a feedback message or to address a problem that they had with the website, they will have to click the contact button which can be found on the top navigation bar of every page and at the footer in every page, labelled 'Contact Us.' The flowchart above shows the procedure to allow this to occur.

Upon clicking on the contact us button, the user is taken to the corresponding HTML page where they can then enter their message. The user is first required to enter all their details before entering the message. To ensure that the user will have multiple attempts to send this message there are loops back to the enter required data process if either the require data has not been fully entered or if the data is not valid. Once the users' details are valid and all fields are entered correctly, the variable is set to yes which will stop the loop from running. The message will not be successfully sent unless the email and number are valid and all details have been entered with no blanks. After that, a message is displayed ("Thank You") indicating that the message has been sent and received (stored to database for me the creator to see).

Searching Product Database





When wanting to search for an item the user can either do this with the navigation bar or using the search function. Above is the algorithm for the search function as it is the most efficient way of finding the item you desire. A feature on the home screen is the ability to search for a product using their ID or name which is represented by this flowchart. First the user searches for the ID or name in the search bar and then the database is searched to try and find a matching result to what was inputted by the user. If a result is not found, the user is informed and prompted to try again as there were no results. If the desired result is found, the user can then select the result from the presented results. When a product from the presented results is clicked, the product will display on a separate page which is shown in this flowchart.

While this flowchart is the simplest, I still felt like it was required as this will be an important feature of the app as it is a feature that will be used by many audiences and must be correct and efficient and able to adapt to the user's inputs very quickly and correctly, so there is no time wasted when the user wants to find a product quickly that is well sought after.

2.3.2 – Pseudocode

Pseudocode is very important in designing a programming project as it can give you an idea of the flow of the program and which programming constructs may need to be used without the need for exact syntax. It can help to work out what calculations and variables are required as well as mapping out procedures and functions and working out when they will be called, what they will do and what they might return.

The pseudocodes below have been separated up into different section of the overall app (This is also part of my decomposition). Each pseudocode represents one or more different screens, specifying which screen will be opened and what the app will do on that screen. Each one is clearly labelled with the overview of what that pseudocode represents so that they are easy to read and follow when implementing later on.

Logging in or Creating an Account

```

start
if user_login_button clicked = true Then
    user_login()
if user_create_account_button clicked = true Then
    user_create_account()

PROCEDURE user_login
    OPEN user_login_screen
    Valid_Login = False
    While Valid_Login == False
        input Email, Password
        OPEN user_database
        if Email == database_email AND Password == database_Password Then
            OPEN home_screen
            Valid_Login = true
        Else
            output "Incorrect log in details, please try again"

```

```

PROCEDURE user_create_account
    OPEN user_create_account_screen
    Valid_details = False
    While Valid_details == False
        input Email, Password, confirm_password
        if (Email and Password and confirm_password are valid) AND (all details are entered) Then
            OPEN home_screen
            Valid_details = true
        Else
            output "Invalid details, please try again"
    end

```

Above you can view the pseudocode for when a user wants to login

Above you can view the pseudocode for when a user wants to create an account.

When logging in or creating an account, the user must click the desired button. If they do not have an account then they are expected to click the create an account button and if they already own an account, they should click the log in button. Each of these buttons, when clicked, will trigger a procedure to run which will load the correct screen for the option selected.

The login and create an account page are similar, however they will be done on different screens so the aesthetic will look simpler for the user as there will not be too much on one page. To ensure that the user will have multiple attempts to log in to the app I have used a while loop with a valid login variable which is set to false to keep the loop running. Once the login is valid, the variable is set to true which will stop the loop from running. The login will not be successful until the email address and password both match the data from the database. It is important to use “AND” in this case

because otherwise a user could log in only knowing one of the details and this would make the app very insecure and very susceptible to hacking.

As with before, a while loop can be used to give the user multiple attempts of entering their personal details and will allow for mistakes to be made if the data is entered in an invalid way or missed out. The user can only successfully create an account if all the information required is entered as well as it being entered in a valid format. It is important that both are checked as it is vital that the information is entered in a way that it can be used and it will make sure that the user will input all their data and not miss any out.

Sending a contact message

```
start
if user contact_button = true Then
    contact_message()

PROCEDURE contact_message
    OPEN contact_message_screen
    valid_details=False
    while valid_details=False
        input name, email, number,message
        if (email and number are valid) AND (all details entered)
            output "Thank you for your message!"
            valid_details=true
            SAVE name, email, number,message to database
        Else
            output "Please enter all required fields and enter correctly!"
```

Above you can view the pseudocode for when a user wants send a contact message

If a user wants to contact the website for either a feedback message or to address a problem that they had with the website, they will have to click the contact button which can be found on the top navigation bar of every page and at the footer in every page, labelled 'Contact Us.' The pseudocode above shows the procedure to allow this to occur.

Upon clicking on the contact us button, the user is taken to the corresponding HTML page where they can then enter their message. The user is first required to enter all their details before entering the message. To ensure that the user will have multiple attempts to send this message I have used a while loop with a valid details variable which is set to false to keep the loop running. Once the users' details are valid, the variable is set to true which will stop the loop from running. The message will not be successfully sent unless the email and number are valid and all details have been entered with no blanks. After that, a message is displayed indicating that the message has been sent and received (stored to database for me the creator to see).

Favouriting Items

```

reminders_list=[]
PROCEDURE favourite_items
    OPEN user_search_results
    logged_in = False
    while logged_in==False|
        Select desired_product
            output desired_product_information
            if favourite_button clicked
                print "Please log in to favourite this item"
                user_login()
                logged_in = True
            if favourite_button clicked
                reminders_list.append(desired_product)
                print "Item favourited, check reminders list to view"
            Endif
        Endif
    END PROCEDURE

```

This diagram shows the pseudocode for favouriting a desired item and then it subsequently adding it to the users' reminders list. First the users search results are displayed and all related products to what the user inputted are displayed on the screen. Then the user will select the product he desires and when that happens, the information about the product will enlarge on screen. A flag and a while loop with a logged in variable checks if the user is logged in when he tries to favourite an item. Initially it is set to false which shows a user not logged in having just loaded into the site. When the favourite button is clicked whilst the log in is false a message is printed prompting the user to log in if he wishes to use the feature. Then the subroutine user log in is called so the user can log in. When this is completed, the flag is set to true and the loop has ended because the user no longer needs any more tries to favourite the item. If the favourite button is now clicked again the reminders list that was initialised at the start will now have the desired item that the user has favourited added to it. After, a message stating that the item has been favourited and successfully added to the list will pop up reassuring the user and that is the end of the procedure

Searching Product Database

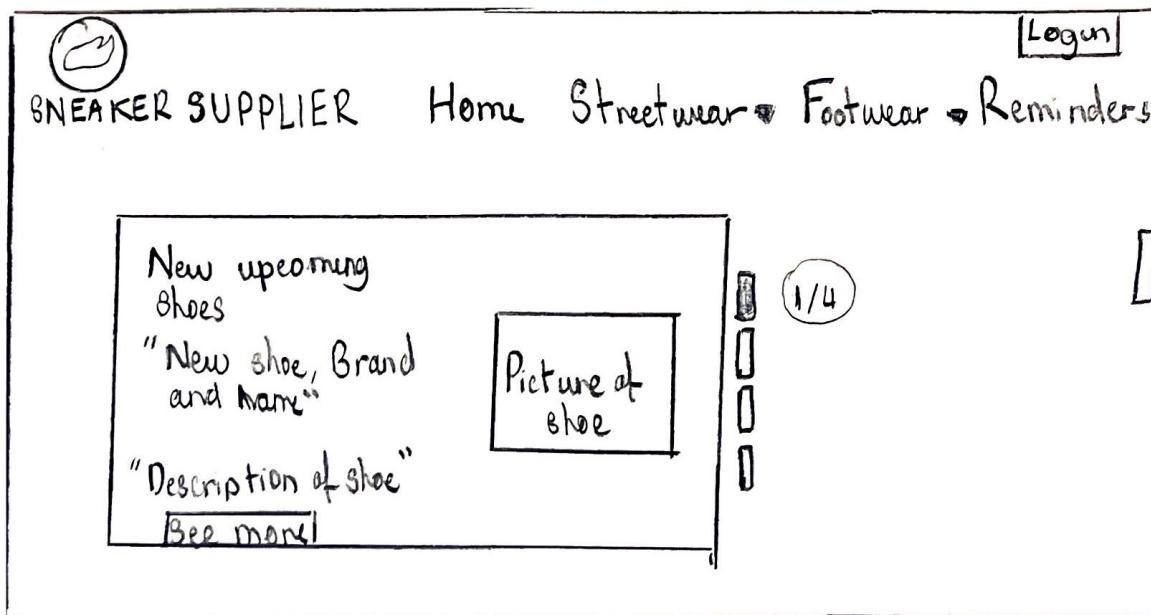
```
Function user_search_results
product_selected=False
while product_selected=False
    input user_search
    SEARCH DATABASE FOR MATCH
        if similar_matches is found Then
            output similar_matches
            if product selected from similar_matches
                output selected_product
                product_selected=True
            Endif
        Else
            output "No matches found, please try again"
```

Above shows the pseudocode for a user searching the website for a specific product if he does not want to navigate with the navigation bar. The database and product screen both need to be opened which is done at the start of this code. If the search button is clicked which would be on the home page in the navigation bar, then the function user search results is called. Once again, a loop is used so the user can input as many searches as they want until they find a match. A user will input a search and the product database is then searched for products like the input of the user. If the matches are found then the products will be displayed on the search bar. If product is selected from the search bar, then the product is displayed on a separate HTML page. The product selected flag is then set to true and the loop will finish. If there are no matches found from the database however, a message stating that no matches were found will be outputted, prompting the user to try and search again.

USER INTERFACE DESIGNS

Login/Create Account Screen

Upon loading the website for the first time, the opening home screen will be displayed. This will include a "Login" button for if the user has an account and wants to login to receive reminders on his favourite items or if the user wants to register an account as well if they do not yet have an account. The app will be used by users primarily so there will be no need for an administrator sign in. The home screen would be a grey, white and black design mainly and this theme will continue through out for the login page and register page. I think the home screen is very easy to understand with very little ambiguity, however it could be made more attractive with bright colours if requested by my end user.



This is the home page that the user will first see upon opening the website. It includes a top navigation bar for different features to navigate with ease through the site. There is also a carousel showing different products that will slide up and down automatically showing different products from the most popular products right now.

LOGIN SCREEN

If the user wishes to log after viewing home page wants to receive reminders, can click the button on top navigation and they will redirected to log in page. login screen features



in
the
and
he
the
bar
be
this
The

textboxes for the user to enter their email and password that are clearly labelled for ease of use. The large login button will

only be enabled once both an email and password have been entered. If the email address or password is incorrect when a user tries to login, a pop up will appear, and they will be prompted to try again. At the bottom of the screen there is a "Register here" link for if the user clicked login without having made an account first. There is also a remember me button that the user can click if he wants his details to be automatically entered next time that he wants to log in to the website without typing his full email and password. Lastly, the forgotten password link is there if a user has an account and has forgot the password. This link will take the user to a page where he can reset his password by using this email.

CREATE SCREEN

The create account screens are relatively basic screens featuring labelled textboxes for the user to enter various required data which will be added to the database. The submit button will be disabled until all required fields have been filled in with valid data.

PRODUCT SCREENS



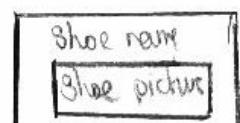
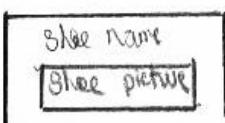
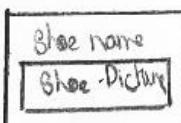
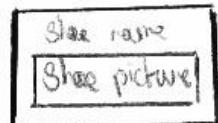
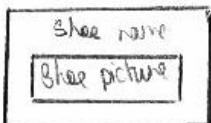
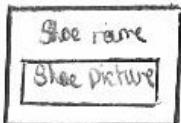
SNEAKER SUPPLIER

Home Streetwear Footwear Reminders

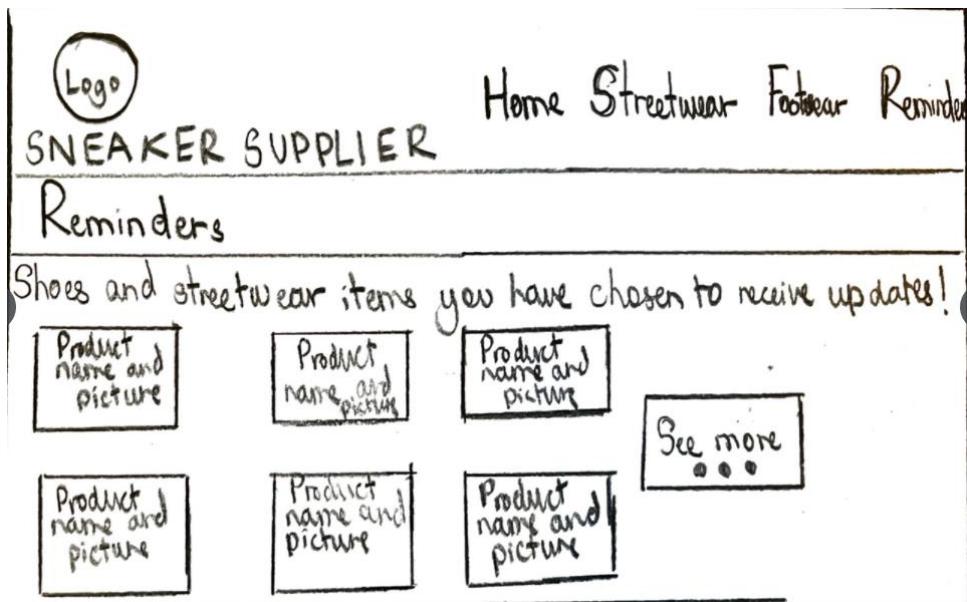
Login

Latest Sneaker Releases

Some of the hottest releases that have dropped recently!



The product screens can be accessed from using the drop-down feature on the top navbar where it says footwear and streetwear. The categories that drop down will include latest sneaker releases, upcoming sneaker releases and bestselling sneaker releases for footwear. For streetwear it will include best upcoming and latest streetwear drops. After clicking on one of these categories, the following page will look like the one shown above, which is an example of the latest sneaker releases category under the footwear category. The design for this is simple and effective, with two headers above the pictures of the products, with one stating the drop-down category name and the other with a description. The top navigation bar is still shown on this page as it is on all the other pages to allow easy access across the website for the user. Pictures of the products are on every page and will be designed out seen above, with the boxes including the shoe name and the picture of the shoe. There will be maximum 6 boxes to keep every page concise. When a product box is clicked in, it will show its information and also have an option to favourite the item so it can be added to a reminders list and also it will show the shoes rating which will be based on its popularity and its resale value. All product pages will be designed in this manner.



After the user favourites his desired items from the product list, the products chosen will be added to this page and information on each shoe that they require will show when they click the shoe as well. There is also a see more slider if the user has more than 6 items listed on the page, and will be able to slide to the next list of items on the slider. These will be the items that the user will receive updates and notifications on. Only if an item has been favoured and added to the reminders list can he receive these notifications. The page will be designed similarly to all the product pages, with two headings and the top navbar for easy navigation through the app. The colours will be consistent to the colours as mentioned before, black grey and white. This page will only load up if the user is signed in.

Contact Now

I will also implement a contact page if the user wishes to contact me if they have any issues with the message, or if they just want to send a feedback message with how good their experience was on the website. This contact section will be accessible from the footer and it includes four text boxes for the name of the user, phone number of the user, email of the user and finally the message that they want to send. When they wish to send the message, they can click submit and a pop up saying "message has been sent" should display to the user whilst the message stores and sends to me, either on my email or stored in a text file database.

Footer


SNEAKER SUPPLIER

[F](#) [D](#) [G](#)

Phone no.  dogunrin@outlook.com

[Products](#)

[New products](#)
[Contact us](#)
[Home](#)
[Best selling](#)

[Address](#)

[Address](#)
[HOME](#)

[Our company](#)

[Contact us](#)
[About us](#)

[Newsletter](#)

[Subscribe](#)

2022 All rights Reserved. The Sneaker Supplier

This is the footer that will be implemented at the bottom of every single page including the home screen. It will be formatted like seen above with the sneaker supplier logo in the top left corner, the social media handles that were made for the page and also contact information for the company, such as the phone number and the email. It will also include a newsletter option where the user can enter an email and click the button subscribe if they want to receive updates and more from the company. The footer will also include an about us link, information about the founder address and also link to product pages on the website. The design for this will coincide with the aesthetic of the whole website.

Improved Interface Design

After speaking to my client and both end users, I have made some improvements to my interface design based on their feedback. For example, many of the screens lacked the menu button which should remain consistent throughout every screen on the app for navigation purposes. It also lacked a settings button so the user can log in, log out. In all the improvements shown below, an image is shown if big changes have been made to the system and just a short succinct description if only a little change has been made to the design of the system. This is to allow for an easier comparison and to also manage time.

Sign up

Enter email:

Enter Password:

Confirm Password:

Password must contain ~~one~~ special character, a number and a capital letter. It must be greater than 8 characters

Already have an account? [Login here](#)

The register page was redesigned in a big way. Firstly, my end user recommended that not the same colours should always be used in the entirety of the project so the register page will be a different colour to the log in page. The sneaker supplier logo was also taken out to make the page look more minimalistic and simplistic so the user doesn't feel overwhelmed when wanting to sign up to the system. The confirm email text box and line was deleted as it was unnecessary. The big bold register at the top was changed to sign up and there is now a helpful list of requirements for the user password to ensure the user meets them and has a secure password that is not easy to hack, adding to the security of the website. Finally, a log in button will be displayed and the usual link to the log in page was kept but is now put in the right corner

Home Page

Changes were made to the home page as requested by the end user as he stated that more vibrant colours were wanted to further the aesthetic of the website and reel the user in, whilst also making the website look more exciting and fun. The consistent theme of grey white and black was maintained in some places like the log in page but for the home screen and other product pages, the colour scheme was changed to pink black and green, giving a more vibrant look to the system. Also, a contact option will also be added to the top navigation bar in all pages of the website to allow easier navigation.

Settings

A settings pop-up screen was also added when the user clicks log in button on the home page. If the user is logged in already, there will now be a sign out option for the user and also a forgotten password link if the user wishes to change their password for any reason.

All previous designs that have not been mentioned in this section will be left unchanged

2.4.3 - Interface Analysis

After the improvements of my interface, I asked for feedback from the client and end users again. The end users said that they thought that the app looked very easy to navigate and the layout was very clear and easy to use, for example the use of many labelled buttons meant that it was clear what every button would do. They also appreciated the fact that every screen had a navigation bar at the top and a means to go back to the page that they were just currently on by clicking this in the top navigation bar. They said this would vastly improve the usability and allow for them making mistakes. It has been suggested that the colour scheme be made of light blue and dark pink with perhaps pops of green and also black grey and white in other areas. This is all to make the website more appealing to the user and they have agreed that those colours will be best fitting giving that the website serves an exciting purpose. The client has said that they are happy with the design of the website as it looks very easy for them, as a reseller, to be able to view the information of products and see where they drop and navigate through the app easily. This is especially important for them if they are under pressure and need an efficient way or getting to a product that is about to drop or already dropped that they want to attain. They appreciated the idea of the carousel and so much information on the home page giving the app a “no time wasting” feel toward it as it immediately tells the user about popular and latest drops on the home page which is important especially to him as a reseller. They also appreciated and the idea the section on the sign-up page where you must re-enter your password and meet the password requirements as this improves the security of the app and helps to ensure that users data is secure.

Validation

Validation is used to check that the data entered in an app or program is valid, sensible and reasonable. It does not check whether the information entered is correct, but it limits the ways in which it can be incorrect, for example, validation cannot check you have entered your date of birth correctly, but it can check you have entered a valid date as your date of birth.

Validation will be very important in my site as it will ensure that the website does not crash or continue if it is not meant to if unexpected data is entered at any point and will help the user use the website in a more sensible way and will help them identify inputs that may not

be reasonable. In the table below, I have outlined the different features of the website that will require validation and what validation they require with a justification as to why this is needed. I have also outlined how I can test that the validation is working as expected which will help when creating my test plan by allowing me to choose relevant test data.

| Validation No. | Feature | Validation required | Justification |
|-----------------------|---------------------|--|--|
| V1 | Login | User should only be allowed to log in if all fields are entered and all required standards are met | This is to ensure the security of the users' details and also to not allow the user to log in with missing details. |
| V2 | Login | User should only be allowed to login with a correct email address and password | This is to ensure that the login function works correctly, and the user can access the app and their data |
| V3 | Login | User should be able to retry if login failed, the app should not crash | This is because someone might have made a mistake while entering their email and password |
| V4 | Contact Message | User should only be able to send message when all required fields have been filled | To ensure that all required data from the message for the database is there and to ensure no blank and meaningless messages are sent |
| V5 | Contact Message | User should meet required field format | For example, the number should be 11 digits and the email should be valid with the @ symbol in between the email provider and email ID |
| V6 | Display products | The app should ensure that the user can view the information of one product at a time | This is because a the app would likely crash if this was not the case. |
| V7 | Searching for items | App should not crash if no results are found when searching for an item | This is because the user could search for a product that doesn't exist or enter an input such as numbers |

| | | | |
|-----|------------------|---|--|
| | | | which would likely not return a search result |
| V8 | Creating account | User must enter info in the required fields and must be in the correct form | This is to ensure that all relevant data needed for the users account is entered and is entered in the correct form and is as sensible as possible |
| V9 | Creating account | User cannot use an email that has already been used | This is so that every user has a unique email |
| V10 | Creating account | User's password must be at least 6 characters | This checks that their password is more secure and make sure it is entered as they meant to. |

2.6 – Success Criteria

| | | |
|-----|--|--|
| SC1 | User should be able to create an account and should be prompted to fill in all the required data in a valid format | If the user does not create an account they cannot use all the app features and if they do not fill in all the details properly and validly, the app may not be as effective |
| SC2 | When the user creates an account, the data should be stored in the correct database | This is so that the information is stored on the app and can be used/changed/deleted and the user will be able to log on to the app |

| | | |
|------|---|--|
| SC3 | User should be able to successfully login with their correct username and password and allow multiple attempts to login | This is to check that the user can login correctly with their username and password with both being correct to allow the user to be logged in. |
| SC4 | User should be allowed to enter a contact message to the company securely | This is to leave a feedback feature so the user can express his feelings to the website and also ensure security |
| SC5 | The app should save the inputted data to a database of messages. The message should be saved with the users email and name to show which users message reading it is | This is so the owner of the website can see messages and also to ensure the data protection and privacy of the users' information. |
| SC6 | The app should have many visual icons in the menu | This will make the app accessible to everyone. |
| SC7 | The app home page should be able to display all most important information and products on that day | To ensure efficiency |
| SC8 | The app should have a top navigation bar and every time the user presses on one of the features: Contact, Reminders, Footwear categories, streetwear Categories and Home to allow easy navigation through app | This ensures easy navigation and also allows the user to return to any page that they came from such as the home page as this top navigation will stay constant across all pages. |
| SC9 | Links to page and displays page when the user clicks the see more button on a product | This will determine if the app can successfully display the data and page that is linked from the home page, for example the app can show the best-selling sneaker on the home page and if the user clicks see more he is taken to the page of best-selling sneakers |
| SC10 | User should be able to receive reminders and notifications after favouriting a desired item, either on the home page or to their email (if possible) | Will determine if the system can successfully send a reminder at the right time to the user |

| | | |
|------|---|---|
| SC11 | User should be able to favourite items if they are logged in , and favourite multiple items if they want to and all of these items should be added to the reminders list | Will determine the functionality of the heart icon if that desired product is then sent to a personalised reminders list for the user |
| SC12 | User should be able to search the product database and most relevant products matching the search appear | This is to make the searching function more effective and ensuring that the most relevant searches appear first on the list making it quicker for the user. |
| SC13 | If there is no search result for a product item, the user should be prompted to try the search again and should be informed that there are no matching results | This is because the item may not be in the database and the user will need to be informed of this and the app needs to not crash if a search is not found |
| SC14 | The user should be able to access the settings menu and change any of the features they wish to regarding their personal details and these will be edited in the database | This is because the user may wish to change their password for security reasons |
| SC15 | The user should be able to log out of the app with all data remaining saved in the database | This is important as the user needs to be able to log out and back in with no data being lost or changed |
| SC16 | User should be able to access links and information to each product so they can purchase and see release information | Important to the purpose of the website, so the user can view release dates and click links to purchase these items. |

2.7 - Test Plan

2.7.1 – White Box Testing

White box testing is a form of testing where the internal implementation of the program is known to the tester. This is important in terms of testing if the app works as expected since the person testing has complete knowledge of how the code should run and what each part of the program should do. They will also know which inputs and test data should be allowed and what should not be allowed which is very important in testing the success of the validation. To test the app using this method, I have created a test plan which I will follow to ensure that all sections of the app perform as expected and

2.7.2 – Test Table

| Test no. | What will it test | Test information | How to test | Expected outcome |
|----------|-------------------|---|---|--|
| T1 | SC1, V8 | This test will be to ensure that when a user creates an account, they have entered all the data required in a valid format by trying to enter invalid and missing data. It will do this by checking for empty input boxes and using validation. | Enter all required information in the valid Format to test for creating an account and to test for unfilled data validation, leave information out of the boxes and press submit. | The app should create the account or app should present a pop up saying information is missing and allow the user to try again |
| T2 | SC1, V9 | This test will be to ensure that the user has entered an email that is unique and has not already been used by another user. It will do this by searching the database for the inputted username. | Enter email that has already been used | Website should prompt the user to use another email or log in with existing email |
| T3 | SC1, V10 | This test will be used to check that the user has entered a password and that it meets the criteria specified in the validation table. This is used to make the password more secure | Enter a password that doesn't fit the criteria | Will prompt user the criteria for a valid password and do not allow user to enter the Second password entry or create an |

| | | | | |
|-----------|------------------|--|---|---|
| | | and to check the user enters the password how they think they have. | | account until first password entry is valid |
| <u>T4</u> | V4,V5,SC4,SC5 | This will test that when a user enters a contact message and clicks send that the information is stored in a database. Will also test that if the user doesn't enter all the required information, they will be prompted to try again | Enter all required information in valid format and click send button, or miss out a required field and click send | Will save in database and display message to user saying thank you, or it will prompt user to fill out all fields in correct format |
| <u>T5</u> | SC8, SC9, V6 | Allow user to navigate through the website and when a feature is clicked either on the top nav bar or the footer or anywhere on the home page (see more), it will take user straight to the corresponding HTML page and display the products | Click a feature such as best-selling sneakers | Will load the page display products and information on that page |
| <u>T6</u> | SC12, SC13,V6,V7 | This will test that when a user enters a product to search and the most relevant products matching the search appear, or if there's no similar searches then no matches will be found | Enter desired product and enter | Will display desired products if matches found and if not will prompt user to search again |
| <u>T7</u> | SC11 | This will test that the favouriting items feature works on all items if the user is logged in and if not will prompt user to log in to use this feature | Search desired product and click on the heart icon next to it | Will change to red if the user is logged in, if not the user is reminded to log in to use the feature |
| <u>T8</u> | SC11 | Will test that all favourited items are added to a reminders list personalised to each user | Check reminders list after favouriting an item | Items will load onto this list |
| <u>T9</u> | SC10 | Will test that a user will get a pop up notification when an item releases | Wait for shoe release time to see if notifications come up | Pop up on home screen or to email will load when item is released |

| | | | | |
|------------|--------------|--|---|---|
| <u>T10</u> | SC7,SC6 | This will test that the carousel feature works on the home page and displays all the hottest releases right now with a range of footwear and streetwear | Load up the home page and the carousel will be displayed and should automatically move up and down, or will move to place clicked by user | Carousel will move as desired and display most popular products with links to their pages |
| <u>T11</u> | SC7 | Will test that the hottest products are all displayed on home page to user so there's no time wasted | Load up home screen and scroll through | All products with their pictures should be displayed |
| <u>T12</u> | SC3,V3,V2,V1 | This will test that when a user enters his details to log in, the details are compared to database and if matches user will be allowed to log in | Enter correct details as one you signed up with and press log in | Will log in successfully or if not matching will say incorrect details try again |
| <u>T13</u> | SC1,SC2,SC3 | Will test that user can link back to create account screen from log in page (or vice versa) if they don't have an account or already have an account. | Press don't already have an account link on log in page or the other way round from the create account screen to log in page | Log in or create account screen will be loaded |
| <u>T14</u> | SC14 | Will test those settings feature works and allow user to change his password or log out | Click on settings icon and press log out or change password | User will be logged out and loaded back into home page or will be taken to screen where password can be changed |
| <u>T15</u> | SC15 | This will test that the app will successfully log out while keeping all the data saved properly in the database | Click the log out button and check the database | The app should return to the login/create account screen and the database should store all the same information it had before the user logged out |
| <u>T16</u> | | Will test that the website will have useful links to specific websites to where you can buy the shoe. Different links for each unique product. Also upon clicking the product release dates and info | Click the desired product. | Upon clicking a product, a pop up should display where the user can click links for the product to go and purchase them |

| | | | | |
|--|--|-------------------|--|--|
| | | will display too. | | |
|--|--|-------------------|--|--|

2.7.3 – Black Box Testing

Black box testing is a method of testing a program with no knowledge of the implementation and programming behind it. This is a very important method of testing since someone without knowledge of the program will not have full knowledge of how it works and will therefore use it in the way they sit fit rather than the way the developer is assuming it will be used. This is very useful for identifying errors in validation since the user may use the app in unexpected ways which can then identify places the app may crash or react unexpectedly.

2.7.4 – Usability Testing

It is important to test the usability of the app with real users testing the app. The white box tests are important to test the functionality of the app and make sure it runs as expected and saves this correctly etc. but it is also important to test how usable the app is and ensure that it can easily be used by someone with no knowledge of the process. It is also important to test that the user will be able to easily work out how to use the app and make the most out of the features of the app. It will also be used to test that the users are happy with the layout of the app as this can be a big deterrent to people. To test this, I will create a questionnaire based on the features of the app and how easy they are to use as well as how useful and helpful they were. This will help with the evaluation section since I will be able to get feedback from multiple people with different opinions and different outlooks in order to get a more useful evaluation.

2.8 – Client Sign-off

I, as the client, am happy with the design.

Name: Richard Chinedu

A handwritten signature in black ink, appearing to read "Richard Chinedu".

Signed: _____

Date: 10/12/2021

IMPLEMENTATION

I will now begin the development section of this project. In this section I will code my solution, testing functionality as I go through. Failed tests, requiring remedial actions, will be shown in red.

The way in which I will progress through this development will be to implement sections in parts, with the help of the abstraction I did in my design and analysis sections. I have a large number of individual sections that I will be required to implement, testing after each section. I will also aim to do regular reviews of the sections programmed up to a certain point, to ensure they are all functioning optimally.

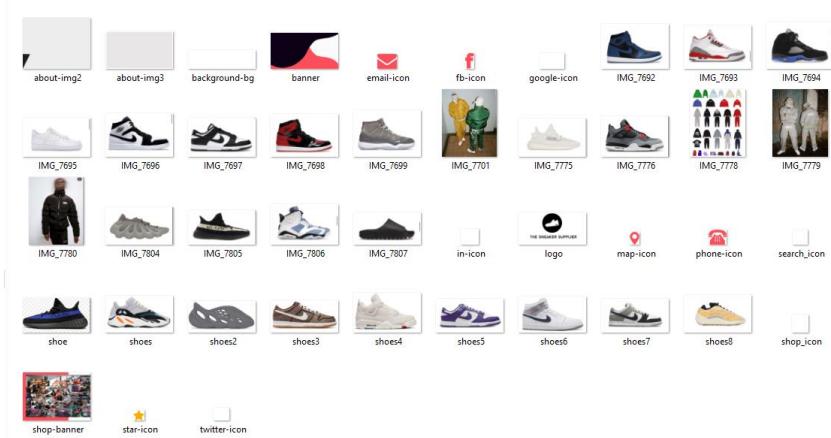
Creating the assets

Before starting my programming, I had to create a range of images for use in the program. For example, for my logo, all the pictures of my products, and for icons. After I created these images and saved them into their section of files called images within my whole project file where I can easily reference them in my HTML code pages.

I started off coding in normal text files but after I gained access to a device that could download visual studio as my platform for coding, this is what I used. Due to this, there might be a mixture of code in a text file and code in visual studio in my screenshots.

| | | | | |
|--|---------------------|-------|---------------|---------------------|
| | css | -- | Folder | Apr 17, 2022, 19:24 |
| | images | -- | Folder | Apr 14, 2022, 14:40 |
| | js | -- | Folder | Apr 17, 2022, 20:42 |
| | about.html | 4 KB | HTML document | Mar 25, 2022, 12:57 |
| | bestsellingsneak... | 16 KB | HTML document | Apr 17, 2022, 20:16 |
| | contact.html | 10 KB | HTML document | Apr 16, 2022, 23:55 |

These are some of my separate files created for each part of my project and have folders inside of my folder for separate languages such as CSS folder, JavaScript folder and my images folder too. Below are the images inside my folder which contain different things such as icons, banners and shoe photos



I started by designing the initial screens where the user can login or create an account. I decided to use two separate pages for the register and the login page. Originally in my design, I decided to do the login and register page on the same document and have the login page and register page differentiated by a slider. In the end however, I coded up the login page and register page separately and put a reference link to each page. If the user had already registered, there would be a link to the log in page on the register page. If the user was on the log in page and had not created an account, there was a link on the log in page to the register page. I also wanted to create a logo and implement it at the top of the page. I had already predesigned my logo using CANVA and I created it in accordance with my colour theme for added coherence. However, when creating this layout, I decided that it was not very appealing as there were too many features on the screen at once. Because of this, I decided to use only one login and create account button and the separation will occur on the next screen. I also changed the colours of the buttons and the text in order to make it

more coherent with the logo and have the app have a very distinct colour theme. I also added in a toolbar with the title of the app because a toolbar will be a feature in all other screens and it will help keep the theme of the app consistent.

To create the layouts of the screens I used CSS and HTML to design the colours on the screen. After designing my logo, I linked it on my html page and designed its positioning on the log in page and also on all my home pages to keep the consistency. Initially when first creating my pages I used a regular text file because I wasn't able to download visual studio on my laptop device.



THE SNEAKER SUPPLIER

This is the logo that was predesigned with the app CANVA, which I then downloaded as a file so I could link it in html. After doing this, I designed my login page and below is the first prototype, where I put the logo above.

The screenshot shows a dark grey login interface. At the top center, there is a large, empty square placeholder for a logo. Below this, the word "Login" is centered in a white font. Underneath "Login", there are two input fields: "Email address:" followed by a text input containing "logumin@outlook.com" and "Password:" followed by a text input containing "*****". To the right of the password field is a checkbox labeled "Remember me". Below these fields are two links: "Forgot password ?" and "Login". At the bottom left is a "Go Home" link.

This is the design of the first page; however, the logo was failing to implement and a link to the register page was not working, but the link to the home page was. I also didn't like the CSS design that well so I decided to change it up a bit, while adding my validation and attempting to implement my database. The database was not working properly however I succeeded in creating my home page design how I wanted, and it successfully

linking to my home and register pages with all the validation working perfectly. All the code will be found below

```
<!--end of preloading-->
<!--login form popup-->
<div class="login-wrapper" id="login-content">
  <div class="login-content">
    <h3>Login</h3>
    <form method="post" action="login.php">
      <div class="row">
        <label for="Email address">
          Email address:
        </label>
        <input type="text" name="Email address" id="Email address" placeholder="dogunrin@outlook.com" pattern="^([a-zA-Z][a-zA-Z0-9-_\.]{8,20})$"/>
      </div>

      <div class="row">
        <label for="password">
          Password:
        </label>
        <input type="password" name="password" id="password" placeholder="*****" pattern="^(?=.*[0-9])(?=.*[A-Z])(?=.*[a-z])(?=.*[\W])((?=.*[.\\n]).{8,})$"/>
      </div>

      <div class="row">
        <div class="remember">
          <div>
            <input type="checkbox" name="remember" value="Remember me"><span>Remember me</span>
          </div>
          <a href="#">Forget password ?</a>
        </div>
      </div>

      <div class="row">
        <button type="submit">Login</button>
      </div>
    </form>
  </div>
</div>
```

This is the coded HTML code in my text file for the screenshot of my login page shown above. I tried to use PHP and SQL to link my database so when the user logs in or registers it is saved to a database and every time the user logs in the email and password combination is checked against the ones already in the database, showing that the user exists. If the password combination is wrong, it should output wrong password and if the combination is correct it should be successfully logged into the system.

```

@charset "utf-8";
h1
{
    margin: 0px;
    padding: 5px;
    background-color:grey;
    font-family: Arial, Helvetica, "sans-serif";
    font-size: 36px;
    text-align: center;
}

h2
{
    margin: 0px;
    padding: 5px;
    background-color:grey;
    font-family: Arial, Helvetica, "sans-serif";
    font-size: 30px;
    text-align: center;
}

h3
{
    margin: 0px;
    padding: 5px;
    background-color:grey;
    font-family: Arial, Helvetica, "sans-serif";
    font-size: 30px;
    text-align: center;
}

body
{
    margin: 0px;
    padding: 0px;
    background-color:grey;
    font-family: Arial, Helvetica, "sans-serif";
    font-size: 18px;
    text-align: center;
}

#page
{
max-width:800px;
margin: 20px auto;
padding: 30px;
background-color: #bfa475;
}

#header /* Styles for Heading */
{
padding: 5px;
background-color: #655035;
text-align: center;
}

img
{
border: double 10px white;
}
<

```

This is the initial CSS coded up for the login systems saved as styles.css. I changed it up a bit but most things stayed the same when trying to perfect the aesthetic that I want for my website.

Second Log in Version



THE SNEAKER SUPPLIER

Login

Email address:

Password:

Remember me

[Forget password ?](#)

[Login](#)

[Dont have an account? Register here](#)

[Go Home](#)

After I edited the CSS from fully grey to a mixture of grey black and white to suit the aesthetic that I want my login and register pages to look like, I also tried again to insert my logo above in the login page and it worked successfully, see below to view the changes made to the HTML and CSS file code

CSS style sheets always referenced to at the start of HTML code

```
<!DOCTYPE html>

<head>
<link href="css/styles.css" rel="stylesheet" type="text/css">

<title>The Sneaker Supplier</title>
<meta charset="UTF-8">
<meta name="description" content="">
<meta name="keywords" content="">
<meta name="author" content="">
<link rel="profile" href="#">

<link rel="stylesheet" href="http://fonts.googleapis.com/css?family=Dosis:400,700,500|Nunito:300,400,600" />
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="format-detection" content="telephone-no">

<link rel="stylesheet" href="css/plugins.css">
<link rel="stylesheet" href="css/style.css">

</head>
<body>

<div id="preloader">

<div id="status">
    <span></span>
    <span></span>
</div>
</div>
<!---end of preloading-->
<!--login form popup-->
<div class="login-wrapper" id="login-content">
<div class="login-content">
    <h3>Login</h3>
    <form method="post" action="login.php">
        <div class="row">
            <label for="Email address">
                Email address:
                <br>
                Email address:
                <input type="text" name="Email address" id="Email address" placeholder="dogunrin@outlook.com" pattern="^([a-zA-Z][a-zA-Z0-9-_\\.]{8,20})$" required="required" />
            </label>
        </div>
        <div class="row">
            <label for="password">
                Password:
                <input type="password" name="password" id="password" placeholder="*****" pattern="^(?=.{8,})(?=.*[0-9])(?=.*[a-zA-Z])(?=.*[!@#$%^&*])" required="required" />
            </label>
        </div>
        <div class="row">
            <div class="remember">
                <div>
                    <input type="checkbox" name="remember" value="Remember me"><span>Remember me</span>
                </div>
                <a href="#">Forget password ?</a>
            </div>
        </div>
        <div class="row">
            <button type="submit">Login</button>
        </div>
    </form>
</div>
</div>
<!---end of login form popup-->

<div class="page-single-2">
<div class="container">
    <div class="row">
        <div class="middle-content">
            <p> Dont have an account? <a href="register.html">Register here</a> </p>
            <a href="home.html">Go Home</a>
        </div>
    </div>
</div>
<script src="js/iuerv.js"></script>

<!---end of login form popup-->

<div class="page-single-2">
<div class="container">
    <div class="row">
        <div class="middle-content">
            <p> Dont have an account? <a href="register.html">Register here</a> </p>
            <a href="home.html">Go Home</a>
        </div>
    </div>
</div>
<script src="js/jquery.js"></script>
<script src="js/plugins.js"></script>
<script src="js/plugins2.js"></script>
<script src="js/custom.js"></script>
</body>

</html>
```

Commentary on the HTML file will be indicated like this <!-->

New feature added, link to register page that works

JavaScript files linked at end still inside body tag, all files imported so no need to display

This log in page also includes the email and password validation, and it works successfully if the user tries to click the log in button without entering anything in the email address and password field due to them being 'required', as seen in the code. However, when entering my email address with the correct format, it prompted me to "please match the correct format". To fix this mistake, I changed the pattern line <input type="text" name="Email address" id="Email address" placeholder="dogunrin@outlook.com" pattern="^[a-zA-Z][a-zA-Z0-9-_\.]{8,20}\$" required="required" /> and corrected it to pattern="/^a-zA-Z0-9.!#\$%&!*+/=?^_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)*\$/i" required /> which incorporates the @ symbol in a specific place, which is the format for an email.



Login

Email:

Password ! Please fill out this field.

[Forgot password ?](#)

[Login](#)

Dont have an account? [Register here](#)

SUCCESS

This shows a successful validation if the user does not input something in the email field.

SUCCESSFUL
VALIDATION

Login

Email:

Password:

Remember Please fill in this field.

[Forget password ?](#)

[Log In](#)

Dont have an account? [Register here](#)

[Go Home](#)

This shows a successful validation if the user does not input something in the password field and tries to log in.

CSS Files for the design:

```
}
```

```
body {  
    color: #666666;  
    font-size: 14px;  
    font-family: Poppins;  
    line-height: 1.80857;  
    font-weight: normal;  
    overflow-x: hidden;  
}
```

Selector specifying which HTML elements are being designed right now. Will be a constant in this project due to a load of HTML so there will be a lot of CSS. HTML elements will need to be designed and this is how Java will specify them. CSS selectors can be selected based on class, id or name (simple selectors) combinator selectors, pseudo-class selectors and attribute selectors. Most of these selector categories will be used in my CSS in this project

```

h1,
h2,
h3,
h4,
h5,
h6 {
    letter-spacing: 0;
    font-weight: normal;
    position: relative;
    padding: 0 0 10px 0;
    font-weight: normal;
    line-height: normal;
    color: #111111;
    margin: 0
}

h1 {
    font-size: 24px
}

h2 {
    font-size: 22px
}

h3 {
    font-size: 18px
}

h4 {
    font-size: 16px
}

h5 {
    font-size: 14px
}

h6 {
    font-size: 13px
}

h1 a,
h2 a,
h3 a,
h4 a,
h5 a,
h6 a {
    color: #212121;
    text-decoration: none !important;
    opacity: 1
}

```

Using a new file called style.css in conjunction with my old CSS file for my first draft of my home page. This CSS is also used in the main pages of my code and this is all to keep the consistency throughout my system.

The links shown above link to my home page and register page each successfully,

Sign Up

Please fill this form to create an account.

Email Address

Password

Confirm Password

Already have an account? [Login here](#)

This screenshot above shows my signing up page which can be referenced to by the user if he opens the log in page and wants to log in but does not have an account yet. They can click the link and they would be redirected to this page where they can register an account which would then allow them to log in. If the user finds themselves on the register page but already has an account there is also a link to the log in page there as well. When a user signs up, his details are saved to a database which are then compared to when the user logs in. If the credentials do not match the user cannot log in and if the users' email is not in the database, it will also not log in and will require the user to go to sign up. The password hash is compared to the hashed password in the system and if they are the same and the email address is linked and is the same as well, the user can log in. If the users' email is entered correctly but they have forgotten their password, the user can click forgotten password, which would allow them to reset their password. However in this first version of the log in and sign up pages the database and the reset password feature had not being fully implemented yet.

```

<html>
    <head>
        <meta charset="UTF-8">
        <title>Sign Up</title>
        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
        <style>
            body{ font: 14px sans-serif; }
            .wrapper{ width: 360px; padding: 20px; }
        </style>
    </head>
    <body>
        <div class="login-wrapper">
            <h2>Sign Up</h2>
            <p>Please fill this form to create an account.</p>
            <form action="register.php">
                <div class="form-group">
                    <label>Email Address</label>

                    <input type="text" name="EmailAddress" placeholder="dogunrin@outlook.com" class="form-control" <?php echo (!empty($EmailAddress_err)) ? 'is-invalid' : ''; ?>

                    <span class="invalid-feedback"> </span>
                </div>
                <div class="form-group">
                    <label>Password</label>
                    <input type="password" name="password" placeholder="*****" class="form-control" <?php echo (!empty($password_err)) ? 'is-invalid' : ''; ?>
                    <span class="invalid-feedback"> </span>
                </div>
                <div class="form-group">
                    <label>Confirm Password</label>
                    <input type="password" name="confirm_password" placeholder="*****" class="form-control" <?php echo (!empty($confirm_password_err)) ? 'is-invalid' : ''; ?>
                    <span class="invalid-feedback"></span>
                </div>
                <div class="form-group">
                    <input type="submit" class="btn btn-primary" value="Submit">
                </div>
                <p>Already have an account? <a href="login.html">Login here</a> </p>
            </form>
        </div>
    </body>
</html>

```

Tried to implement php validation and SQL database, did not work, also some mistakes in code. Test: **FAIL**

Second Version (Sign Up Page)

After speaking to my end user, I was made to reconsider my design to make it look more aesthetically pleasing and also more basic, as the user didn't like the ideas of the placeholders and he also said that the confirm password section was not necessary as it was just excessive code. He also reminded me to add in the password requirements under the text boxes as a reminder to the user what kinds of passwords will be accepted. This was also discussed under my design section. Thus, I added this feature and I changed some colours on the register page, as well as editing the placeholders and this is how it looks below:

Sign Up

Please fill in this form to create an account.

Email

Enter Email

Password

Password

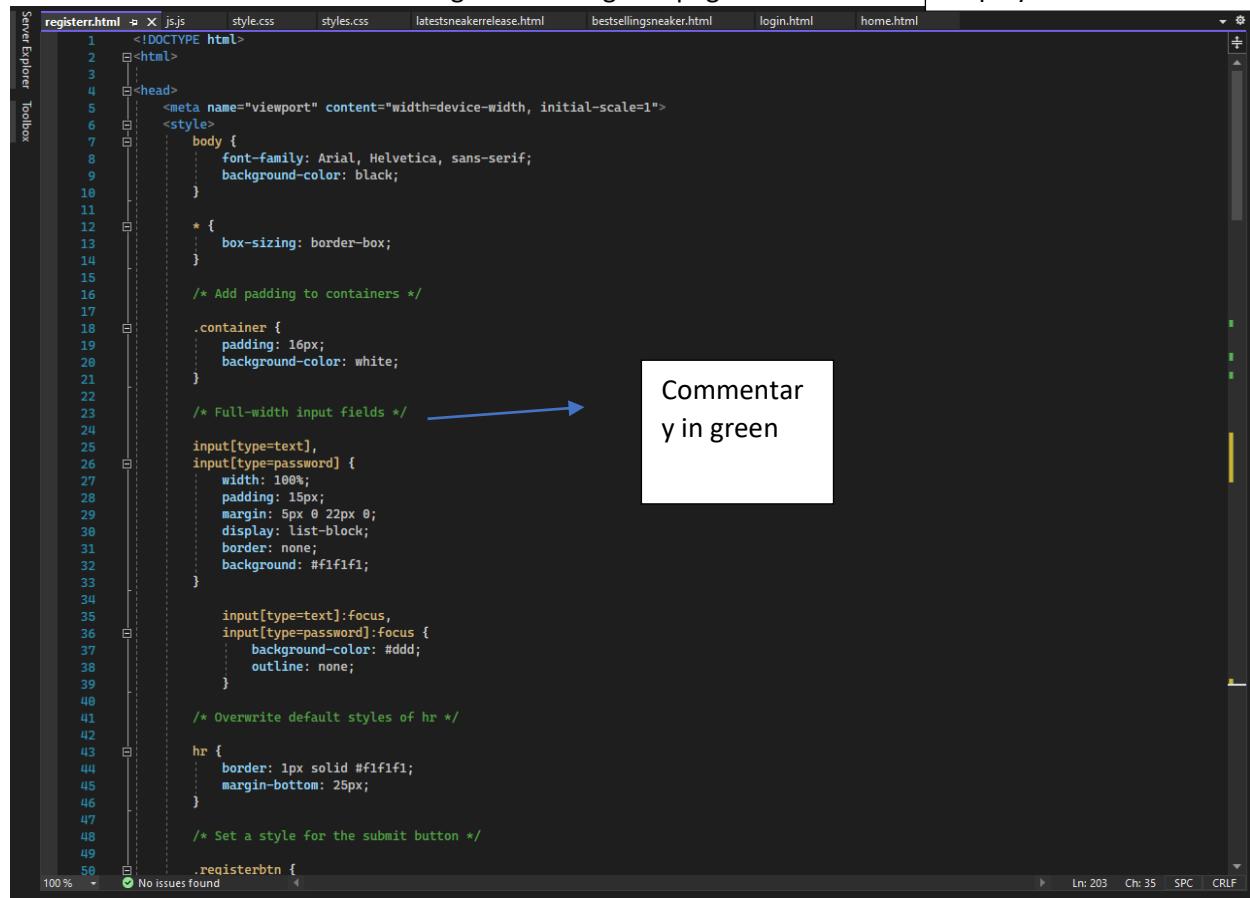
Password should be at least 6 characters

Register

Already have an account? [Log in here.](#)

This is the new version code and design for this register page:

Validation/security. Ensuring user picks long enough password so less likelihood for hacking and if requirement not met, an error message will be displayed



```
register.html  X js.js    style.css  styles.css  latestsneakerrelease.html  bestsellingsneaker.html  login.html  home.html
```

```
register.css
1  <!DOCTYPE html>
2  <html>
3  |
4  |<head>
5  |  <meta name="viewport" content="width=device-width, initial-scale=1">
6  |<style>
7  |  body {
8  |    font-family: Arial, Helvetica, sans-serif;
9  |    background-color: black;
10 |  }
11 |
12 |  * {
13 |    box-sizing: border-box;
14 |  }
15 |
16 |  /* Add padding to containers */
17 |
18 |  .container {
19 |    padding: 16px;
20 |    background-color: white;
21 |  }
22 |
23 |  /* Full-width input fields */
24 |
25 |  input[type=text],
26 |  input[type=password] {
27 |    width: 100%;
28 |    padding: 15px;
29 |    margin: 5px 0 22px 0;
30 |    display: list-block;
31 |    border: none;
32 |    background: #f1f1f1;
33 |  }
34 |
35 |  input[type=text]:focus,
36 |  input[type=password]:focus {
37 |    background-color: #ddd;
38 |    outline: none;
39 |  }
40 |
41 |  /* Overwrite default styles of hr */
42 |
43 |  hr {
44 |    border: 1px solid #f1f1f1;
45 |    margin-bottom: 25px;
46 |  }
47 |
48 |  /* Set a style for the submit button */
49 |
50 |  .registerbtn {
```

Commentary in green

```

49     .registerbtn {
50         border-color: #04AA6D;
51         color: white;
52         padding: 10px 20px;
53         margin: 8px 0;
54         border: none;
55         cursor: pointer;
56         width: 50%;
57         opacity: 0.9;
58     }
59
60     .registerbtn:hover {
61         opacity: 1;
62     }
63
64     /* Add a blue text color to links */
65     a {
66         color: dodgerblue;
67     }
68
69     /* Set a grey background color and center the text of the "sign in" section */
70     .signin {
71         background-color: #f1f1f1;
72         text-align: center;
73     }
74
75     </style>
76
77 </head>
78
79 <body>
80
81     <form>
82         <div class="container">
83             <h1>Sign Up</h1>
84             <p>Please fill in this form to create an account.</p>
85             <n>
86
87             <label for="email"><b>Email</b></label>
88             <input type="text" placeholder="Enter Email" name="email" id="email" required>
89
90             <label for="password"><b>Password</b></label>
91             <input type="password" placeholder="Password" name="psw" id="psw" required>
92
93             <br>
94             <p>Password should be at least 6 characters. <a href="#">Terms & Privacy</a>.</p>
95
96             <button type="button" id="submitData" name="submitData" class="registerbtn">Register</button>
97         </div>
98
99     <div class="container signin">
100        <p>Already have an account? <a href="login.html">Log in</a>.</p>
101    </div>
102
103 </form>
104
105 </body>
106
107 </html>

```

No issues found

Note that the CSS was now implemented inside the HTML page rather than it being separate.

As stated before, when the user enters his credentials to create an account for my website, his details should be stored permanently in a database and the user should be logged in and taken directly to the home page.

| Identifier | Providers | Created | Signed in | User UID | Actions |
|------------|-----------|---------|-----------|----------|---|
| | | | | | <button>Add user</button> <button>Reload</button> C : |

No users for this project yet

Firebase system is now being used and myPHP and SQL have now been abandoned. This is the database table that is set up in firebase and when a user registers an account, refreshing this page and we will see their details stored in this table and the user will also be alerted that their account has been created successfully with a JavaScript alert. Below shows multiple screenshots of the process of someone creating their account and clicking this register button

Sign Up

Please fill in this form to create an account.

Email

dogunrin@outlook.com

Password

.....



Password should be at least 6 characters

Register

Already have an account? [Log in here.](#)

There is also a subtle little feature that I implemented where the user can toggle the eye icon when entering their password to see and to unsee what password they will actually be inputting into the system. This could be helpful to a user if they want to see exactly what they're entering as a password so they do not forget it and are able to memorise it easier. This feature is shown below:

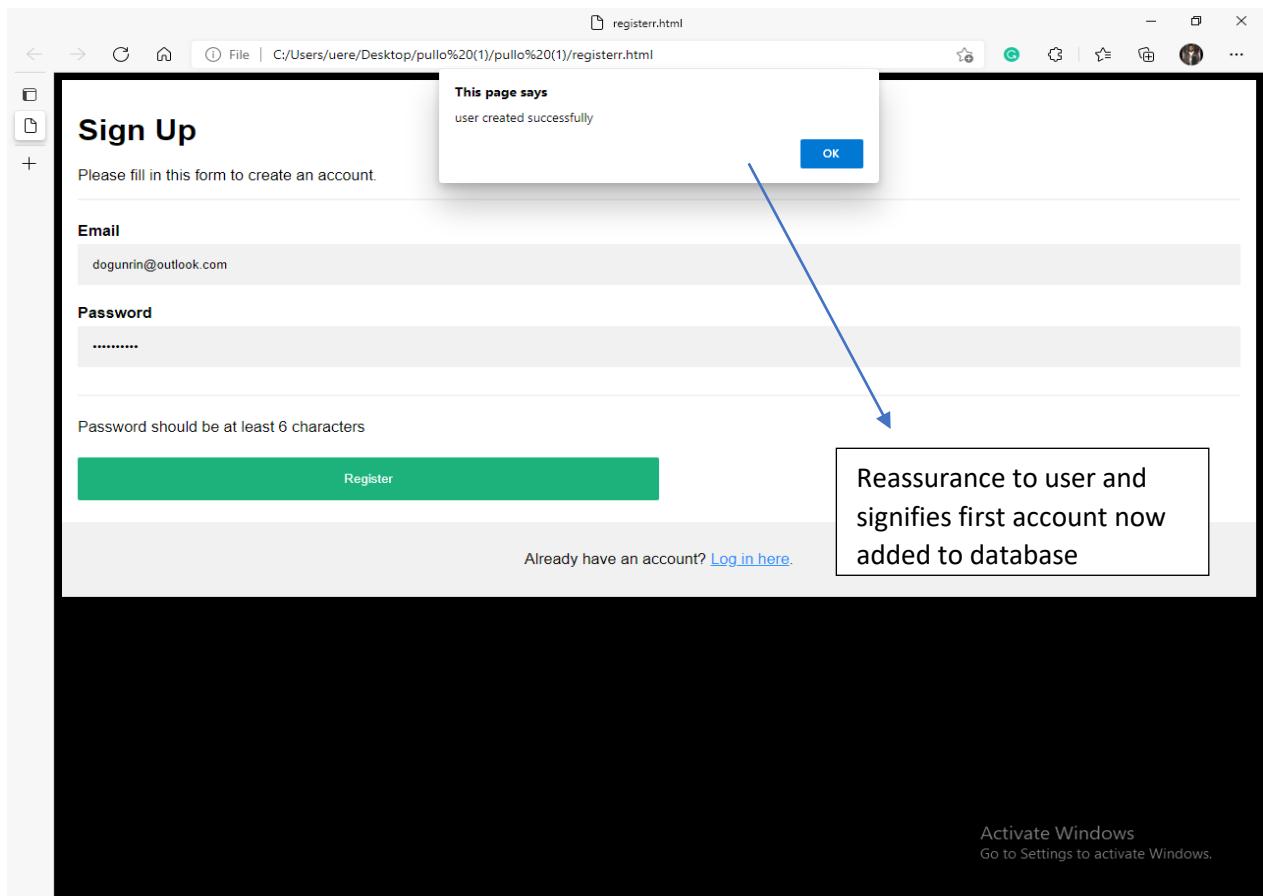
dogunrin@outlook.com

Password

Ogunrin123

After the user clicks the register button a message is dropped down from the top of the screen indicating to the user that their user has been created successfully.

When toggled
password will be
shown



The code for this process can be viewed below

```

140
141 submitData.addEventListener('click', (e) => {
142
143     var email = document.getElementById('email').value;
144     var password = document.getElementById('psw').value;
145
146     //sign up user
147     createUserWithEmailAndPassword(auth, email, password)
148         .then((UserCredential) => {
149             // Signed in
150             const user = UserCredential.user;
151             // ...
152             set(ref(database, 'users/' + user.uid), {
153                 email: email,
154                 password: password
155             })
156             .then(() => {
157                 // Data saved successfully!
158                 alert('user created successfully');
159             })
160             .catch((error) => {
161                 // The write failed...
162                 alert(error);
163             });
164         })
165         .catch((error) => {
166             const errorCode = error.code;
167             const errorMessage = error.message;
168             // ...
169             alert(errorMessage);
170         });
171     }
172

```

Function is called when the register button is clicked. submitData is id name for register button so that must be specified.

Details email and password saved to database

As a test for if this works, I refreshed the database table and checked if these details were added

| Search by email address, phone number or user UID | | | | | Add user | ⋮ |
|---|-----------|-------------|-------------|-------------------------------|----------|---|
| Identifier | Providers | Created | Signed in | User UID | ↑ | |
| dogunrin@outlook.com | ✉ | 18 Apr 2022 | 18 Apr 2022 | Bu9VwXECajUMUvLhEgBmk6adfJ... | | |
| | | | | | | |
| Rows per page 50 1 – 1 of 1 | | | | | ◀ | ▶ |

As we can see creating a new account and adding it to a database worked for me successfully with the details stored in this table with an automatic and unique user ID created for each user to uniquely identify them.

The screenshot shows the Firebase Realtime Database interface. On the left is the Project Overview sidebar with options like Authentication, Firestore Database, Realtime Database, Storage, Hosting, Functions, and Machine Learning. The main area is titled 'Realtime Database' and shows a single user node under 'Data'. The node path is 'https://login-a6a98-default-rtdb.firebaseio.com/users/Bu9VwXECajUMUvLhEgBmk6adfJn1'. Inside this node, there are two children: 'email' with the value 'dogunrin@outlook.com' and 'password' with the value 'Ogunrin123'. A note at the top of the database screen says 'Protect your Realtime Database resources from abuse, such as billing fraud or phishing' with a 'Configure App Check' link.

We can also use the firebase system to view these details in a Realtime database, showing the users email address and password along with their ID.

The code and process for implementing this database and linking/initialising it into my project code is shown below.

```

107
108 <script type="module">
109   // Import the functions you need from the SDKs you need
110   import { initializeApp } from "https://www.gstatic.com/firebasejs/9.6.10/firebase-app.js";
111   import {
112     getAuth,
113     createUserWithEmailAndPassword,
114     signInWithEmailAndPassword,
115     signOut
116   } from "https://www.gstatic.com/firebasejs/9.6.10/firebase-auth.js";
117   import { getDatabase, set, ref, update } from "https://www.gstatic.com/firebasejs/9.6.10.firebaseio.js";
118
119
120   // TODO: Add SDKs for Firebase products that you want to use
121   // https://firebase.google.com/docs/web/setup#available-libraries
122
123   const firebaseConfig = {
124     apiKey: "AIzaSyAMF39H6DZSik_yx9I-2U70nJJ9BKUlyx-Q",
125     authDomain: "login-a6a98.firebaseio.com",
126     databaseURL: "https://login-a6a98-default-rtdb.firebaseio.com",
127     projectId: "login-a6a98",
128     storageBucket: "login-a6a98.appspot.com",
129     messagingSenderId: "964797922748",
130     appId: "1:964797922748:web:1b79c5d17c01f7bd88fbfb9",
131     measurementId: "G-E2BLRVN8RL"
132   };
133
134
135
136
137   // Initialize Firebase
138   const app = initializeApp(firebaseConfig);
139   const auth = getAuth();
140   const database = getDatabase(app);

```

Functions imported from firebase

Database info copied and linked to HTML pages

Database initialised

After a successful registration system was implemented into my website, my next step was to try and code implement my log in system, where the user can log in freely after creating an account and these log in details will be compared and searched for in the database and if these details don't match any details in the database an error message will display. On the other hand, if the details do match, an alert message will display saying log in successful and the user will be automatically taken to the home page. Below you can see my first attempt at this process

```
loginData.addEventListener('click', (e) => {
    var Email = document.getElementById('Email').value;
    var password = document.getElementById('password').value;

    // log in user
    signInWithEmailAndPassword(auth, Email, psw)
        .then((userCredential) => {
            // Signed in
            const user = userCredential.user;
            //

            // save log in details into real time database
            var lgDate = new Date();
            updateRef(database, 'users/' + user.uid), {
                last_login: lgDate,
            })
            .then(() => {
                // Data saved successfully!
                alert('user logged in successfully');
            })
            .catch((error) => {
                // The write failed...
                alert(error);
            });
        })
        .catch((error) => {
            const errorCode = error.code;
            const errorMessage = error.message;
            alert(errorMessage);
        });
    signOut(auth).then(() => {
        // Sign-out successful.
    }).catch((error) => {
        // An error happened.
    });
});
```



THE SNEAKER SUPPLIER

Login

Email:

Password:

Remember me

[Forget password ?](#)

[Log In](#)

Dont have an account? [Register here](#)

[Go Home](#)

After entering the correct details and clicking the button log in, nothing happened. This suggested that there was something wrong with my code as there should be a pop up that says log in successful and I should be taken to the home page

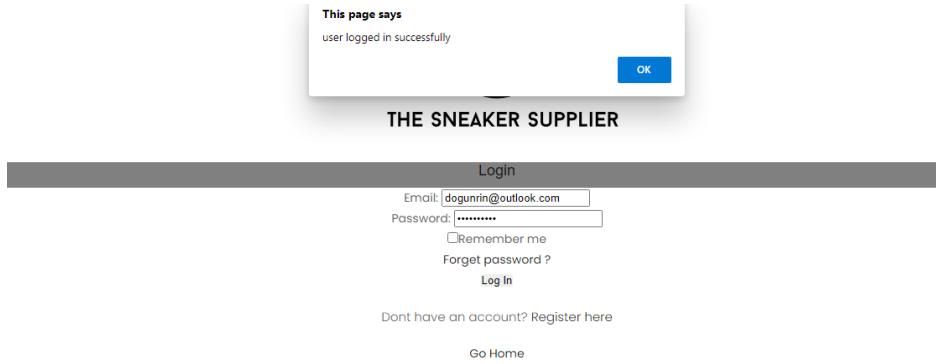
```
141 // Initialize Firebase
142 const app = initializeApp(firebaseConfig);
143 const auth = getAuth();
144 const database = getDatabase(app);
145
146
147
148 const email = document.getElementById('email');
149 const password = document.getElementById('password');
150 const submit = document.getElementById('sub_btn');
151
152 sub_btn.addEventListener('click', RegisterUser)
153
154 function Validation() {
155     let emailregex = /^[a-zA-Z0-9]@[gmail|yahoo|outlook|hotmail].com/;
156     if (!emailregex.test(email.value)) {
157         alert("Enter a valid email");
158         return false;
159     }
160     return true;
161 }
162
163 function RegisterUser() {
164     if (!Validation()) {
165         return;
166     }
167
168     const databaseRef = ref(database)
169
170     getChild(databaseRef, "usersList/" + email.value).then((snapshot) => {
171         if (snapshot.exists()) {
172             alert("Account already exists!");
173         }
174         else {
175             setRef(database, "UsersList/" + email.value,
176             {
177                 email: email.value,
178                 password: pass.value
179             })
180             .then(() => {
181                 alert("user added successfully");
182             })
183             .catch((error) => {
184                 alert("error" + error);
185             })
186         }
187     })
188 }
```

I also attempted to validate myself with these functions and I also tried another way I learned of signing up a user, but it failed to work, my attempt can be seen above

Second Attempt (Log in)

These are my second attempts after analysing and editing my code to fix the problem that I was having which was in the signInWithEmailAndPassword line, the variable/id in the parameter of that function was wrong and password was set to 'psw' and not to the correct id that I set in my HTML log in of 'password'. After I fixed that minor issue, these are the results

```
173 // log in user
174 signInWithEmailAndPassword(auth, email, password)
175 .then((userCredential) => {
176     // Signed in
177     const user = userCredential.user;
178     // ...
179
180     // save log in details into real time database
181     var lgDate = new Date();
182     update(ref(database, 'users/' + user.uid), {
183         last_login: lgDate,
184     })
185     .then(() => {
186         // Data saved successfully!
187         alert('user logged in successfully');
188     })
189     .catch((error) => {
190         // The write failed...
191         alert(error);
192     });
193 })
194 .catch((error) => {
195     const errorCode = error.code;
196     const errorMessage = error.message;
197     alert(errorMessage);
198 });
199
200 signOut(user)
201 signOut(auth).then(() => {
202     // Sign-out successful.
203 }).catch((error) => {
204     // An error happened.
205 });
206
207 });
208
209 </script>
```



As seen above, the code is now successfully working after I entered the correct details that match the ones in my firebase database with the notice saying user has logged in successfully. The code also allows me to see the last log in of any users that have logged in before. After logging into my system, checked the real time database and it successfully showed with the right time that I logged in, proving my log in system successful thus far.

```

https://login-a6a98-default-rtbd.firebaseio.com/.json
{
  "users": {
    "Bu9VwXECajUMUvLhEgBmk6adfJn1": {
      "email": "dogunrin@outlook.com",
      "last_login": "2022-04-05T22:02:43.818Z",
      "password": "Ogunrin123"
    }
  }
}
  
```

Also, if the user tries to enter the wrong password combination an alert is shown saying invalid, as shown below



THE SNEAKER SUPPLIER

Login

Email:

Password:

Remember me

[Forgot password ?](#)

[Log In](#)

Dont have an account? [Register here](#)

[Go Home](#)

This is me entering a wrong password on purpose and me clicking the hide password function to display this.

This page says

Firebase: Error (auth/wrong-password).

[OK](#)

THE SNEAKER SUPPLIER

Login

Email:

Password:

Remember me

[Forgot password ?](#)

[Log In](#)

Dont have an account? [Register here](#)

[Go Home](#)

After checking that this is the wrong password combination for this user, this is the output that comes saying wrong password, showing that this test is **SUCCESSFUL**

The next step would be the page automatically redirecting to the home page after a successful log in:

```

// log in user
signInWithEmailAndPassword(auth, Email, password)
  .then((userCredential) => {
    // Signed in
    const user = userCredential.user;
    // ...

    // save log in details into real time database
    var lgDate = new Date();
    update(ref(database, 'users/' + user.uid), {
      last_login: lgDate,
    })
    .then(() => {
      // Data saved successfully!
      alert('user logged in successfully');
      window.location='home.html'
    })
    .catch((error) => {
      // The write failed...
      alert(error);
    });
  })
  .catch((error) => {
    const errorCode = error.code;
    const errorMessage = error.message;
    alert(errorMessage);
  });
}

```

This line of code takes it to home page after successful log in.
TEST: **SUCCESS**

I implemented this with the line `window.location="home.html"` which automatically takes the user to the home page after clicking the log in button and the logged in successfully alert comes through. This worked out successfully

The next thing that I tried to do was to add and implement my sign-out function, for this I had to implement a logout function on the top navigation bar on the home page after the user had logged in, I did this in the code displayed below:

```

42   <div class="topnav" id="myTopnav">
43     <a href="home.html" class="active">Home</a>
44     <div class="dropdown">
45       <button class="dropbtn">
46         Streetwear</i>
47       </button>
48       <div class="dropdown-content">
49         <a href="lateststreetwearrelease.html">Latest Streetwear Releases</a>
50         <a href="upcomingstreetwearrelease.html">Upcoming Streetwear Releases</a>
51       </div>
52     </div>
53     <a href="contact.html">Contact</a>
54     <a href="Login.html">Login</a>
55     <button class="btn btn-danger" id="logoutData">Logout</button>
56     <div class="dropdown">
57       <button class="dropbtn">
58         Footwear</i>
59       </button>
60       <div class="dropdown-content">
61         <a href="bestsellingsneaker.html">Best Selling Sneakers</a>
62         <a href="latesstsneakerrelease.html">Latest Sneaker Releases</a>
63         <a href="upcomingsneaker.html">Upcoming Sneaker Releases</a>
64       </div>
65     </div>

```

This outputted a logout button in red (after editing with CSS) on my top navbar after a user logged in and the function that I designed to be run after clicking that logout button is shown in the code below:

```
firebase.auth().onAuthStateChanged((user) => {
  if (!user) {
    location.replace("login.html")
  } else {
    document.getElementById("user").innerHTML = "Hello, " + user.email
  }
})

logoutData.addEventListener('click', (e) => {
  const auth = getAuth();
  signOut(auth).then(() => {
    // Sign-out successful.
  }).catch((error) => {
    // An error happened.
  });
});
```

However, I ran into some trouble as not only was the function not logging out and returning back to the log in page, it was not made clear that the user had logged in. I tried various ways to change this and they were not working:

HOME SCREEN

Under this section I will provide code and explanation for all the features on the home screen. As stated, before in my design section a lot of features and products will be available on the home page so it gives the user a maximum time efficiency feels as they can click on items straight from the home page. Below you will find code that will be annotated and the display after running that code

Starting with initialising the HTML document and inputting all the necessary css style sheets, some you will find screenshots for in later on. The title is inputted for the page and the logo that I designed for my company will also be implemented at the top of the page, and will be consistent throughout all pages on my website.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <!-- basic -->
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <!-- mobile metas -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="viewport" content="initial-scale=1, maximum-scale=1">
    <!-- site metas -->
    <title>The Sneaker Supplier</title>
    <meta name="keywords" content="">
    <meta name="description" content="">
    <meta name="author" content="">
    <!-- bootstrap css -->
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <!-- style css -->
    <link rel="stylesheet" href="css/style.css">
    <!-- Responsive-->
    <link rel="stylesheet" href="css/responsive.css">
    <!-- favicon -->
    <link rel="icon" href="images/favicon.png" type="image/gif" />
    <!-- Scrollbar Custom CSS -->
    <link rel="stylesheet" href="css/jquery.mCustomScrollbar.min.css">
    <!-- Tweaks for older IEs-->
    <link rel="stylesheet" href="https://netdna.bootstrapcdn.com/font-awesome/4.0.3/css/font-awesome.css">
    <!-- owl stylesheets -->
    <link rel="stylesheet" href="css/owl.carousel.min.css">
    <link rel="stylesheet" href="css/owl.theme.default.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/fancybox/2.1.5/jquery.fancybox.min.css" media="screen">
        <link rel="stylesheet" href="css/navbar.css">
</head>
<!-- body -->
<body class="main-layout">
    <!-- header section start -->
    <div class="header_section">
        <div class="container">
            <div class="row">
                <div class="col-sm-3">
                    <div class="logo"><a href="#"></a></div>
                </div>

```

Top navigation bar code and display

```

<div class="header_section">
    <div class="container">
        <div class="row">
            <div class="col-sm-3">
                <div class="logo"><a href="#"></a></div>
            </div>
<div class="topnav" id="myTopnav">
    <a href="home.html" class="active">Home</a>
    <div class="dropdown">
        <button class="dropbtn">Streetwear<i class="fa fa-caret-down"></i>
        </button>
        <div class="dropdown-content">
            <a href="lateststreetwearrelease.html">Latest Streetwear Releases</a>
            <a href="upcomingstreetwearrealease.html">Upcoming Streetwear Releases</a>
        </div>
    </div>
    <a href="contact.html">Contact</a>
    <a href="login.html">Login</a>

    <div class="dropdown">
        <button class="dropbtn">Footwear<i class="fa fa-caret-down"></i>
        </button>
        <div class="dropdown-content">
            <a href="bestsellingsneaker.html">Best Selling Sneakers</a>
            <a href="latesnsneakerrelease.html">Latest Sneaker Releases</a>
            <a href="upcomingsneaker.html">Upcoming Sneaker Releases</a>
        </div>
    </div>
    <a href="reminders.html">Reminders</a>
    <a href="about.html">About</a>

```

HTML section for the logo and the navigation bar in the heading section of the page, classed in the line “header_section”. The class and id topnav give an indication for the id that is used to design this top nav in CSS and also to make it responsive in JavaScript, as seen in the JavaScript screenshot below. The class in ‘active’ makes the home feature on the bar turn a highlighted green colour due to the style in my CSS design as referenced when describing the CSS for this code. All features in the bar are within the top nav class and are listed and labelled such as [Contact](contact.html). This similar line of code is used for each feature that is not a dropdown feature. The [is for the link of the page that the user will be taken to when they click on the Contact feature and thus will be the same for any other feature clicked.](contact.html)

<div class="dropdown"> This is a separate class for any feature in the navbar that has a dropdown menu with listed categories and this line of code identifies that

 <button class="dropbtn"> Another class for the button, when clicked the drop down should show the categories listed

 Streetwear<i class="fa fa-caret-down"></i> This line is the symbol for the drop down which is the arrow pointing down. This is the first example of a visual symbol that is used in my website so to help users that are more visual and to also add to the aesthetic of the site.

 </button>

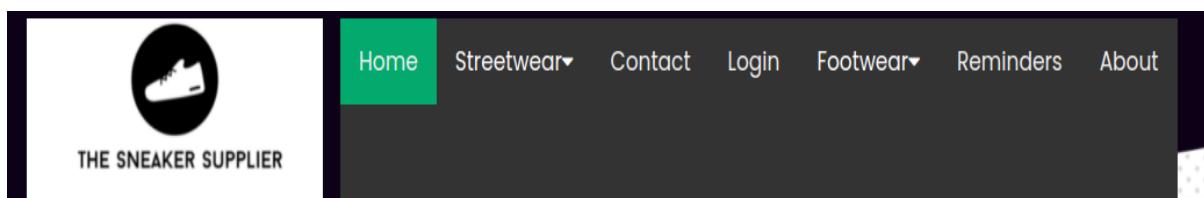
<div class="dropdown-content"> Another separate class for the dropdown content

[Latest Streetwear Releases](lateststreetwearrelease.html)

[Upcoming Streetwear Releases](upcomingstreetwearrealease.html)

Again, both href lines e links to the separate pages that are titled in the line of code for example Latest streetwear releases and they will be both under the feature dropdown. When clicking on either link it will take you to the corresponding HTML page. This is the same for the footwear feature.

 </div>



```

function myFunction() {
  var x = document.getElementById("myTopnav");
  if (x.className === "topnav") {
    x.className += " responsive";
  } else {
    x.className = "topnav";
  }
}

```

CSS code is under the file name navbar.css, specifically a separate CSS file for the navigation bar design.

```

/* Add a black background color to the top navigation */
.topnav {
  background-color: #333;
  overflow: hidden;
}

/* Style the links inside the navigation bar */
.topnav a {
  float: left;
  display: block;
  color: #f2f2f2;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 17px;
}

/* Add an active class to highlight the current page */
.active {
  background-color: #04AA6D;
  color: white;
}

/* Hide the link that should open and close the topnav on small screens */
.topnav .icon {
  display: none;
}

/* Dropdown container - needed to position the dropdown content */
.dropdown {
  float: left;
  overflow: hidden;
}

```

```
/* Style the dropdown button to fit inside the topnav */
.dropdown .dropbtn {
  font-size: 17px;
  border: none;
  outline: none;
  color: white;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;
}

/* Style the dropdown content */
.dropdown-content {
  display: list-item;
  position: absolute;
  background-color: #f9f9f9;
  min-width: 160px;
  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}

/* Style the links inside the dropdown */
.dropdown-content a {
  float: none;
  color: black;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
}

/* Add a dark background on topnav links and the dropdown button on hover */
.topnav a:hover, .dropdown:hover .dropbtn {
  background-color: #555;
  color: white;
}
```

```

/* Show the dropdown menu when the user moves the mouse over the dropdown button */
.dropdown:hover .dropdown-content {
  display: list-item;
}

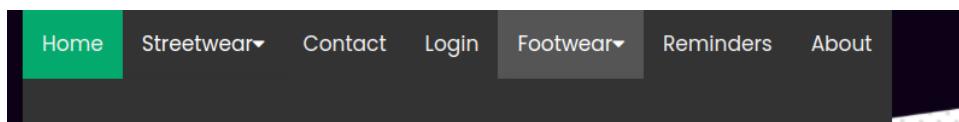
/* When the screen is less than 600 pixels wide, hide all links, except for the first one ("Home"). Show the link that contains should open and close the topnav (.icon) */
@media screen and (max-width: 600px) {
  .topnav a:not(:first-child), .dropdown .dropbtn {
    display: none;
  }
  .topnav a.icon {
    float: right;
    display: block;
  }
}

/* The "responsive" class is added to the topnav with JavaScript when the user clicks on the icon. This class makes the topnav look good on small screens (display the links vertically instead of horizontally) */
@media screen and (max-width: 600px) {
  .topnav.responsive {position: relative;}
  .topnav.responsive a.icon {
    position: absolute;
    right: 0;
    top: 0;
  }
  .topnav.responsive a {
    float: none;
    display: block;
    text-align: left;
  }
  .topnav.responsive .dropdown {float: none;}
  .topnav.responsive .dropdown-content {position: relative;}
  .topnav.responsive .dropdown .dropbtn {
    display: block;
    width: 100%;
    text-align: left;
  }
}

```

This is all the CSS design code that I used to design my navigation bar. As you can see, I coded with comments annotations so it is easy to decipher which line of code designs each bit in my bar. For example, the active class to highlight the current displayed page code, starting with .active, is used to highlight the feature in the navbar that the website is currently displaying and, in the screenshot shown, the current page displayed is the home page which is why the feature on the top navbar labelled 'home' is highlighted in green due to the line of code: background-color: #04AA6D;

When first trying to code and use the drop-down feature in my top navigation bar, when hovering over the streetwear and footwear features their subcategories failed to display and drop down, as shown below:



This photo shows me hovering over the footwear section but it failing to actually drop down, to fix this I analysed my HTML code for my navbar and I checked for any errors. As of now this is test is a

FAIL

Product Carousel

```
<a href="javascript:void(0);> class="icon" onclick="myFunction()">&#9776;</a>
</div>
<div class="banner_section">
    <div class="container-fluid">
        <section class="slide-wrapper">
<div class="container-fluid">
    <div id="myCarousel" class="carousel slide" data-ride="carousel">
        <!-- Indicators -->
        <ol class="carousel-indicators">
            <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
            <li data-target="#myCarousel" data-slide-to="1"></li>
            <li data-target="#myCarousel" data-slide-to="2"></li>
            <li data-target="#myCarousel" data-slide-to="3"></li>
        </ol>
```

Above is the section of code in HTML that is responsible for the carousel on my home page. The carousel is the most important feature of my home page as it meets the clients' needs of having a quick and efficient way of navigating through the site to access any releases that they would otherwise have a slim chance of coping. A JavaScript function for the animation and item slider is called to when clicked called "myFunction". Code for carousel sliders and items is classed under banner_section. As annotated in the code as "Indicators", the four lines under this represents the 4 items in the carousel and data-slide-to="0" shows the item numbers of each carousel item in the ordered list and shows which item the carousel will slide next to, going from 0-3 in consecutively. class="active"> shows that this is the first item that should load up and will always be active and shown first.

CSS for carousel

```

.owl-carousel, .owl-carousel .owl-item{-webkit-tap-highlight-color:transparent;position:relative}.owl-carousel{display:none;width:100%;z-index:1}.owl-carousel .owl-stage{position:relative;-ms-touch-action:pan-Y;touch-action:manipulation;-moz-backface-visibility:hidden}.owl-carousel .owl-stage:after{content:".";display:block;clear:both;visibility:hidden;line-height:0;height:0}.owl-carousel .owl-stage-outer{position:relative;overflow:hidden;-webkit-transform:translate3d(0,0,0)}.owl-carousel .owl-item, .owl-carousel .owl-wrapper{-webkit-backface-visibility:hidden;-moz-backface-visibility:hidden;-ms-backface-visibility:hidden;-webkit-transform:translate3d(0,0,0);-moz-transform:translate3d(0,0,0);-ms-transform:translate3d(0,0,0)}.owl-carousel .owl-item{min-height:1px;float:left;-webkit-backface-visibility:hidden;-webkit-touch-callout:none}.owl-carousel .owl-item img{display:block;width:100%}.owl-carousel .owl-dots.disabled, .owl-carousel .owl-nav.disabled{display:none}.no-js .owl-carousel, .owl-carousel.owl-loaded{display:block}.owl-carousel .owl-dot, .owl-carousel .owl-nav .owl-next, .owl-carousel .owl-prev{cursor:pointer;-webkit-user-select:none;-khtml-user-select:none;-moz-user-select:none;-ms-user-select:none;user-select:none}.owl-carousel .owl-nav button.owl-next, .owl-carousel .owl-nav button.owl-prev, .owl-carousel button.owl-dot{background:0 0;color:inherit;border:none;padding:0!important;font:inherit}.owl-carousel.owl-loading{opacity:0;display:block}.owl-carousel.owl-hidden{opacity:0}.owl-carousel.owl-refresh .owl-item{visibility:hidden}.owl-carousel.owl-drag .owl-item{-ms-touch-action:none;touch-action:none;-webkit-user-select:none;-moz-user-select:none;-ms-user-select:none;user-select:none}.owl-carousel.owl-grab{cursor:move;cursor:grab}.owl-carousel.owl-rtl{direction:rtl}.owl-carousel.owl-rtl .owl-item{float:right}.owl-carousel .animated{animation-duration:1s;animation-fill-mode:both}.owl-carousel .owl-animated-in[z-index:0].owl-carousel .owl-animated-out{z-index:1}.owl-carousel .fadeOut{animation-name:fadeOut}@keyframes fadeOut{0%{opacity:1}100%{opacity:0}}.owl-height{transition:height .5s ease-in-out}.owl-carousel .owl-item .owl-lazy{opacity:0;transition:opacity .4s ease}.owl-carousel .owl-item img.owl-lazy{transform-style:preserve-3d}.owl-carousel .owl-video-wrapper{position:relative;height:100%;background:#000}.owl-carousel .owl-video-play-icon{position:absolute;height:80px;width:80px;left:50%;top:50%;margin-left:-40px;margin-top:-40px;background:url(owl.video.play.png) no-repeat;cursor:pointer;z-index:1;-webkit-backface-visibility:hidden;transition:transform .1s ease}.owl-carousel .owl-video-play-icon:hover{-ms-transform:scale(1.3,1.3);transform:scale(1.3,1.3)}.owl-carousel .owl-video-playing .owl-video-play-icon,.owl-carousel .owl-video-playing .owl-video-tn{display:none}.owl-carousel .owl-video-tn{opacity:0;height:100%;background-position:center center;background-repeat:no-repeat;background-size:contain;transition:opacity .4s ease}.owl-carousel .owl-video-frame{position:relative;z-index:1;height:100%;width:100%}

```

This designs the animation, positioning of text and pictures, transitions in slides, background colours etc. For my carousel saved in a file owl.carousel.min.css, referenced to in the beginning of the HTML page

JavaScript for carousel

```

$(document).ready(function () {
    var owl = $('#product-in-slider');
    owl.owlCarousel({
        loop: true,
        nav: true,
        margin: 10,
        navText: ["<i class='fa fa-angle-left'></i>", "<i class='fa fa-angle-right'></i>"],
        responsive: {
            0: {
                items: 1
            },
            600: {
                items: 2
            },
            960: {
                items: 3
            },
            1200: {
                items: 4
            }
        }
    });
    owl.on('mousewheel', '.owl-stage', function (e) {
        if (e.deltaY > 0) {
            owl.trigger('next.owl');
        } else {
            owl.trigger('prev.owl');
        }
        e.preventDefault();
    });
});

```

This function for the carousel shows how the product slides up and down and also describes the loop that takes place for the carousel to repeatedly keep scrolling through items in the line loop: true.

1st carousel item code and display

```

<!-- Wrapper for slides -->
<div class="carousel-inner">
  <div class="carousel-item active">
    <div class="row">
      <div class="col-sm-2 padding_0">
        <p class="mens_taital">Men Shoes</p>
        <div class="page_no">1/4</div>
        <p class="mens_taital_2">Men Shoes</p>
      </div>
      <div class="col-sm-5">
        <div class="banner_taital">
          <h1 class="banner_text">New Upcoming Shoes </h1>
          <h1 class="mens_text"><strong>Adidas Yeezy 350 V2 "Dazzling Blue"</strong></h1>
          <p class="lorem_text">The Adidas Yeezy 350 lives up to its name with its striking design. This edition of the Boost 350 V2 silhouette showcases classic features redressed in a sleek colourway. Shop for Yeezy Boost 350 on the Sneaker Supplier website. Be sure to make an account to stay on top of the newest drops from Kanye West's Yeezy.</p>
          <form>
            <button class="more_bt" formaction="upcomingsneaker.html">See More</button>
          </form>
        </div>
      </div>
      <div class="col-sm-5">
        <div class="shoes_img"></div>
      </div>
    </div>
  </div>

```

All classes needed for the carousels that can be referenced to in Java and CSS are initialised with `<div class="">` and will end with `</div>`. The classes for paragraphs and headings are initialised in the same way but with h1 and p instead of div.

`<div class="page_no">1/4</div>` Indication of which item you are on the carousel. For this specific example it shows that we are on the 1st item the Adidas Yeezys. The ¼ is shown in a circular symbol and the left-hand side which was designed in the carousel CSS, as where all pictures, buttons and text.

`<h1 class="banner_text">New Upcoming Shoes </h1>` This is the heading which indicates whether the product is upcoming, bestselling or latest releases which are all pages in the footwear section. This at the top in big letters and it's in pink font to draw the user's attention upon entering the home page and looking at the carousel items. Gives strong sense of presence and efficiency to the website

`<h1 class="mens_text">Adidas Yeezy 350 V2 "Dazzling Blue"</h1>` This line is for the full name of the shoe which is bold letters through the use strong in the HTML code. This is to so the user can see clearly the shoe name first as to see if it is the shoe that they are looking for.

`<p class="lorem_text">The Adidas Yeezy 350 lives up to its name with its striking design. This edition of the Boost 350 V2 silhouette showcases classic features redressed in a sleek colourway. Shop for Yeezy Boost 350 on the Sneaker Supplier website. Be sure to make an account to stay on top of the newest drops from Kanye West's Yeezy.</p>` Brief paragraph description in the carousel of the shoe for information if the user wishes to read

```

<form>
  <button class="more_bt" formaction="upcomingsneaker.html">See More</button>

```

`</form>` This section is for the button of see more and will take the user to the page where similar items and the item itself resides on the website and from here the user can then favourite the product if they're logged in or just select the products and click the link to the website to where they can buy the item.

```

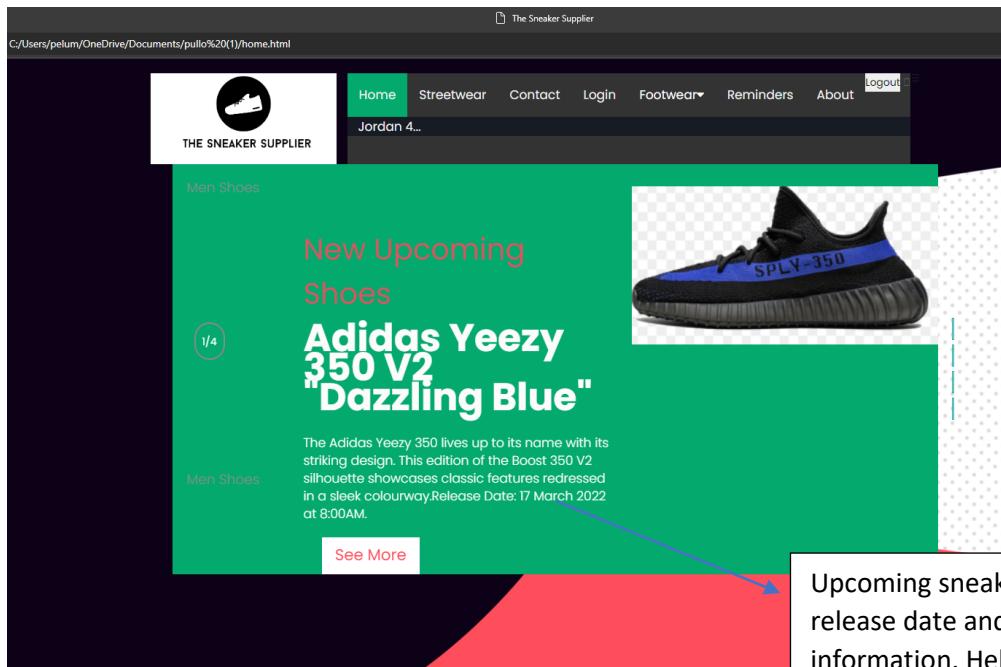
        </div>
    </div>
<div class="col-sm-5">

```

<div class="shoes_img"></div>

The section where the picture of the shoe is implemented through the link. All images used in this whole website is stored in the file images

This code template is used for all remaining slides in the carousel as shown in all the screenshots below.



2nd carousel item code and display

Upcoming sneaker so it includes the release date and time in the information. Helpful to user if they want to know when this shoe drops and reinforces the purpose of the app, informing users of restocks and releases of hot products

```

<div class="carousel-item">
    <div class="row">
        <div class="col-sm-2 padding_0">
            <p class="mens_taital">Men Shoes</p>
            <div class="page_no">2/4</div>
            <p class="mens_taital_2">Men Shoes</p>
        </div>
        <div class="col-sm-5">
            <div class="banner_taital">
                <h1 class="banner_text">Latest Sneaker Drop</h1>
                <h1 class="mens_text"><strong>Nike SB Dunk Low Chlorophyll</strong></h1>
                <p class="lorem_text">The Nike SB Dunk Low Chlorophyll brings back a famed colourway, best known for its appearance across the Air Trainer 1 and Air Jordan 3. Its striking mix of bold green, black and medium grey is easily adopted by the SB Dunk Low Pro.</p>

```

Men Shoes

Latest Sneaker Drop Nike SB Dunk Low Chlorophyll

2/4

Men Shoes

See More



The Nike SB Dunk Low Chlorophyll brings back a famed colourway, best known for its appearance across the Air Trainer 1 and Air Jordan 3. Its striking mix of bold green, black and medium grey is easily adopted by the SB Dunk Low Pro.

3rd carousel item code and display

```
<div class="carousel-item">
  <div class="row">
    <div class="col-sm-2 padding_0">
      <p class="mens_taital">Men Streetwear</p>
      <div class="page_no">3/4</div>
      <p class="mens_taital_2">Men Streetwear</p>
    </div>
    <div class="col-sm-5">
      <div class="banner_taital">
        <h1 class="banner_text">Latest Streetwear Drop </h1>
        <h1 class="mens_text"><strong>Mens Hyperdrive Trapstar Jacket</strong></h1>
        <p class="lorem_text">One of the Best Streewear brands about right now, see the latest new Trapstar
        hyperdrive Jacket from the latest Trapstar Drop</p>
        <form>
          <button class="more_bt" formaction="lateststreetwearrelease.html">See More</button>
        </form>
      </div>
    </div>
    <div class="col-sm-5">
      <div class="shoes_img"></div>
    </div>
  </div>
</div>
```

Latest Streetwear

Drop

3/4

Mens Hyperdrive Trapstar Jacket

One of the Best Streetwear brands about right now, see the latest new Trapstar Hyperdrive Jacket from the latest Trapstar Drop

[See More](#)



4th Carousel item code and display

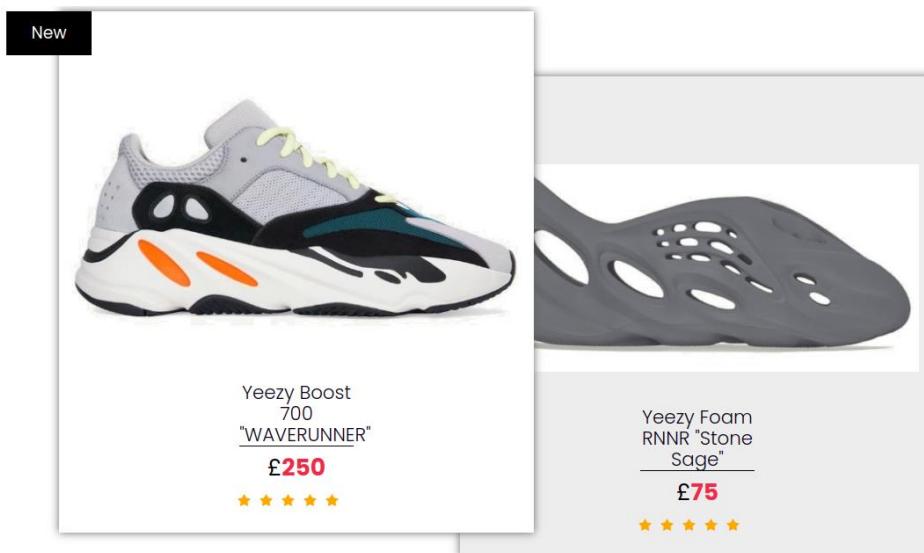
```

<div class="carousel-item">
    <div class="row">
        <div class="col-sm-2 padding_0">
            <p class="mens_taital">Men Streetwear</p>
            <div class="page_no">4/4</div>
            <p class="mens_taital_2">Men Streetwear</p>
        </div>
        <div class="col-sm-5">
            <div class="banner_taital">
                <h1 class="banner_text">Latest Corteiz Drop</h1>
                <h1 class="mens_text"><strong>Grey Cargo Tracksuit</strong></h1>
                <p class="lorem_text">Another Hyped Brand, see the latest Corteiz drop here</p>
                <form>
                    <button class="more_bt" formaction="lateststreetwearrelease.html">See More</button>
                </form>
            </div>
        </div>
        <div class="col-sm-5">
            <div class="shoes_img"></div>
        </div>
    </div>
</div>
</div>
</div>
<!-- header section end -->
```



Carousel feature
successfully implemented,
TEST: **SUCCESS**

As you scroll down the website after the carousel you will see a section which looks like this:



The 'New' pop up on the top left corner shows that these are new items and are on the home page just to show the user what has recently dropped that they might be interested in, and also shows information such as the price and the items popularity rating as well as the name. The popularity rating is determined by me after thorough research on what shoes are best-selling, how many people are buying them, the amount of people talking about the shoe on social media and many different factors that allowed me to come to a judgement to what rating the shoes should have. In the example above each shoe showcases a 5-star rating indicating that these are two of the best shoes that are about right now, which is another reason as to why they are on the home page.

The code for this part of the website is shown below:

The title for this section is labelled “Yeezy March” and is in bold text using the strong tags.

<div class="shoes-img2"></div> The class for the image of the shoes is initialised and the shoe image file link is referenced to from the images file. This particular product is the second shoe that I saved so it is saved as shoes2.jpg as a JPG image.

<p class="sport_text">Yeezy Foam RNNR "Stone Sage"</p> This is the full name of the shoe

<div class="dolar_text">£<strong style="color: #f12a47;">75 </div>

This line shows the prices of the shoes with bold writing with more strong tags. In this case the shoes are worth £75. There is some in line CSS in this line of code for the color of the price and it is displayed in bold pink which is a consistent theme colour in the whole of my website.

<div class="star_icon"> This div class shows how the star ratings were implemented onto the item cards, placed under the price. For this specific item it was given a 5-star rating

The icons were implemented with an actual image that I found online and then I changed the size of them and designed them to all be in line in a list using CSS to resize and put them next to each other.

Scrolling down from the yeezy march section on the home page, the next section that we come across is the section for the bestselling shoe. Best selling shoes is already a category in itself under the footwear dropdown and can be accessed on a separate HTML page by clicking this category on the dropdown navigation bar. On the home page however only the shoe ranked #1 in bestselling is shown to the user and in this case, it is the Dunk Low Retro “White Black”. This is basically informing the user what is the most popular shoe right now in the shoe market and if the user is a reseller like my client, he can use this information to his advantage and potentially make a profit from this shoe by buying them and reselling them. The user is taken to the bestselling sneaker page when the user clicks the see more button under the shoe picture name and price where he can favourite the item and also click on it to view the links to where the user can purchase the shoe. The page will also show all the other best-selling shoes as well that the user can do the same thing with. A picture and the code for this section of the home page is viewed below.

```
<div class="collection_section">
  <div class="container">
    <h1 class="new_text"><strong>Best Selling Shoe</strong></h1>
    <p class="consectetur_text">The Shoe that's the most popular right now amongst the people! Click below for more info</p>
  </div>
</div>
<div class="racing_shoes">
  <div class="row">
    <div class="collectionn_section_3 layout_padding">
      <div class="container">
        <div class="col-md-8">
          <div class="shoes-img3"></div>
        </div>
        <div class="col-md-4">
          <div class="sale_text"><strong>Dunk Low<br><span style="color: #0a0506;">White</span></strong><br><strong>£<span style="color: #0a0506">89.99</span></strong></div>
          <form>
            <button class="seemore" formaction="bestsellingsneaker.html">See More</button>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
```

The Shoe that's the most popular right now amongst the people! Click below for more info



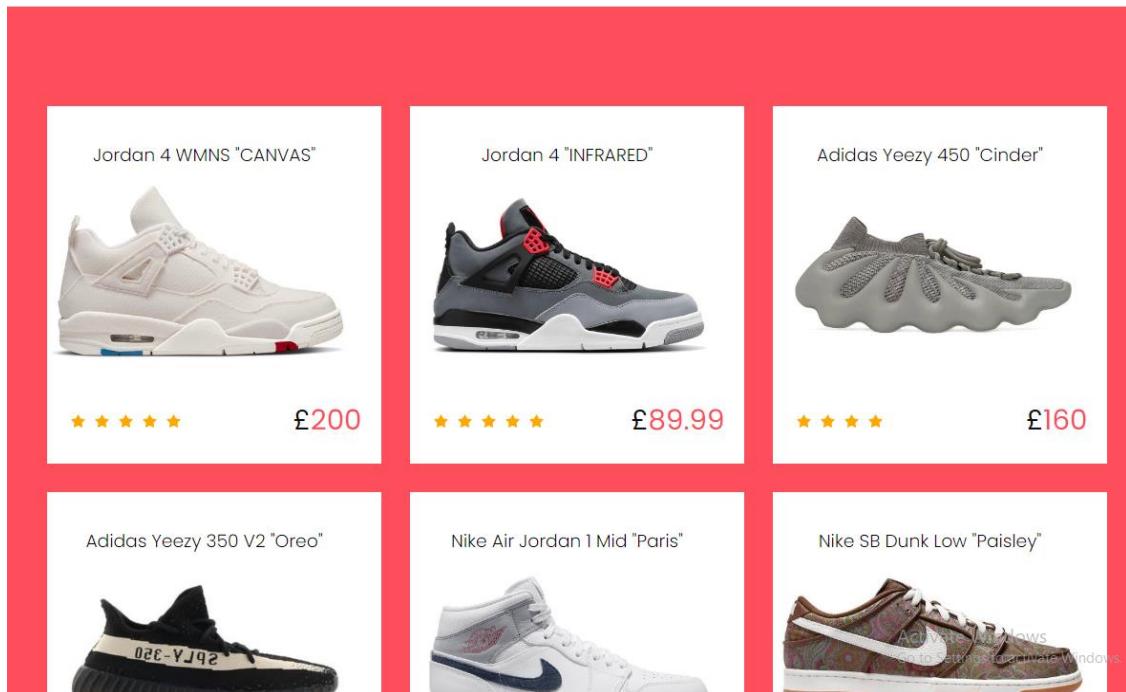
**Dunk Low
"White
Black"**
Nike Mens
£89.99

See More

Shows the most popular items are implemented into my home page, this test is a **SUCCESS**

Upcoming Releases

These are some of the hottest releases dropping in the next few weeks!



As you scroll down from the best-selling shoe you will get to the upcoming shoes section on the homepage which has 6 item cards as discussed in my design for how my products will look on their pages. This section is identical to the separate HTML page under the footwear section labelled upcoming sneaker releases. As you can see, on each product card there is in this order:

Full name and title of the shoe

Image of the shoe

Star rating of the shoe

Price of the shoe

Favourite icon (not yet been implemented on this page)

This is the code for all of these features below

```
div class="collection_section layout_padding">
  <div class="container">
    <h1 class="new_text"><strong>Upcoming Releases</strong></h1>
    <p class="consectetur_text">These are some of the hottest releases dropping in the next few weeks!</p>
  </div>
  !-- new arrivals end -->
  !-- Upcoming Releases section start -->
<div class="layout_padding gallery_section">
  <div class="container">
    <div class="row">
      <div class="col-sm-4">
        <div class="best_shoes">
          <p class="best_text">Jordan 4 WMNS "CANVAS" </p>
          <div class="shoes_icon"></div>
          <div class="star_text">
            <div class="left_part">
              <ul>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
              </ul>
            </div>
            <div class="right_part">
              <div class="shoes_price">£<span style="color: #ff4e5b;">200</span></div>
            </div>
          </div>
        </div>
        <div class="col-sm-4">
          <div class="best_shoes">
            <p class="best_text">Jordan 4 "INFRARED" </p>
            <div class="shoes_icon"></div>
            <div class="star_text">
              <div class="left_part">
                <ul>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                </ul>
              </div>
              <div class="right_part">
                <div class="shoes_price">£<span style="color: #ff4e5b;">89.99</span></div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

Contact Section

Contact Now

Send



```
release.html      bestsellsneaker.html      js.js      HTMLPage1.html      login.html      home.html* ▶ X
[...]
</div>
</div>
<!-- New Arrivals section end -->
<!-- contact section start -->


# <Contact Now>



<form action="/action_page.php">
  <div class="form-group">
    <input type="text" class="email-bt" placeholder="Name" name="Name">
  </div>
  <div class="form-group">
    <input type="text" class="email-bt" placeholder="Phone Number" name="Name">
  </div>
  <div class="form-group">
    <input type="text" class="email-bt" placeholder="Email" name="Email">
  </div>

  <div class="form-group">
    <textarea class="massage-bt" placeholder="Message" rows="5" id="comment" name="Message"></textarea>
  </div>
</form>



<button class="main_bt">Send</button>



<button class="out_shop_bt">Our Shop</button>


<!-- contact section end -->


```

Activate Windows

As we scroll down from the upcoming releases section and scroll down to the final sections of the home page, we come across a contact section that is also a separate page that can be accessed through my top navigation bar. The contact section includes the contact now header and under this are the text boxes that requires the user's input. The information that is requested if the user wishes to send a message to the website is the name of the user, the phone number, the email and the actual message they want to write. This will all be stored in a database and the process is shown below.

```
77
78
79     <div class="collection_text">Contact Us</div>
80     <div class="layout_padding contact_section">
81         <div class="container">
82             <h1 class="new_text"><strong>Contact Now</strong></h1>
83         </div>
84         <div class="conatiner-fluid ram">
85             <div class="row">
86                 <div class="col-md-6">
87                     <div class="email_box">
88                         <div class="input_main">
89                             <div class="container">
90
91                                 <form action="" id="contactForm">
92                                     <div class="alert">Your message sent</div>
93
94                                     <div class="form-group">
95                                         <input type="text" class="email-bt" id="name" placeholder="Full Name" />
96                                     </div>
97
98                                     <div class="form-group">
99                                         <input type="email" class="email-bt" id="emailid" placeholder="Email Address" />
100                                    </div>
101
102                                     <div class="form-group">
103                                         <input type="number" class="email-bt" id="number" placeholder="Number" />
104                                     </div>
105
106                                     <div class="form-group">
107                                         <textarea class="massage-bt" id="msgContent" rows="5" placeholder="Message"></textarea>
108                                     </div>
109
110                                     <div class="form-group">
111                                         <button class="main_bt" type="submit">Send</button>
112                                     </div>
113                                 </form>
114                             </div>
115                         </div>
116                     </div>
117                 </div>
118                 <div class="col-md-6">
119                     <div class="shop_banner">
120                         <div class="our_shop">
121                             <button class="our_shop_bt">Our Shop</button>
122                         </div>
123                     </div>
124                 </div>
125             </div>
126         </div>
127     </div>
128
129     <!-- section footer start -->
```

Activate Windows
Go to Settings to activate Windows

To make this work perfectly with perfect functionality I tweaked my contact system class names and ids to make it easier for me to understand and implement in my JavaScript code when implementing into my database, and also for the CSS code that I will use

```

pullo (1) JavaScript Content Files          submitForm           setTimeout() callback
1  const firebaseConfig = {
2    apiKey: "AIzaSyC8UHIBI3MeISS57YP2zx2AeuINNg3HZRc",
3    authDomain: "contactform-59cef.firebaseio.com",
4    databaseURL: "https://contactform-59cef-default.firebaseio.com",
5    projectId: "contactform-59cef",
6    storageBucket: "contactform-59cef.appspot.com",
7    messagingSenderId: "638237278138",
8    appId: "1:630237278138:web:c290b185f64f1333fb3783",
9    measurementId: "G-4LRVND6EWJ"
10   };
11
12
13 // Initialize Firebase
14 firebase.initializeApp(firebaseConfig);
15
16
17 // reference your database
18 var contactFormDB = firebase.database().ref("contactForm");
19
20 document.getElementById("contactForm").addEventListener("submit", submitForm);
21
22
23 function submitForm(e) {
24   e.preventDefault();
25
26   var name = getElementVal("name");
27   var emailid = getElementVal("emailid");
28   var msgContent = getElementVal("msgContent");
29
30   saveMessages(name, emailid, msgContent);
31
32   // enable alert
33   document.querySelector(".alert").style.display = "block";
34
35   // remove the alert
36   setTimeout(() => {
37     document.querySelector(".alert").style.display = "none";
38   }, 3000);
39
40   // reset the form
41   document.getElementById("contactForm").reset();
42 }
43
44 const saveMessages = (name, emailid, msgContent) => {
45   var newContactForm = contactFormDB.push();
46
47   newContactForm.set({
48     name: name,
49     emailid: emailid,
50     msgContent: msgContent,
51   });
52
53 }
54
55 const getElementVal = (id) => {
56   return document.getElementById(id).value;
57 }
58

```

Form is reset after the user has sent his message and all input fields are cleared up for another message

Function for when the user clicks the submit/send

Setting variables

Initializing the firebase so after the messages are sent, they are saved here successfully

Here I tried to make a pop-up alert after the user submits to signify it has been received. The alert would disappear after a period of time using the line setTimeout.

The data and contents of all input fields are saved in this format in the database after submitting

This was my code that I used to implement my database into my HTML page. The database saves the users inputs and their feedback message for me to see and after the user clicks send, an alert pops up saying thank you and then the contact form is immediately reset, and then the alert message goes away automatically after a while. Whereas the code for this was very hard and challenging for me, the output makes it look simply. The output when running and testing is displayed below:

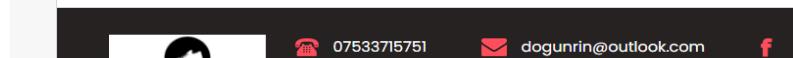
Contact Now

Daniel

dogunrin@outlook.com

Fantastic Website!

Send



These are the inputs that I entered into my text boxes just to test the system and below are the outputs into the database

Realtime Database

Data Rules Backups Usage

https://contactform-59cef-default-rtbd.firebaseio.com/contactForm/contactForm/-N07-hm10BioiNir3wa3

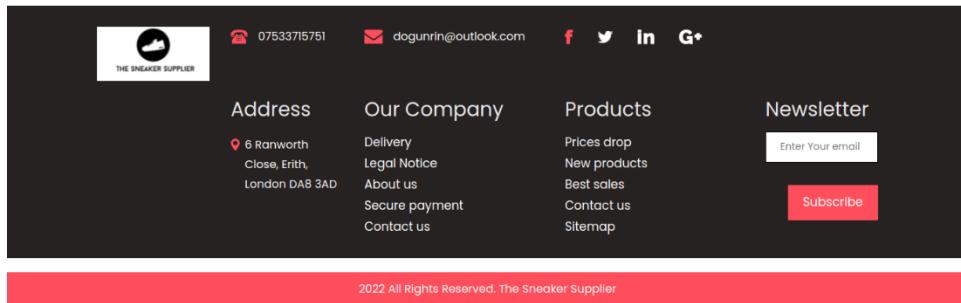
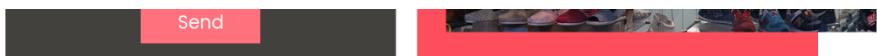
Contact form database id for certain message, different for each unique message for differentiation

As you can see the exact details that I entered the system is what was automatically outputted into database system under its own (contactForm) section and ID. By implementing this feature into my system, I can review and see any feedback messages or problems with the website that the user needs to address by them using this contact form and then I can check by logging into firebase and looking at this real-time database. Note that this is a separate system/database from my user's database, meaning I had to create another database using firebase and then initialize it and link it again into my code, then code all new functions and then connect it to my contact form. This was extremely complicated to do the first time but having successfully done it the first time it was easier

this time around, as I used the creation and implementation of the user database as a template. This is another example of the use of computational thinking when coding up my solution.

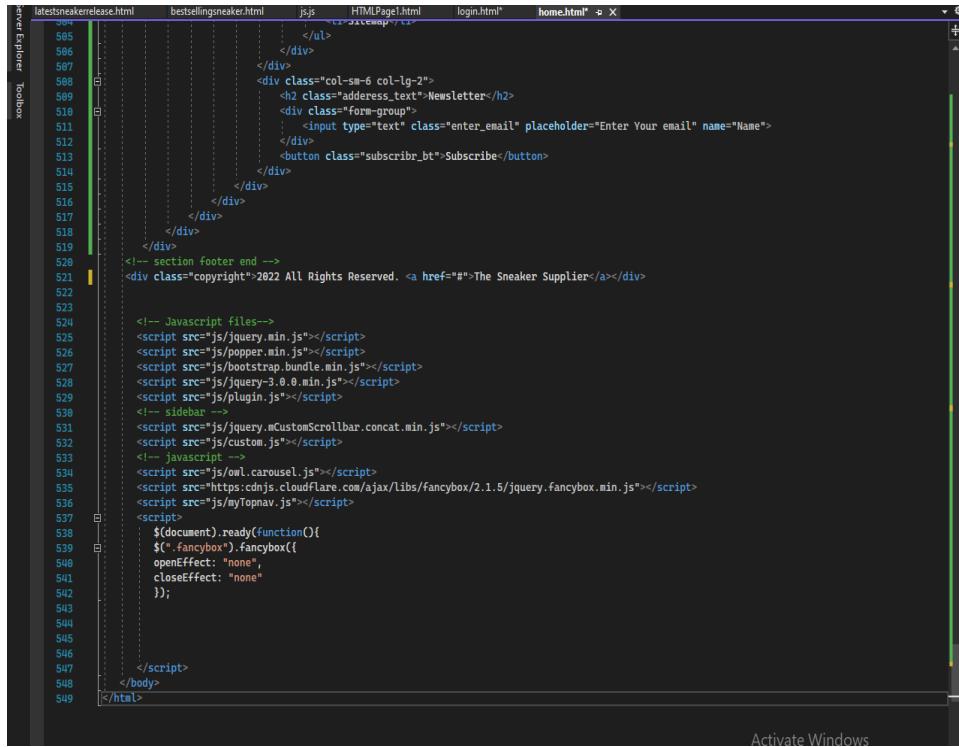
Footer

Finally, after scrolling past the contact section that has now been fully implemented on the home page, the last section that concludes this page and concludes all pages on my website is in the footer. It is included on all pages and is consistent throughout the website and it includes my website logo, contact details, and the HQ (my home address). It also includes all social media handles that would link to the social media pages of my company if they existed. There is also a newsletter section but is no implemented, everything on the footer page is just for design so it can add to the overall look of the website. Below is the picture and code for my footer section:



```
<!-- contact_section_end -->
<div class="section_footer">
<div class="container">
<div class="mail_section">
<div class="row">
<div class="col-sm-6 col-lg-2">
<div><a href="#"></a></div>
</div>
<div class="col-sm-6 col-lg-2">
<div><span class="map_text">07533715751</span></div>
</div>
<div class="col-sm-6 col-lg-3">
<div><span class="map_text">dogunrin@outlook.com</span></div>
</div>
<div class="col-sm-6 col-lg-3">
<div class="social_icon">
<ul>
<li><a href="#"></a></li>
<li><a href="#"></a></li>
<li><a href="#"></a></li>
<li><a href="#"></a></li>
</ul>
</div>
</div>
<div class="col-sm-2"></div>
</div>
</div>
<div class="row">
<div class="col-sm-4 col-lg-2">
<p class="dummy_text"> </p>
</div>
<div class="col-sm-4 col-lg-2">
<h2 class="shop_text">Address </h2>
<div><span class="pet_text">6 Ranworth Close, Erith, London DA8 3AD</span>
</div>
<div class="col-sm-4 col-md-6 col-lg-3">
<h2 class="shop_text">Our Company </h2>
<div class="delivery_text">
<ul>
<li>Delivery</li>
<li>Legal Notice</li>
<li>About us</li>
<li>Secure payment</li>
<li>Contact us</li>
</ul>
</div>
</div>
<div class="col-sm-6 col-lg-3">
```

JavaScript files for the home page implemented at the end.



The screenshot shows the Visual Studio code editor with the 'home.html' file open. The code is a combination of HTML and JavaScript. It includes sections for a newsletter sign-up form, a footer with copyright information, and a list of external JavaScript files. A message at the bottom states 'no issues found.' The code is well-structured with proper indentation and line numbers on the left side.

```
504      latestsneakerrelease.html    bestsellingsneaker.html    js.js      HTMLPage1.html    login.html    home.html*  X
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
```

Activate Windows

As you can see, after editing all mistakes in code with the help of the IDE in the visual base studio I was able to code up the entire home page without any errors and with all required features working, as seen at the bottom of the page it says, 'no issues found.' Thus, this concludes the end of my home page, and I can deem it a success.

Below you can view all the CSS for my home page designs which will be the same consistent CSS file that I will use for every page on my website:

```
contact.html contact1.js register.html style.css* experiemnt.html latestsneakerrelease.html bestse
4  /*
5
6  @import url('https://fonts.googleapis.com/css?family=Rajdhani:300,400,500,600,700');
7  @import url('https://fonts.googleapis.com/css?family=Poppins:100,100i,200,200i,300,300i,400,400i,500,500i,600,600i,700,700i,800,800i,900,900i');
8
9  /*----- 1) font-family: 'Rajdhani', sans-serif;
10 [----- 2) font-family: 'Poppins', sans-serif;
11 [----- *****/
12
13
14  /*
15
16  @import url/animate.min.css);
17  @import url(normalize.css);
18  @import url(icomoon.css);
19  @import url(css/font-awesome.min.css);
20  @import url(meannmenu.css);
21  @import url/owl.carousel.min.css);
22  @import url(swiper.min.css);
23  @import url/slick.css);
24  @import url/jquery.fancybox.min.css);
25  @import url/jquery-ui.css);
26  @import url(nice-select.css);
27
28  /*
29
30  * {
31  [----- box-sizing: border-box !important;
32  [----- transition: ease all 0.5s;
33  }
34
35  html {
36  [----- scroll-behavior: smooth;
37  }
38
39  body {
40  [----- color: #666666;
41  [----- font-size: 14px;
42  [----- font-family: Poppins;
43  [----- line-height: 1.88857;
44  [----- font-weight: normal;
45  [----- overflow-x: hidden;
46  }
47
48  a {
49  [----- color: #1f1f1f;
50  [----- text-decoration: none !important;
51  [----- outline: none !important;
52  [----- -webkit-transition: all .3s ease-in-out;
53  [----- -moz-transition: all .3s ease-in-out;
54  [----- -ms-transition: all .3s ease-in-out;
55  [----- -o-transition: all .3s ease-in-out;
56  [----- transition: all .3s ease-in-out;
57  }
```

```
contact.html contact1.js register.html style.css* experiemnt.html latestsneakerrelease.html bestse
58
59  h1,
60  h2,
61  h3,
62  h4,
63  h5,
64  h6 {
65  [----- letter-spacing: 0;
66  [----- font-weight: normal;
67  [----- position: relative;
68  [----- padding: 0 0 10px 0;
69  [----- font-weight: normal;
70  [----- line-height: normal;
71  [----- color: #111111;
72  [----- margin: 0;
73
74
75  h1 {
76  [----- font-size: 24px;
77
78
79  h2 {
80  [----- font-size: 22px;
81
82
83  h3 {
84  [----- font-size: 18px;
85
86
87  h4 {
88  [----- font-size: 16px;
89
90
91  h5 {
92  [----- font-size: 14px;
93
94
95  h6 {
96  [----- font-size: 13px;
97
98
99  *,
100  *::after,
101  *::before {
102  [----- -webkit-box-sizing: border-box;
103  [----- -moz-box-sizing: border-box;
104  [----- box-sizing: border-box;
105
106
107  h1 a,
108  h2 a,
109  h3 a,
110  h4 a,
```

```
contact.html      contact.js      registr.html      style.css* ✎ X experiemnt.html      latestsneakerrelease.html      bestsellingsneaker.html
186 h1 a,
188 h2 a,
190 h3 a,
192 h4 a,
194 h5 a,
196 h6 a {
197   color: #212121;
198   text-decoration: none !important;
199   opacity: 1
200 }
201
202 button:focus {
203   outline: none;
204 }
205
206 ul,
207 li,
208 ol {
209   margin: 0px;
210   padding: 0px;
211   list-style: none;
212 }
213
214 p {
215   margin: 20px;
216   font-weight: 300;
217   font-size: 15px;
218   line-height: 24px;
219 }
220
221 a {
222   color: #222222;
223   text-decoration: none;
224   outline: none !important;
225 }
226
227 a,
228 .btn {
229   text-decoration: none !important;
230   outline: none !important;
231   -webkit-transition: all .3s ease-in-out;
232   -moz-transition: all .3s ease-in-out;
233   -ms-transition: all .3s ease-in-out;
234   -o-transition: all .3s ease-in-out;
235   transition: all .3s ease-in-out;
236 }
237
238 img {
239   max-width: 100%;
240   height: auto;
241 }
```

```
contact.html      contact.js      registr.html      style.css* ✎ X experiemnt.html      latestsneakerrelease.html      bestsellingsneaker.html
157 []
158
159 :focus {
160   outline: 0;
161 }
162
163 .paddind_bottom_0 {
164   padding-bottom: 0 !important;
165 }
166
167 .btn-custom {
168   margin-top: 20px;
169   background-color: transparent !important;
170   border: 2px solid #ddd;
171   padding: 12px 40px;
172   font-size: 16px;
173 }
174
175 .load {
176   font-size: 18px;
177   line-height: 30px;
178   color: #767676;
179   margin: 0;
180   padding: 0;
181 }
182
183 .form-control:focus {
184   border-color: #fffffe !important;
185   box-shadow: 0 0 0 .2rem rgba(255, 255, 255, .25);
186 }
187
188 .navbar-form input {
189   border: none !important;
190 }
191
192 .badge {
193   font-weight: 500;
194 }
195
196 .blockquote {
197   margin: 20px 0 20px;
198   padding: 30px;
199 }
200
201 button {
202   border: 0;
203   margin: 0;
204   padding: 0;
205   cursor: pointer;
206 }
207
208 .full {
209   float: left;
210 }
```

```
ntact.html contact.js register.html style.css* experiemnt.html latestsneakerrelease.html bestsellingsneaker.html
208 full {
209   float: left;
210   width: 100%;
211 }
212
213 .layout_padding {
214   padding-top: 90px;
215   padding-bottom: 90px;
216 }
217
218 .header_section{
219   width: 100%;
220   float: left;
221   background-image: url(./images/banner.png);
222   height: auto;
223   padding: 20px 0px;
224   background-size: 100%;
225   background-repeat: no-repeat;
226 }
227
228 .padding_0{ width: 100%; float: left; }
229
230 .mens_taital{ width: 100%; float: left; color: #908d92; font-size: 14pt; }
231 .mens_taital_2 {
232   width: 100%;
233   float: left;
234   color: #908d92;
235   font-size: 14pt;
236   margin-top: 150px;
237 }
238 .page_no {
239   width: 28%;
240   /* margin: 0 auto; */
241   text-align: center;
242   color: #fffff;
243   border: 2px solid #8d8791;
244   border-radius: 30px;
245   margin-top: 150px;
246   padding-bottom: 10px;
247   display: inline-block;
248   padding-top: 12px;
249   margin-left: 30px;
250 }
251 .bg-light {
252   background-color: transparent !important;
253 }
254
255 .navbar-light .navbar-nav .nav-link {
256   color: #fff;
257   font-size: 16pt;
258 }
259
260 .navbar-light .navbar-nav .nav-link:focus, .navbar-light .navbar-nav .nav-link:hover {
261   color: #a5c2e0;

```

```
ntact.html contact.js register.html style.css* experiemnt.html latestsneakerrelease.html bestsellingsneaker.html log
259
260 .navbar-light .navbar-nav .nav-link:focus, .navbar-light .navbar-nav .nav-link:hover {
261   color: #db5660;
262 }
263 .navbar-expand-lg .navbar-nav .nav-link {
264   padding-right: 0.9rem;
265   padding-left: 1.6rem;
266 }
267
268
269 .last{ padding-left: 30px !important; }
270 .banner_text{
271   width: 100%;
272   float: left;
273   font-size: 30pt;
274   color: #db5660;
275 }
276
277 .mens_text{
278   width: 100%;
279   float: left;
280   font-size: 39pt;
281   color: #fdfcfc;
282   line-height: 40px;
283 }
284 .lorem_text{
285   width: 100%;
286   float: left;
287   font-size: 13pt;
288   color: #fefefe;
289   margin-left: 0px;
290 }
291
292 .buy_bt{
293   width: 32%;
294   height: 50px;
295   color: #fffff;
296   background-color: #ff4e5b;
297   color: #fffff;
298   font-size: 16pt;
299 }
300
301 .more_bt{
302   width: 32%;
303   height: 50px;
304   color: #fffff;
305   background-color: #fffff;
306   color: #ff4e5b;
307   font-size: 16pt;
308   margin-left: 25px;
309 }
310
311 .banner_taital{
312   width: 100%;
313   height: 100%;
314   background-color: #fffff;
315   background-position: center;
316   background-size: cover;
317   background-repeat: no-repeat;
318   position: absolute;
319   top: 0px;
320   left: 0px;
321   z-index: 1000;
322 }
323
324 .mens_taital{
325   width: 100%;
326   height: 100%;
327   background-color: #fffff;
328   background-position: center;
329   background-size: cover;
330   background-repeat: no-repeat;
331 }
332
333 .lorem_taital{
334   width: 100%;
335   height: 100%;
336   background-color: #fffff;
337   background-position: center;
338   background-size: cover;
339   background-repeat: no-repeat;
340 }
341
342 .buy_bt{
343   width: 32%;
344   height: 50px;
345   color: #fffff;
346   background-color: #ff4e5b;
347   color: #fffff;
348   font-size: 16pt;
349 }
350
351 .more_bt{
352   width: 32%;
353   height: 50px;
354   color: #fffff;
355   background-color: #fffff;
356   color: #ff4e5b;
357   font-size: 16pt;
358   margin-left: 25px;
359 }
360
361 .mens_taital{
362   width: 100%;
363   height: 100%;
364   background-color: #fffff;
365   background-position: center;
366   background-size: cover;
367   background-repeat: no-repeat;
368 }
369
370 .lorem_taital{
371   width: 100%;
372   height: 100%;
373   background-color: #fffff;
374   background-position: center;
375   background-size: cover;
376   background-repeat: no-repeat;
377 }
378
379 .buy_bt{
380   width: 32%;
381   height: 50px;
382   color: #fffff;
383   background-color: #ff4e5b;
384   color: #fffff;
385   font-size: 16pt;
386 }
387
388 .more_bt{
389   width: 32%;
390   height: 50px;
391   color: #fffff;
392   background-color: #fffff;
393   color: #ff4e5b;
394   font-size: 16pt;
395   margin-left: 25px;
396 }
397
398 .mens_taital{
399   width: 100%;
400   height: 100%;
401   background-color: #fffff;
402   background-position: center;
403   background-size: cover;
404   background-repeat: no-repeat;
405 }
406
407 .lorem_taital{
408   width: 100%;
409   height: 100%;
410   background-color: #fffff;
411   background-position: center;
412   background-size: cover;
413   background-repeat: no-repeat;
414 }
415
416 .buy_bt{
417   width: 32%;
418   height: 50px;
419   color: #fffff;
420   background-color: #ff4e5b;
421   color: #fffff;
422   font-size: 16pt;
423 }
424
425 .more_bt{
426   width: 32%;
427   height: 50px;
428   color: #fffff;
429   background-color: #fffff;
430   color: #ff4e5b;
431   font-size: 16pt;
432   margin-left: 25px;
433 }
434
435 .mens_taital{
436   width: 100%;
437   height: 100%;
438   background-color: #fffff;
439   background-position: center;
440   background-size: cover;
441   background-repeat: no-repeat;
442 }
443
444 .lorem_taital{
445   width: 100%;
446   height: 100%;
447   background-color: #fffff;
448   background-position: center;
449   background-size: cover;
450   background-repeat: no-repeat;
451 }
452
453 .buy_bt{
454   width: 32%;
455   height: 50px;
456   color: #fffff;
457   background-color: #ff4e5b;
458   color: #fffff;
459   font-size: 16pt;
460 }
461
462 .more_bt{
463   width: 32%;
464   height: 50px;
465   color: #fffff;
466   background-color: #fffff;
467   color: #ff4e5b;
468   font-size: 16pt;
469   margin-left: 25px;
470 }
471
472 .mens_taital{
473   width: 100%;
474   height: 100%;
475   background-color: #fffff;
476   background-position: center;
477   background-size: cover;
478   background-repeat: no-repeat;
479 }
480
481 .lorem_taital{
482   width: 100%;
483   height: 100%;
484   background-color: #fffff;
485   background-position: center;
486   background-size: cover;
487   background-repeat: no-repeat;
488 }
489
490 .buy_bt{
491   width: 32%;
492   height: 50px;
493   color: #fffff;
494   background-color: #ff4e5b;
495   color: #fffff;
496   font-size: 16pt;
497 }
498
499 .more_bt{
500   width: 32%;
501   height: 50px;
502   color: #fffff;
503   background-color: #fffff;
504   color: #ff4e5b;
505   font-size: 16pt;
506   margin-left: 25px;
507 }
508
509 .mens_taital{
510   width: 100%;
511   height: 100%;
512   background-color: #fffff;
513   background-position: center;
514   background-size: cover;
515   background-repeat: no-repeat;
516 }
517
518 .lorem_taital{
519   width: 100%;
520   height: 100%;
521   background-color: #fffff;
522   background-position: center;
523   background-size: cover;
524   background-repeat: no-repeat;
525 }
526
527 .buy_bt{
528   width: 32%;
529   height: 50px;
530   color: #fffff;
531   background-color: #ff4e5b;
532   color: #fffff;
533   font-size: 16pt;
534 }
535
536 .more_bt{
537   width: 32%;
538   height: 50px;
539   color: #fffff;
540   background-color: #fffff;
541   color: #ff4e5b;
542   font-size: 16pt;
543   margin-left: 25px;
544 }
545
546 .mens_taital{
547   width: 100%;
548   height: 100%;
549   background-color: #fffff;
550   background-position: center;
551   background-size: cover;
552   background-repeat: no-repeat;
553 }
554
555 .lorem_taital{
556   width: 100%;
557   height: 100%;
558   background-color: #fffff;
559   background-position: center;
560   background-size: cover;
561   background-repeat: no-repeat;
562 }
563
564 .buy_bt{
565   width: 32%;
566   height: 50px;
567   color: #fffff;
568   background-color: #ff4e5b;
569   color: #fffff;
570   font-size: 16pt;
571 }
572
573 .more_bt{
574   width: 32%;
575   height: 50px;
576   color: #fffff;
577   background-color: #fffff;
578   color: #ff4e5b;
579   font-size: 16pt;
580   margin-left: 25px;
581 }
582
583 .mens_taital{
584   width: 100%;
585   height: 100%;
586   background-color: #fffff;
587   background-position: center;
588   background-size: cover;
589   background-repeat: no-repeat;
590 }
591
592 .lorem_taital{
593   width: 100%;
594   height: 100%;
595   background-color: #fffff;
596   background-position: center;
597   background-size: cover;
598   background-repeat: no-repeat;
599 }
600
601 .buy_bt{
602   width: 32%;
603   height: 50px;
604   color: #fffff;
605   background-color: #ff4e5b;
606   color: #fffff;
607   font-size: 16pt;
608 }
609
610 .more_bt{
611   width: 32%;
612   height: 50px;
613   color: #fffff;
614   background-color: #fffff;
615   color: #ff4e5b;
616   font-size: 16pt;
617   margin-left: 25px;
618 }
619
620 .mens_taital{
621   width: 100%;
622   height: 100%;
623   background-color: #fffff;
624   background-position: center;
625   background-size: cover;
626   background-repeat: no-repeat;
627 }
628
629 .lorem_taital{
630   width: 100%;
631   height: 100%;
632   background-color: #fffff;
633   background-position: center;
634   background-size: cover;
635   background-repeat: no-repeat;
636 }
637
638 .buy_bt{
639   width: 32%;
640   height: 50px;
641   color: #fffff;
642   background-color: #ff4e5b;
643   color: #fffff;
644   font-size: 16pt;
645 }
646
647 .more_bt{
648   width: 32%;
649   height: 50px;
650   color: #fffff;
651   background-color: #fffff;
652   color: #ff4e5b;
653   font-size: 16pt;
654   margin-left: 25px;
655 }
656
657 .mens_taital{
658   width: 100%;
659   height: 100%;
660   background-color: #fffff;
661   background-position: center;
662   background-size: cover;
663   background-repeat: no-repeat;
664 }
665
666 .lorem_taital{
667   width: 100%;
668   height: 100%;
669   background-color: #fffff;
670   background-position: center;
671   background-size: cover;
672   background-repeat: no-repeat;
673 }
674
675 .buy_bt{
676   width: 32%;
677   height: 50px;
678   color: #fffff;
679   background-color: #ff4e5b;
680   color: #fffff;
681   font-size: 16pt;
682 }
683
684 .more_bt{
685   width: 32%;
686   height: 50px;
687   color: #fffff;
688   background-color: #fffff;
689   color: #ff4e5b;
690   font-size: 16pt;
691   margin-left: 25px;
692 }
693
694 .mens_taital{
695   width: 100%;
696   height: 100%;
697   background-color: #fffff;
698   background-position: center;
699   background-size: cover;
700   background-repeat: no-repeat;
701 }
702
703 .lorem_taital{
704   width: 100%;
705   height: 100%;
706   background-color: #fffff;
707   background-position: center;
708   background-size: cover;
709   background-repeat: no-repeat;
710 }
711
712 .buy_bt{
713   width: 32%;
714   height: 50px;
715   color: #fffff;
716   background-color: #ff4e5b;
717   color: #fffff;
718   font-size: 16pt;
719 }
720
721 .more_bt{
722   width: 32%;
723   height: 50px;
724   color: #fffff;
725   background-color: #fffff;
726   color: #ff4e5b;
727   font-size: 16pt;
728   margin-left: 25px;
729 }
730
731 .mens_taital{
732   width: 100%;
733   height: 100%;
734   background-color: #fffff;
735   background-position: center;
736   background-size: cover;
737   background-repeat: no-repeat;
738 }
739
740 .lorem_taital{
741   width: 100%;
742   height: 100%;
743   background-color: #fffff;
744   background-position: center;
745   background-size: cover;
746   background-repeat: no-repeat;
747 }
748
749 .buy_bt{
750   width: 32%;
751   height: 50px;
752   color: #fffff;
753   background-color: #ff4e5b;
754   color: #fffff;
755   font-size: 16pt;
756 }
757
758 .more_bt{
759   width: 32%;
760   height: 50px;
761   color: #fffff;
762   background-color: #fffff;
763   color: #ff4e5b;
764   font-size: 16pt;
765   margin-left: 25px;
766 }
767
768 .mens_taital{
769   width: 100%;
770   height: 100%;
771   background-color: #fffff;
772   background-position: center;
773   background-size: cover;
774   background-repeat: no-repeat;
775 }
776
777 .lorem_taital{
778   width: 100%;
779   height: 100%;
780   background-color: #fffff;
781   background-position: center;
782   background-size: cover;
783   background-repeat: no-repeat;
784 }
785
786 .buy_bt{
787   width: 32%;
788   height: 50px;
789   color: #fffff;
790   background-color: #ff4e5b;
791   color: #fffff;
792   font-size: 16pt;
793 }
794
795 .more_bt{
796   width: 32%;
797   height: 50px;
798   color: #fffff;
799   background-color: #fffff;
800   color: #ff4e5b;
801   font-size: 16pt;
802   margin-left: 25px;
803 }
804
805 .mens_taital{
806   width: 100%;
807   height: 100%;
808   background-color: #fffff;
809   background-position: center;
810   background-size: cover;
811   background-repeat: no-repeat;
812 }
813
814 .lorem_taital{
815   width: 100%;
816   height: 100%;
817   background-color: #fffff;
818   background-position: center;
819   background-size: cover;
820   background-repeat: no-repeat;
821 }
822
823 .buy_bt{
824   width: 32%;
825   height: 50px;
826   color: #fffff;
827   background-color: #ff4e5b;
828   color: #fffff;
829   font-size: 16pt;
830 }
831
832 .more_bt{
833   width: 32%;
834   height: 50px;
835   color: #fffff;
836   background-color: #fffff;
837   color: #ff4e5b;
838   font-size: 16pt;
839   margin-left: 25px;
840 }
841
842 .mens_taital{
843   width: 100%;
844   height: 100%;
845   background-color: #fffff;
846   background-position: center;
847   background-size: cover;
848   background-repeat: no-repeat;
849 }
850
851 .lorem_taital{
852   width: 100%;
853   height: 100%;
854   background-color: #fffff;
855   background-position: center;
856   background-size: cover;
857   background-repeat: no-repeat;
858 }
859
860 .buy_bt{
861   width: 32%;
862   height: 50px;
863   color: #fffff;
864   background-color: #ff4e5b;
865   color: #fffff;
866   font-size: 16pt;
867 }
868
869 .more_bt{
870   width: 32%;
871   height: 50px;
872   color: #fffff;
873   background-color: #fffff;
874   color: #ff4e5b;
875   font-size: 16pt;
876   margin-left: 25px;
877 }
878
879 .mens_taital{
880   width: 100%;
881   height: 100%;
882   background-color: #fffff;
883   background-position: center;
884   background-size: cover;
885   background-repeat: no-repeat;
886 }
887
888 .lorem_taital{
889   width: 100%;
890   height: 100%;
891   background-color: #fffff;
892   background-position: center;
893   background-size: cover;
894   background-repeat: no-repeat;
895 }
896
897 .buy_bt{
898   width: 32%;
899   height: 50px;
900   color: #fffff;
901   background-color: #ff4e5b;
902   color: #fffff;
903   font-size: 16pt;
904 }
905
906 .more_bt{
907   width: 32%;
908   height: 50px;
909   color: #fffff;
910   background-color: #fffff;
911   color: #ff4e5b;
912   font-size: 16pt;
913   margin-left: 25px;
914 }
915
916 .mens_taital{
917   width: 100%;
918   height: 100%;
919   background-color: #fffff;
920   background-position: center;
921   background-size: cover;
922   background-repeat: no-repeat;
923 }
924
925 .lorem_taital{
926   width: 100%;
927   height: 100%;
928   background-color: #fffff;
929   background-position: center;
930   background-size: cover;
931   background-repeat: no-repeat;
932 }
933
934 .buy_bt{
935   width: 32%;
936   height: 50px;
937   color: #fffff;
938   background-color: #ff4e5b;
939   color: #fffff;
940   font-size: 16pt;
941 }
942
943 .more_bt{
944   width: 32%;
945   height: 50px;
946   color: #fffff;
947   background-color: #fffff;
948   color: #ff4e5b;
949   font-size: 16pt;
950   margin-left: 25px;
951 }
952
953 .mens_taital{
954   width: 100%;
955   height: 100%;
956   background-color: #fffff;
957   background-position: center;
958   background-size: cover;
959   background-repeat: no-repeat;
960 }
961
962 .lorem_taital{
963   width: 100%;
964   height: 100%;
965   background-color: #fffff;
966   background-position: center;
967   background-size: cover;
968   background-repeat: no-repeat;
969 }
970
971 .buy_bt{
972   width: 32%;
973   height: 50px;
974   color: #fffff;
975   background-color: #ff4e5b;
976   color: #fffff;
977   font-size: 16pt;
978 }
979
980 .more_bt{
981   width: 32%;
982   height: 50px;
983   color: #fffff;
984   background-color: #fffff;
985   color: #ff4e5b;
986   font-size: 16pt;
987   margin-left: 25px;
988 }
989
990 .mens_taital{
991   width: 100%;
992   height: 100%;
993   background-color: #fffff;
994   background-position: center;
995   background-size: cover;
996   background-repeat: no-repeat;
997 }
998
999 .lorem_taital{
1000  width: 100%;
1001  height: 100%;
1002  background-color: #fffff;
1003  background-position: center;
1004  background-size: cover;
1005  background-repeat: no-repeat;
1006 }
1007
1008 .buy_bt{
1009  width: 32%;
1010  height: 50px;
1011  color: #fffff;
1012  background-color: #ff4e5b;
1013  color: #fffff;
1014  font-size: 16pt;
1015 }
1016
1017 .more_bt{
1018  <
```

```
contact.html contact1.js register.html style.css* experiemnt.html latestsneakerrelease.html bestsellingsneaker.html login.html* home.html*
```

```
310
311 .banner_taital{
312     width: 100%;
313     float: left;
314     margin-top: 85px;
315 }
316
317 body{
318     width:100%;
319     height:100%;
320 }
321 .carousel-inner,.carousel,.carousel-item,.container,.fill {
322     height:100%;
323     width:100%;
324     background-position:center center;
325     background-size:cover;
326 }
327 .slide-wrapper{display:inline;}
328 .slide-wrapper .container{padding:0;}
329
330 /*----- vertical bootstrap slider-----*/
331
332 .carousel-inner> .carousel-item.carousel-item-next ,
333 .carousel-inner > .carousel-item.active.carousel-item-right{
334     transform: translate3d(0, 100%, 0); -webkit-transform: translate3d(0, 100%, 0); -ms-transform: translate3d(0, 100%, 0); -moz-transform: translate3d(0, 100%, 0);
335 }
336
337 .carousel-inner > .carousel-item.carousel-item-prev ,
338 .carousel-inner > .carousel-item.active.carousel-item-left{
339     transform: translate3d(0,-100%, 0); -webkit-transform: translate3d(0,-100%, 0); -ms-transform: translate3d(0,-100%, 0); -moz-transform: translate3d(0,-100%, 0);
340 }
341
342 .carousel-inner > .carousel-item.next.carousel-item-left ,
343 .carousel-inner > .carousel-item.carousel-item-prev.carousel-item-right ,
344 .carousel-inner > .carousel-item.active{
345     transform:translate3d(0,0,0); -webkit-transform:translate3d(0,0,0);-ms-transform:translate3d(0,0,0); -moz-transform:translate3d(0,0,0); -o-transform:translate3d(
346
347 }
348
349 /*----- vertical carousel indicators -----*/
350 .carousel-indicators{
351     position: absolute;
352     top: 0;
353     bottom: 0;
354     margin: auto;
355     height: 20px;
356     right: -84px;
357     left: auto;
358     width: auto;
359     -webkit-transform: rotate(90deg);
360     -moz-transform: rotate(90deg);
361     -ms-transform: rotate(90deg);
362     -o-transform: rotate(90deg);
363 }
```

Activate Windows
Go to Settings to activate Windows.

```
contact.html contact1.js register.html style.css* experiemnt.html latestsneakerrelease.html bestsellingsne  
361 | -moz-transform: rotate(90deg);  
362 | -ms-transform: rotate(90deg);  
363 | -o-transform: rotate(90deg);  
364 | transform: rotate(90deg);  
365 |  
366 | .carousel-indicators li{  
367 | display:block; margin-bottom:5px; border:1px solid #00a199;  
368 | margin-bottom: 5px;}  
369 |  
370 | .carousel-indicators li.active{  
371 | margin-bottom:5px; background:#00a199;  
372 |}  
373 |  
374 |  
375 .collection_section{ width: 100%; float: left; padding-bottom: 0px; }  
376 | .new_text{  
377 | width: 100%;  
378 | float: left;  
379 | font-size: 24pt;  
380 | color: #0c0116;  
381 |}  
382 |  
383 | .consectetur_text{  
384 | width: 100%;  
385 | float: left;  
386 | font-size: 14pt;  
387 | color: #0c0116;  
388 | font-weight: 400;  
389 | margin-left: 0px;  
390 | padding-bottom: 30px;  
391 |}  
392 |  
393 |  
394 | .collection_section_2{  
395 | width: 100%;  
396 | float: left;  
397 |}  
398 |  
399 | .about-img {  
400 | width: 100%;  
401 | float: left;  
402 | background-color: #ffffff;  
403 | height: auto;  
404 | margin-left: 64px;  
405 | box-shadow: 0px 0px 10px 0px;  
406 | position: relative;  
407 | z-index: 1;  
408 |}  
409 | .new_bt{  
410 | width: 18%;  
411 | float: left;  
412 | background-color: #000;  
413 | font-size: 14pt;  
414 | text-align: center;  
415 | color: #ffffff;  
416 |  
417 | height: 50px;  
418 | margin-left: -60px;  
419 |  
420 | .shoes-img{  
421 | width: 100%;  
422 | float: left;  
423 | text-align: center;  
424 | padding-top: 40px;  
425 | padding-bottom: 40px;  
426 | margin: 0px;  
427 |}  
428 |  
429 | .sport_text {  
430 | width: 24%;  
431 | text-align: center;  
432 | padding-top: 30px;  
433 | font-size: 16pt;  
434 | color: #0c0116;  
435 | border-bottom: 1px solid#000;  
436 | margin: 0 auto;  
437 |}  
438 | .dolar_text{  
439 | width: 100%;  
440 | float: left;  
441 | text-align:center;  
442 | color: #000;  
443 | font-size: 20pt;  
444 |}  
445 .star.icon{ width: 25%; margin: 0 auto; text-align: center; }  
446 .star.icon ul{ margin: 0px; padding: 0px; }  
447 .star.icon li{ float:left; padding-right:10px; padding-bottom: 25px; }  
448 | .star.icon{  
449 | padding: 7px;  
450 | color: goldenrod;  
451 | background: black;  
452 | font-size: 14px;  
453 | border-radius: 15px;  
454 |}  
455 | .about-img2{  
456 | width: 100%;  
457 | float: left;  
458 | background-image: url(..../images/about-img2.png);  
459 | height: auto;  
460 | box-shadow: 0px 0px 10px 0px;  
461 | margin-left: -50px;  
462 | margin-top: 74px;  
463 |}  
464 |  
465 | .shoes_img{ width: 100%; float: left; padding-top: 30px; }  
466 | .shoes-img2{  
467 | width: 100%;
```

```
Server Explorer contact.html contact1.js registerr.html style.css* experiemnt.html latestsneakerrelease.html bestsellingsne  
415 | color: #ffffff;  
416 | height: 50px;  
417 | margin-left: -60px;  
418 |}  
419 |  
420 | .shoes-img{  
421 | width: 100%;  
422 | float: left;  
423 | text-align: center;  
424 | padding-top: 40px;  
425 | padding-bottom: 40px;  
426 | margin: 0px;  
427 |}  
428 |  
429 | .sport_text {  
430 | width: 24%;  
431 | text-align: center;  
432 | padding-top: 30px;  
433 | font-size: 16pt;  
434 | color: #0c0116;  
435 | border-bottom: 1px solid#000;  
436 | margin: 0 auto;  
437 |}  
438 | .dolar_text{  
439 | width: 100%;  
440 | float: left;  
441 | text-align:center;  
442 | color: #000;  
443 | font-size: 20pt;  
444 |}  
445 .star.icon{ width: 25%; margin: 0 auto; text-align: center; }  
446 .star.icon ul{ margin: 0px; padding: 0px; }  
447 .star.icon li{ float:left; padding-right:10px; padding-bottom: 25px; }  
448 | .star.icon{  
449 | padding: 7px;  
450 | color: goldenrod;  
451 | background: black;  
452 | font-size: 14px;  
453 | border-radius: 15px;  
454 |}  
455 | .about-img2{  
456 | width: 100%;  
457 | float: left;  
458 | background-image: url(..../images/about-img2.png);  
459 | height: auto;  
460 | box-shadow: 0px 0px 10px 0px;  
461 | margin-left: -50px;  
462 | margin-top: 74px;  
463 |}  
464 |  
465 | .shoes_img{ width: 100%; float: left; padding-top: 30px; }  
466 | .shoes-img2{  
467 | width: 100%;
```

```
contact.html      contact.js      registerr.html      style.css*  ▶  experiemnt.html      latestsneakerrelease.html      best
466 .shoes_img{ width: 100%; float: left; padding-top: 30px; }
467
468 .shoes-img2{
469   width: 100%;
470   float: left;
471   text-align: center;
472   padding-top: 100px;
473   padding-bottom: 40px;
474   margin: 0px;
475 }
476
477 .seemore_bt{
478   width: 32%;
479   height: 50px;
480   color: #ffffff;
481   background-color: #fff4e5b;
482   color: #fffffff;
483   font-size: 16pt;
484   margin-top: 35px;
485   margin-bottom: 90px;
486 }
487
488 .collectionn_section_3{ width: 100%; float: left; padding-top: 0px; }
489 .racing_shoes {
490   width: 100%;
491   float: left;
492   background-color: #eaec7e7;
493   height: 70px;
494   padding-top: 70px;
495   background-size: 100%;
496   padding-bottom: 70px;
497 }
498 .shoes-img3{ width: 100%; float: left; text-align: center; }
499
500 .collection_text{
501   width: 100%;
502   float: left;
503   color: #fffffff;
504   text-align: center;
505   font-weight: bold;
506   font-size: 40px;
507   background-color: #db5660;
508   padding: 30px 0px;
509 }
510
511 .sale_text{
512   width: 100%;
513   float: left;
514   font-size: 45pt;
515   color: #ed2540;
516   line-height: 75px;
517 }
518
519 .number_text{
520   width: 100%;
```



```
520   width: 100%;
521   float: left;
522   font-size: 45pt;
523   color: #ed2540;
524   line-height: 75px;
525 }
526
527
528 .seemore{
529   width: 50%;
530   height: 50px;
531   color: #fffffff;
532   background-color: #fff4e5b;
533   color: #fffffff;
534   font-size: 16pt;
535   margin-top: 35px;
536 }
537
538 .gallery_section{
539   width: 100%;
540   float: left;
541   background-color: #fff4e5b;
542   height: auto;
543 }
544
545 .best_shoes{
546   width: 100%;
547   float: left;
548   background-color: #ffffff;
549   height: auto;
550   padding: 20px 20px 20px 0px;
551   margin-top: 15px;
552   margin-bottom: 15px;
553 }
554
555 .best_text{
556   width: 100%;
557   float: left;
558   font-size: 14pt;
559   color: #0000;
560   text-align: center;
561   margin-left: 0px;
562 }
563
564 .shoes_icon{
565   width: 100%;
566   float: left;
567   text-align: center;
568   min-height: 220px;
569 }
570
571 .star_text{
572   width: 100%;
```

```
571 .star_text{  
572     width: 100%;  
573     float: left;  
574 }  
575 .left_part {  
576     width: 60%;  
577     float: left;  
578     padding-left: 15px;  
579     padding-top: 15px;  
580 }  
581 .left_part ul{ margin: 0px; padding: 0px; }  
582 .left_part li{ float: left; padding-left: 10px; }  
583 .right_part{ width: 40%; float: left; }  
584 .shoes_price{  
585     width: 100%;  
586     float: left;  
587     font-size: 22pt;  
588     color: #0f0d10;  
589     text-align: right;  
590 }  
591 .buy_now_bt{  
592     width: 50%;  
593     margin: 0 auto;  
594     text-align: center;  
595 }  
596 .buy_text{  
597     width: 32%;  
598     height: 50px;  
599     color: #ffffff;  
600     background-color: #131111;  
601     color: #ffffff;  
602     font-size: 16pt;  
603     margin-top: 35px;  
604 }  
605 .contact_section{  
606     width: 100%;  
607     float: left;  
608 }  
609 .email_box {  
610     width: 85%;  
611     float: right;  
612     background-color: #14120e;  
613     height: auto;  
614     padding: 20px 20px;  
615     opacity: 0.80;  
616 }  
617 .form-group {  
618     margin-bottom: 2rem;  
619 }  
620 .email_bt {
```

```
621 contact.html    contact1.js    register.html*    style.css*  experiemnt.html    latestsneakerrelease.html  
622  
623 .email_bt {  
624     border: 1px solid #000;  
625     color: #000000;  
626     width: 100%;  
627     height: 55px;  
628     font-size: 22px;  
629     padding: 20px;  
630 }  
631 .message_bt {  
632     border: 1px solid #000;  
633     color: #000000;  
634     width: 100%;  
635     height: 20px;  
636     font-size: 22px;  
637     padding: 70px 20px 10px 20px;  
638 }  
639 .send_btn {  
640     width: 100%;  
641     margin: 0 auto;  
642     text-align: center;  
643 }  
644 .main_bt f {  
645     background: #ff4e5b;  
646     border: none;  
647     width: 170px;  
648     height: 50px;  
649     font-size: 26px;  
650     color: #ffffff;  
651 }  
652 .ram{  
653     width: 100%;  
654     float: left;  
655 }  
656 a:hover {  
657     color: #fff;  
658 }  
659 .shop_banner{  
660     width: 100%;  
661     float: left;  
662     background-image: url(..../images/shop-banner.png);  
663     height: auto;  
664     padding: 70px;  
665 }  
666 .our_shop {  
667     width: 100%;  
668     float: left;  
669     padding-top: 20px;  
670 }
```

```
contact.html      contact.js      register.html*      style.css*  experiemnt.html      latestsneakerrelease.html      bests
55
56
57
58
59
59 Toolbox
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
389 
```

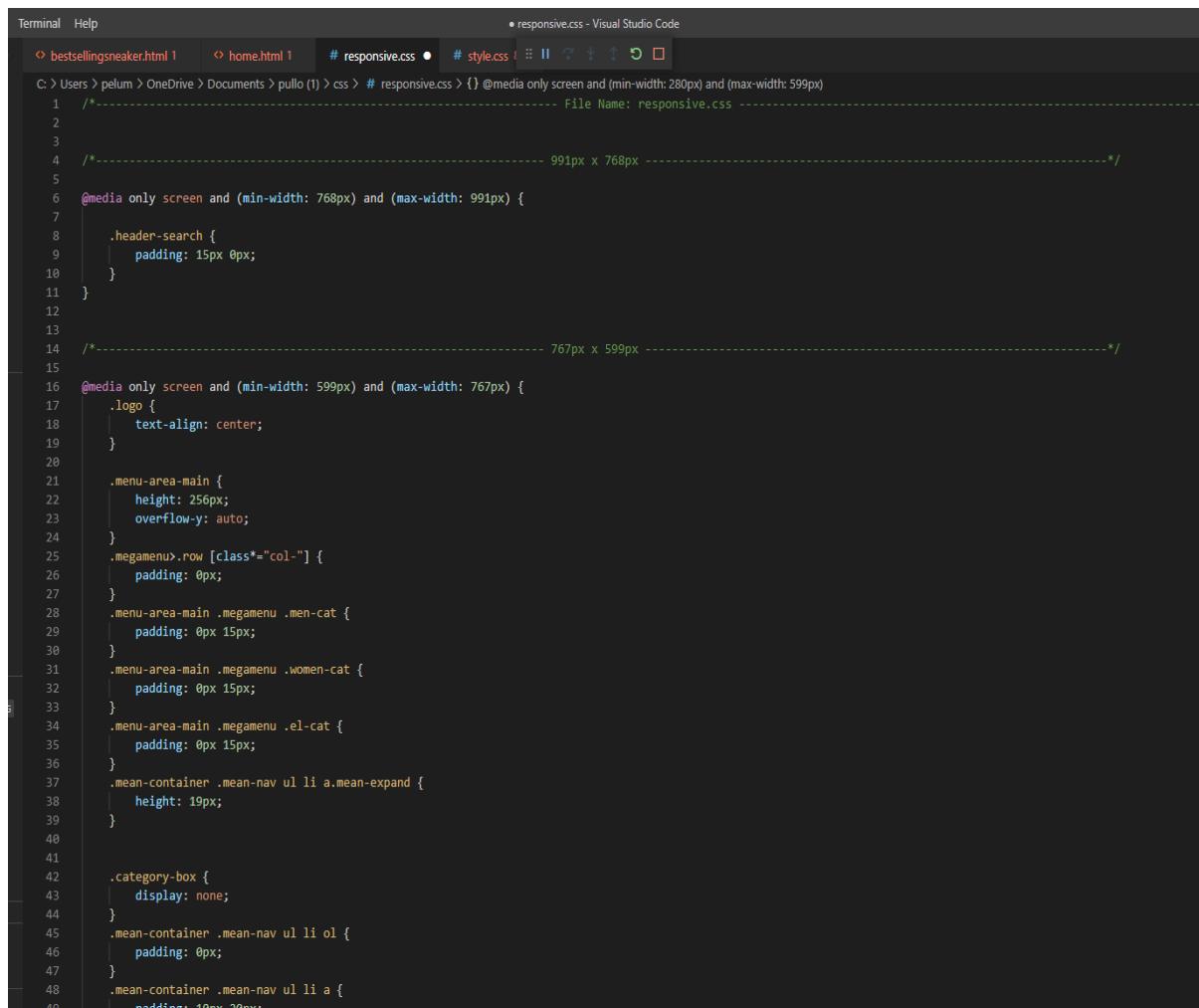
```
< bestsellingsneaker.html 1      # style.css 8 >
C: > Users > pulum > OneDrive > Documents > pullo (1) > css > # style.css > ↵ .adderess_text
726   float: left;
727   font-size: 12pt;
728   color: ■#ffffff;
729   margin-left: 0px;
730   margin-top: 55px;
731   line-height: 25px;
732 }
733 .image-icon {
734   width: 100%;
735   float: left;
736   padding-top: 6px;
737 }
738 .pet_text {
739   font-size: 12pt;
740   color: ■#ffffff;
741   padding-left: 16px;
742   display: flex;
743   top: -26px;
744   position: relative;
745   padding-left: 20px;
746 }
747 .adderess_text {
748   width: 100%;
749   float: left;
750   font-size: 28px;
751   color: ■#ebebeb;
752 }
753 .copyright {
754   width: 100%;
755   float: left;
756   font-size: 12pt;
757   color: ■#ffffff;
758   text-align: center;
759   padding: 10px 0px;
760   background-color: ■#ff4e5b;
761   font-family: Poppins;
762 }
763 .copyright a{ color: ■#ffffff; }
764 .copyright a:hover{ color: □#000; }
765
766 .map_text {
767   padding-left: 15px;
768 }
769
770 .delivery_text{
771   width: 100%;
772   float: left;
773 }
```

```
bestsellingsneaker.html 1 # style.css 8 x
C: > Users > pelum > OneDrive > Documents > pullo (1) > css > # style.css > ↗ .address_text
771   .delivery_text{
772     width: 100%;
773     float: left;
774     font-size: 17px;
775     color: ■#fffff;
776   }
777
778   .enter_email{
779     border: 1px solid □#000;
780     color: □#000000;
781     width: 100%;
782     height: 44px;
783     font-size: 14px;
784     padding: 20px;
785   }
786
787   .subscribe_bt {
788     width: 80%;
789     float: right;
790     background-color: ■#ff4e5b;
791     font-size: 13pt;
792     height: 50px;
793     color: ■#fffff;
794   }
795
796
797   .about_main {
798     width: 100%;
799     float: left;
800     padding-bottom: 30px;
801   }
802
803   .header_main {
804     width: 100%;
805     height: auto;
806     background: □#0c0116;
807   }
808
809   .shoes_icon img:hover{
810     transform: scale(1.1);
811   }
812
813   .button1{
814     padding: 6px;
815     border-radius: 50%;
816     background: ■pink;
817     cursor: pointer;
818   }
819
820   .toggle-fa-heart-o{
821     background: ■orangered;
822     color: ■white;
823   }
824
825
826   .alert{
827     width: 100%;
828     background: ■rgb(0, 255, 106);
829     padding: 10px 20px;
830     border-radius: 5px;
831     text-align:center;
832     font-size: 18px;
833     font-weight:900;
834     display:none;
835   }
836 }
```

```
bestsellingsneaker.html 1 # style.css 8 x
C: > Users > pelum > OneDrive > Documents > pullo (1) > css > # style.css > ↗ .address_text
803   .header_main {
804     width: 100%;
805     height: auto;
806     background: □#0c0116;
807   }
808
809   .shoes_icon img:hover{
810     transform: scale(1.1);
811   }
812
813   .button1{
814     padding: 6px;
815     border-radius: 50%;
816     background: ■pink;
817     cursor: pointer;
818   }
819
820   .toggle-fa-heart-o{
821     background: ■orangered;
822     color: ■white;
823   }
824
825
826   .alert{
827     width: 100%;
828     background: ■rgb(0, 255, 106);
829     padding: 10px 20px;
830     border-radius: 5px;
831     text-align:center;
832     font-size: 18px;
833     font-weight:900;
834     display:none;
835   }
836 }
```

This style.css file includes most of the designs for my website pages. This concludes most of the CSS code for my whole website. Another main CSS file for my pages and for the home page especially is

the responsive CSS file saved as responsive.css which is referenced again at the start of this HTML page. The CSS for this is displayed in all the screenshots below:



The screenshot shows the Visual Studio Code interface with the responsive.css file open. The code editor displays CSS rules for a website's responsive design, specifically targeting screens between 768px and 991px wide. The file includes media queries for different screen widths, styles for header search, logo, menu areas, megamenus, and category boxes, and various padding and height specifications.

```
/*----- 991px x 768px -----*/
@media only screen and (min-width: 768px) and (max-width: 991px) {
    .header-search {
        padding: 15px 0px;
    }
}

/*----- 767px x 599px -----*/
@media only screen and (min-width: 599px) and (max-width: 767px) {
    .logo {
        text-align: center;
    }

    .menu-area-main {
        height: 256px;
        overflow-y: auto;
    }

    .megamenu>.row [class*="col-"] {
        padding: 0px;
    }

    .menu-area-main .megamenu .men-cat {
        padding: 0px 15px;
    }

    .menu-area-main .megamenu .women-cat {
        padding: 0px 15px;
    }

    .menu-area-main .megamenu .el-cat {
        padding: 0px 15px;
    }

    .mean-container .mean-nav ul li a.mean-expand {
        height: 19px;
    }

    .category-box {
        display: none;
    }

    .mean-container .mean-nav ul li ol {
        padding: 0px;
    }

    .mean-container .mean-nav ul li a {
        padding: 10px 20px;
    }
}
```

```
> Users > pelum > OneDrive > Documents > puluo() > css > # responsive.css > () @media only screen and (min-width: 280px) and (max-width: 599px) > .menu-area-main .megamenu .men-cat
45     .mean-container .mean-nav ul li ol {
46         padding: 0px;
47     }
48     .mean-container .mean-nav ul li a {
49         padding: 10px 20px;
50         width: 94.8%;
51     }
52     .mean-container .mean-nav ul li li a {
53         width: 92%;
54         padding: 1em 4%;
55     }
56     .mean-container .mean-nav ul li li li a {
57         width: 100%;
58     }
59     .header-search {
60         padding: 15px 0px;
61     }
62     #collapseFilter.d-md-block {
63         padding: 30px 0px;
64     }
65 }
66
67
68 /*----- 599px x 280px -----*/
69
70 @media only screen and (min-width: 280px) and (max-width: 599px) {
71     .megamenu .row [class*="col-"] {
72         padding: 0px;
73     }
74     .menu-area-main .megamenu .men-cat {
75         padding: 0px 15px;
76     }
77     .menu-area-main .megamenu .women-cat {
78         padding: 0px 15px;
79     }
80     .menu-area-main .megamenu .el-cat {
81         padding: 0px 15px;
82     }
83     .mean-container .mean-nav ul li a {
84         padding: 1em 4%;
85         width: 92%;
86     }
87     .mean-container .mean-nav ul li li a {
88         width: 90%;
89         padding: 1em 5%;
90     }
91     .mean-container .sub-full.megamenu-categories ol li a {
92         padding: 5px 0px;
93     }
```

```
90     padding: 1em 5%;  
91 }  
92 .mean-container .sub-full.megamenu-categories ol li a {  
93     padding: 5px 0px;  
94     text-transform: capitalize;  
95     width: 100%;  
96 }  
97 .megamenu .sub-full.megamenu-categories .womens-box .banner-up-text a {  
98     width: auto;  
99     border: none;  
100    float: none;  
101 }  
102 .menu-area_main {  
103     height: 45px;  
104     overflow-y: auto;  
105 }  
106 .mean-container .mean-nav ul li a.mean-expand {  
107     top: 0;  
108 }  
109 .banner_text { font-size: 20pt; }  
110 .mens_text { font-size: 25pt; }  
111 .lorem_text { display: none; }  
112 .buy_bt { display: none; }  
113 .more_bt { display: none; }  
114 .star_icon { width: 23%; }  
115 .seemore_bt { width: 55%; }  
116 .sport_text { width: 50%; }  
117 .left_part { padding-left: 10px; }  
118 .massage_bt { height: 160px; padding: 55px 20px 10px 20px; }  
119  
120  
121  
122  
123  
124  
125 }  
126  
127 @media (min-width: 992px) and (max-width: 1199px) {  
128  
129     .titlepage h2::after { width: 58%; }  
130     .about-box .titlepage h2::after {  
131         width: 52%;  
132     right: -22px;  
133     }  
134     .Nursery-img .text-box h3 { padding: 111px 50px; }  
135     .contact .titlepage h2::after {  
136         left: -650px;  
137         width: 30%;  
138     }  
139 }
```

```
bestsellingsneaker.html 1      home.html 1      # responsive.css ●  # style.css ⏺ II
C: > Users > pelum > OneDrive > Documents > pullo (1) > css > # responsive.css > {} @media only screen and (max-width: 768px) {
135     width: 30%; }
138 }
139 .sporrt_text { margin-top: 0px; }
140 #main_slider a.carousel-control-prev {
141     display: none;
142 }
143 #main_slider a.carousel-control-next {
144     display: none;
145 }
146
147 .navbar-light .navbar-nav .nav-link { font-size: 13pt; }
148 .navbar-expand-lg .navbar-nav .nav-link {
149     padding-right: 0.5rem;
150     padding-left: 1.5rem;
151 }
152 .banner_text { font-size: 28pt; }
153 .mens_text { font-size: 35pt; }
154 .lorem_text { font-size: 10pt; }
155 .buy_bt { width: 40%; }
156 .more_bt { width: 40%; }
157 .page_no { width: 40%; }
158 .mens_taital_2 { margin-top: 100px; }
159 .sport_text { width: 30%; }
160 .star_icon { width: 30%; }
161 .footer-logo { font-size: 10pt; }
162 .map_text { padding-left: 9px; }
163 .mail_section li { padding-right: 28px; }
164 .dummy_text { font-size: 10pt; }
165 .pet_text { font-size: 10pt; }
166 .delivery_text { font-size: 10pt; }
167 .shop_text { font-size: 20px; }
168 .address_text { font-size: 20px; }
169 .dummy_text { margin-top: 45px; }
170 .email_box { width: 95%; padding: 20px 0px; }
171 .last{ padding-left: 12px !important; }
172
173
174
175 }
176
177 @media (min-width: 768px) and (max-width: 991px) {
178     .main-menu ul > li a {padding: 7px 6px ; }
179     .main-menu ul>li a { font-size: 14px; }
180
181     .banner-main .carousel-caption h1 {
182         font-size: 36px;
183         padding-top: 0px;
184
185 }
```

```
bestsellingsneaker.html 1      home.html 1      # responsive.css ●  # style.css ⏺ II
C: > Users > pelum > OneDrive > Documents > pullo (1) > css > # responsive.css > {} @media only screen and (max-width: 768px) {
195     display: none;
196 }
197 .btn_main { margin-top: 58px; width: 65%; }
198 #main_slider a.carousel-control-next {
199     display: none;
200 }
201 .sporrt_text { margin-top: 0px; font-size: 24px; }
202 .email_btn { padding: 41px; }
203 .btn-primary {max-width: 144px; }
204 .titlepage h2::after {width: 79%; }
205 .about_box .titlepage h2::after {width: 73%; }
206     right: -88px;
207 }
208 .contact .titlepage h2::after {
209     left: -403px;
210     width: 41%; }
211 .padding-right {
212     padding-right: 15px;
213 }
214 .padding-left {
215     padding-left: 15px;
216 }
217
218 .banner_text { font-size: 20pt; }
219 .mens_text { font-size: 20pt; }
220 .lorem_text { display: none; }
221 .buy_bt { width: 45%; font-size: 14pt; }
222 .more_bt { width: 45%; font-size: 14pt; margin-left: 10px; }
223 .mens_taital { display: none; }
224 .page_no { display: none; }
225 .mens_taital_2 { display: none; }
226 .new_bt { width: 30%; }
227 .sport_text { width: 37%; }
228 .star_icon { width: 40%; }
229 .seemore_bt { width: 45%; }
230 .sale_text { font-size: 36pt; }
231 .seemore { width: 70%; }
232 .left_part { padding-left: 0px; }
233 .left_part li { padding-left: 0px; }
234 .shoes_price { font-size: 18pt; }
235 .buy_text { width: 45%; }
236 .shop_banner { padding: 47px; }
237
238 .navbar-light .navbar-toggler {
239     color: #ffffff;
240     border-color: #fff;
241     background: #white;
242 }
```

```
④ bestsellingsneaker.html | ④ home.html | # responsive.css • # style.css | :: || . . .
C: > Users > pelum > OneDrive > Documents > pullo (1) > css > # responsive.css > {} @media only screen and (min-
290 }
291
292 .about-box .titlepage h2::after {width: 45%;}
293 right: -281px;}}
294 .sport_text { margin-top: 0px;}
295 .Nursery-img .text-box h3 {
296 font-size: 36px;
297 padding: 126px 40px;
298 }
299 .main {
300 width: 20%;}
```

```
3 > bestsellingsneaker.html 1   <- home.html 1   # responsive.css •   # style.css ⚡   ||   ⏹   ⏷
4 > Users > pelum > OneDrive > Documents > pullo (1) > css > # responsive.css > {} @media only screen and (min-
5 d/
6 38
7 39   @media (max-width: 575px) {
8 40     .last { display: none; }
9 41     .carousel-caption {display: none;}
10 42     .header { padding-top: 0px; }
11 43     .logo { float: inherit; padding-bottom: 15px; text-align: center; }
12 44     .btn-primary {max-width: 126px;
13 45     margin-right: 2px;
14 46     font-size: 14px; }
15 47     .banner-main .carousel-caption h1 {font-size: 30px; line-height: 38px; }
16 48     .titlepage h2 {
17 49       font-size: 27px;
18 50     }
19 51     .sporrt_text{
20 52       font-size: 23px;
21 53       margin-top: 0px;
22 54     }
23 55     .main {
24 56       width: 30%;
25 57     }
26 58     .btn_main {
27 59       margin-top: 30px;
28 60       padding-bottom: 30px;
29 61       width: 55%; }
30 62
31 63 #main_slider a.carousel-control-prev {
32 64   position: absolute;
33 65   left: 175px;
34 66   top: 90%;
35 67   display: none;
36 68 }
37 69
38 70 #main_slider a.carousel-control-next {
39 71   position: absolute;
40 72   left: 236px;
41 73   top: 90%;
42 74   display: none;
43 75 }
44 76
45 77 .titlepage h2::after {
46 78   width: 93%; }
47 79
48 80   .about-box .titlepage h2::after {
49 81     width: 238px;
50 82     right: -50px;
51 83   }
52 84   .Gallery .titlepage h2 {
53 85     font-size: 47px;
```

```
.about-box .titlepage h2::after {
    width: 238px;
    right: -50px;
}
.Gallery .titlepage h2 {
    font-size: 47px;
}
.Gallery .titlepage h2::after {
    width: 89%;
}
.padding-right {
    padding-right: 15px;
}
.padding-left {
    padding-left: 15px;
}
.Nursery-img .text-box h3 {
    font-size: 23px;
    padding: 30px 14px;
}
.contact .titlepage h2::after {
    left: -8px;
    width: 281px;
}
.footer .headinga span {
    font-size: 16px;
}
.menu-bottom {
    margin-bottom: 45px;
}
ul.link li {
    padding: 8px 12px;
    padding-bottom: 0px;
}
.mens_taital { display: none; }
.mens_taital_2 { display: none; }
.page_no { display: none; }
.bg-light { background-color: #db5660 !important; }
.header_section { background-size: cover; }
.header_section { background-color: cover; background-repeat: no-repeat; }
```

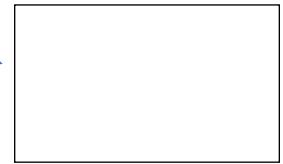
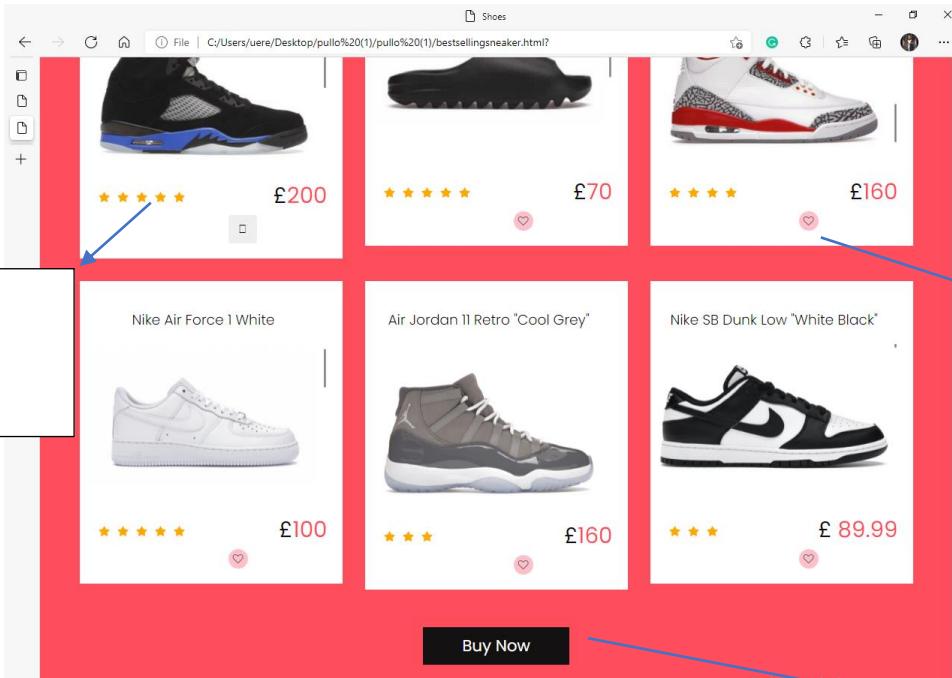
```
.buy_text { width: 100%; }
.email_box { width: 100%; }
.shop_banner { margin-top: 20px; }
.footer-logo { padding-top: 20px; }
.mail_section li { padding-top: 20px; }
.subscribe_bt { width: 55%; }
.out_shop_bt {
    width: 100%;
    font-size: 14pt;
    height: 50px;
}

.about-img { margin-left: 0px; }
.new_bt { margin-left: 0px; width: 33%; }
.about-img2 { margin-left: 0px; }

}
```

This concludes all the main parts on my main HTML home page.

Best-Selling Sneaker Page



```
128 <div class="collection_section layout_padding">
129   <div class="container">
130     <h1 class="new_text"><strong>Best Selling</strong></h1>
131     <p class="consectetur_text">These are some of the most popular releases that have sold the
132       most recently.
133   </div>
134 </div>
135 <div class="layout_padding gallery_section">
136   <div class="container">
137     <div class="row">
138       <div class="col-sm-4">
139         <div class="best_shoes">
140           <p class="best_text">Jordan 5 "Racer Blue" </p>
141           <div class="shoes_icon"></div>
142           <div class="star_text">
143             <div class="left_part">
144               <ul>
145                 <li><a href="#"></a></li>
146                 <li><a href="#"></a></li>
147                 <li><a href="#"></a></li>
148                 <li><a href="#"></a></li>
149                 <li><a href="#"></a></li>
150               </ul>
151             </div>
152             <div class="right_part">
153               <div class="shoes_price">£<span style="color: #ff4e5b;">200</span></div>
154               <button id="bttn1" class="btn" type="button"><i class="fas fa-heart"></i></button>
155             </div>
156           </div>
157         </div>
158       <div class="col-sm-4">
159         <div class="best_shoes">
160           <p class="best_text">Adidas Yeezy Slide "Onyx" </p>
161
162           <div class="shoes_icon"></div>
163           <div class="star_text">
164             <div class="left_part">
165               <ul>
166                 <li><a href="#"></a></li>
167                 <li><a href="#"></a></li>
168                 <li><a href="#"></a></li>
169                 <li><a href="#"></a></li>
170                 <li><a href="#"></a></li>
171               </ul>
172             </div>
173             <div class="right_part">
174               <div class="shoes_price">£<span style="color: #ff4e5b;">70</span></div>
175               <i class="fa fa-heart-o add-to-cart"></i>
176             </div>
177           ..
```

```

155      :</div>
156      :</div>
157    </div>
158    <div class="col-sm-4">
159      <div class="best_shoes">
160        <p class="best_text">Adidas Yeezy Slide "Onyx" </p>
161        <div class="shoes_icon"></div>
162        <div class="star_text">
163          <div class="left_part">
164            <ul>
165              <li><a href="#"></a></li>
166              <li><a href="#"></a></li>
167              <li><a href="#"></a></li>
168              <li><a href="#"></a></li>
169              <li><a href="#"></a></li>
170            </ul>
171          </div>
172          <div class="right_part">
173            <div class="shoes_price">&lt;span style="color: #ff4e5b;">70</span></div>
174            <i class="fa fa-heart-o add-to-cart"></i>
175          </div>
176        </div>
177      </div>
178    </div>
179  </div>
180  <div class="col-sm-4">
181    <div class="best_shoes">
182      <p class="best_text">Air Jordan 3 Retro "Cardinal Red" </p>
183      <div class="shoes_icon"></div>
184      <div class="star_text">
185        <div class="left_part">
186          <ul>
187            <li><a href="#"></a></li>
188            <li><a href="#"></a></li>
189            <li><a href="#"></a></li>
190            <li><a href="#"></a></li>
191          </ul>
192        </div>
193        <div class="right_part">
194          <div class="shoes_price">&lt;span style="color: #ff4e5b;">160</span></div>
195          <i class="fa fa-heart-o add-to-cart"></i>
196        </div>
197      </div>
198    </div>
199  </div>
200  </div>
201  </div>
202  </div>
203  <div class="row">
204    <div class="col-sm-4">

```

Div. class for rating system to be styled in CSS, under shows how stars are implemented. 5 lists with star image link will show 5 stars and so on

Code for heart button feature

```

bestsellingsneaker.html ✘ X js.js HTMLPage1.html login.html home.html
263  <div class="row">
264    <div class="col-sm-4">
265      <div class="best_shoes">
266        <p class="best_text">Nike Air Force 1 White</p>
267        <div class="shoes_icon"></div>
268        <div class="star_text">
269          <div class="left_part">
270            <ul>
271              <li><a href="#"></a></li>
272              <li><a href="#"></a></li>
273              <li><a href="#"></a></li>
274              <li><a href="#"></a></li>
275              <li><a href="#"></a></li>
276            </ul>
277          </div>
278          <div class="right_part">
279            <div class="shoes_price">&lt;span style="color: #ff4e5b;">100</span></div>
280            <i class="fa fa-heart-o add-to-cart"></i>
281          </div>
282        </div>
283      </div>
284    </div>
285    <div class="col-sm-4">
286      <div class="best_shoes">
287        <p class="best_text">Air Jordan 11 Retro "Cool Grey" </p>
288        <div class="shoes_icon"></div>
289        <div class="star_text">
290          <div class="left_part">
291            <ul>
292              <li><a href="#"></a></li>
293              <li><a href="#"></a></li>
294              <li><a href="#"></a></li>
295              <li><a href="#"></a></li>
296            </ul>
297          </div>
298          <div class="right_part">
299            <div class="shoes_price">&lt;span style="color: #ff4e5b;">160</span></div>
300            <i class="fa fa-heart-o add-to-cart"></i>
301          </div>
302        </div>
303      </div>
304    </div>
305    <div class="col-sm-4">
306      <div class="best_shoes">
307        <p class="best_text">Nike SB Dunk Low "White Black" </p>
308        <div class="shoes_icon"></div>
309        <div class="star_text">
310          <div class="left_part">

```

Price of item stored in separate div class so it can be styled as desired, in this case in the font red

All of my other product pages are all formatted in the same way;

Top nav bar always present with active class on active page

Product table with pictures, rating, price and favourite function

Buy now at bottom of product page with link to website URL where you can find all shoes

Footer at the bottom.

Having said this, there is no need to annotate the code for all the product pages as they're all in the same format with just different titles and pictures. Below is the code or the rest of my pages and if there is anything that needs to be annotated, it will be

Latest Sneaker Releases Page

The Latest Sneaker Releases

These are some of the hottest releases that have dropped recently!

| | | |
|--|--|---|
| <p>Jordan 6 Retro UNC "University Blue"</p>  <p>★★★★★</p> <p>£200</p> | <p>Nike Dunk Low "Court Purple"</p>  <p>★★★★★</p> <p>£89.99</p> | <p>Nike Air Jordan 1 Mid "Diamond Shorts"</p>  <p>★★★★★</p> <p>£150</p> |
| <p>Adidas Yeezy 700 V3 "Mono Safflower"</p>  <p>★★★</p> <p>£200</p> | <p>Nike Dunk Low "Chlorophyll"</p>  <p>★★★★★</p> <p>£89.99</p> | <p>Adidas Yeezy 700 "Waverunner"</p>  <p>★★★★★</p> <p>£ 250</p> |

[Buy Now](#)

  07533715751  dogunrin@outlook.com    

```
.html ✘ X bestsellingsneaker.html      js.js      HTMLPage1.html      login.html      home.html*
<div class="collection_section layout_padding">
  <div class="container">
    <h1 class="new_text"><strong>The Latest Sneaker Releases </strong></h1>
    <p class="consectetur_text">These are some of the hottest releases that have dropped recently</p>
  </div>
<!-- new arrivals end -->
<!-- Upcoming Releases section start -->
<div class="layout_padding gallery_section">
  <div class="container">
    <div class="row">
      <div class="col-sm-4">
        <div class="best_shoes">
          <p class="best_text">Jordan 6 Retro UNC "University Blue" </p>
          <div class="shoes_icon"></div>
          <div class="star_text">
            <div class="left_part">
              <ul>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
              </ul>
            </div>
            <div class="right_part">
              <div class="shoes_price">£<span style="color: #ff4e5b;">200</span></div>
            </div>
          </div>
        </div>
      </div>
      <div class="col-sm-4">
        <div class="best_shoes">
          <p class="best_text">Nike Dunk Low "Court Purple" </p>
          <div class="shoes_icon"></div>
          <div class="star_text">
            <div class="left_part">
              <ul>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
                <li><a href="#"></a></li>
              </ul>
            </div>
            <div class="right_part">
              <div class="shoes_price">£<span style="color: #ff4e5b;">89.99</span></div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```
release.html ✘ X bestsellingsneaker.html      js.js      HTMLPage1.html      login.html      home.html*
<div class="col-sm-4">
  <div class="best_shoes">
    <p class="best_text">Nike Air Jordan 1 Mid "Diamond Shorts" </p>
    <div class="shoes_icon"></div>
    <div class="star_text">
      <div class="left_part">
        <ul>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
        </ul>
      </div>
      <div class="right_part">
        <div class="shoes_price">£<span style="color: #ff4e5b;">150</span></div>
      </div>
    </div>
  </div>
</div>
<div class="row">
  <div class="col-sm-4">
    <div class="best_shoes">
      <p class="best_text">Adidas Yeezy 700 V3 "Mono Safflower" </p>
      <div class="shoes_icon"></div>
      <div class="star_text">
        <div class="left_part">
          <ul>
            <li><a href="#"></a></li>
            <li><a href="#"></a></li>
            <li><a href="#"></a></li>
          </ul>
        </div>
        <div class="right_part">
          <div class="shoes_price">£<span style="color: #ff4e5b;">200</span></div>
        </div>
      </div>
    </div>
  </div>
  <div class="col-sm-4">
    <div class="best_shoes">
      <p class="best_text">Nike Dunk Low "Chlorophyll" </p>
      <div class="shoes_icon"></div>
      <div class="star_text">
        <div class="left_part">
          <ul>
            <li><a href="#"></a></li>
          </ul>
        </div>
      </div>
    </div>
  </div>

```

```

lease.html ✘ X bestsellingsneaker.html      js.js      HTMLPage1.html      login.html*      home.html*

```

```

</div>
<div class="section_footer">
  <div class="container">
    <div class="mail_section">
      <div class="row">
        <div class="col-sm-6 col-lg-2">
          <div><a href="#"></a></div>
        </div>
        <div class="col-sm-6 col-lg-2">
          <div class="footer-logo"><span class="map_text">Call us</span></div>
        </div>
        <div class="col-sm-6 col-lg-3">
          <div class="footer-logo"><span class="map_text">Email us</span></div>
        </div>
        <div class="col-sm-6 col-lg-3">
          <div class="social_icon">
            <ul>
              <li><a href="#"></a></li>
              <li><a href="#"></a></li>
              <li><a href="#"></a></li>
              <li><a href="#"></a></li>
            </ul>
          </div>
        </div>
        <div class="col-sm-2"></div>
      </div>
    </div>
    <div class="footer_section_2">
      <div class="row">
        <div class="col-sm-4 col-lg-2">
          <p class="dummy_text"> </p>
        </div>
        <div class="col-sm-4 col-lg-2">
          <h2 class="shop_text">Address </h2>
          <div class="image-icon"><span class="pet_text">View map</span></div>
        </div>
        <div class="col-sm-4 col-md-6 col-lg-3">
          <h2 class="shop_text">Our Company </h2>
          <div class="delivery_text">
            <ul>
              <li>Delivery</li>
              <li>Legal Notice</li>
              <li>About us</li>
              <li>Secure payment</li>
              <li>Contact us</li>
            </ul>
          </div>
        </div>
      </div>
      <div class="col-sm-6 col-lg-3">
        <h2 class="address_text">Products</h2>
      </div>
    </div>
  </div>

```

The footer will be implemented on every page and is a copy of the footer that was described and annotated on the home page, so there is no need for me to display all code for every HTML page as a lot of my code is repeated. This keeps my app consistent looking and is also a use of a computational thinking technique that was described in my analysis section of my project: pattern recognition as I am making connections between similar problems and using this to make it easier for me to code and quicker for me to code.

Code:

```
○ bestsellingsneaker.html ● mains # favourite.css # responsive.css ○ latestsneakerrelease.html ● ○ lateststreetwearrelease.html ● ○

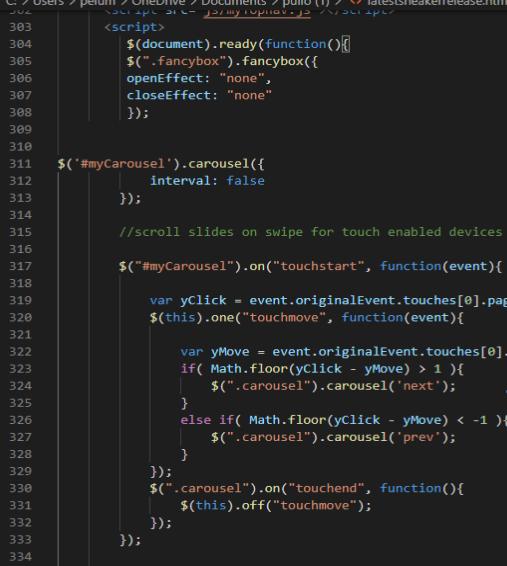
C:\Users\pelum\OneDrive\Documents\pull!() \ lateststreetwearrelease.html > HTML > body.main-layout > body.main-layout > div.layout_padding.collection_section

46 <a href="lateststreetwearrelease.html" class="active">Streetwear</a>
47 <a href="home.html">Home</a>
48 <a href="login.html">Login</a>
49
50 <div class="dropdown">
51   <button class="dropdownbtn">Footwear<i class="fa fa-caret-down"></i>
52   </button>
53   <div class="dropdown-content">
54     <a href="bestsellingsneaker.html">Best Selling Sneakers</a>
55     <a href="latesstsneakerrelease.html">Latest Sneaker Releases</a>
56     <a href="upcomingsneakerrelease.html">Upcoming Sneaker Releases</a>
57   </div>
58   </div>
59   <a href="reminders.html">Reminders</a>
60   <a href="about.html">About</a>
61
62   <a href="javascript:void(0);" class="icon" onclick="myFunction()">#9776;</a>
63 </div>
64   |   <div>
65   |   |   <div>
66   |   |   </div>
67   |   </div>
68
69 <div class="layout_padding collection_section">
70   <div class="container">
71     <h1 class="new_text"><strong>Latest Streetwear Releases</strong></h1>
72     <p class="consectetur_text">these are the latest and hottest streetwear items that have been dropped recently!</p>
73     <div class="collection_section_2">
74       <div class="row">
75         <div class="col-md-6">
76           <div class="about-img">
77             <button class="new_bt">New</button>
78             <div class="shoes-img"></div>
79             <p class="sport_text">Trapstar Drop March 11th <img alt="calendar icon" style="color: #f12a47;"></p>
80             <div class="dolar_text"><strong>See More</strong></div>
81           </div>
82           <form>
83             <button class="seemore_bt" formaction="https://heatsneakers.co.uk/collections/trapstar">See More</button>
84           </form>
85         </div>
86       </div>
87       <div class="col-md-6">
88         <div class="about-img">
89           <div class="shoes-img"></div>
90           <p class="sport_text">Cortefi Bolo Jacket Drop:</p>
91           <div class="dolar_text"><strong>See More</strong></div>
92         </div>
93       </div>
94     </div>
95   </div>
96 </div>
```

Makes streetwear in navbar highlighted to show actively in use

Displays new tag and
is designed via CSS to
be on item card

When clicking on the see more button, the user is taken to the page where he can buy these items for the cheapest prices available right now



```
C:\> Users > pelum > OneDrive > Documents > pullo (1) > > latestsneakerrelease.html > ...
303     <script>
304         $(document).ready(function(){
305             $(".fancybox").fancybox({
306                 openEffect: "none",
307                 closeEffect: "none"
308             });
309
310
311             $('#myCarousel').carousel({
312                 interval: false
313             });
314
315             //scroll slides on swipe for touch enabled devices
316
317             $("#myCarousel").on("touchstart", function(event){
318
319                 var yClick = event.originalEvent.touches[0].pageY;
320                 $(this).one("touchmove", function(event){
321
322                     var yMove = event.originalEvent.touches[0].pageY;
323                     if( Math.floor(yClick - yMove) > 1 ){
324                         $(".carousel").carousel('next');
325                     }
326                     else if( Math.floor(yClick - yMove) < -1 ){
327                         $(".carousel").carousel('prev');
328                     }
329                 });
330                 $(".carousel").on("touchend", function(){
331                     $(this).off("touchmove");
332                 });
333             });
334
335
336         </script>
337     </body>
338
339 </html>
```

Additional JavaScript function for when a user is using a touch enabled device, and this code will allow them to scroll with touch

This is important as it makes the app more accessible and portable, as people can use their touch screen phones with the pp and it will work perfectly. This code implemented on all pages.

Display:

The screenshot shows a dark-themed website for 'THE SNEAKER SUPPLIER'. At the top, there's a navigation bar with links: Contact, Streetwear (which is highlighted in green), Home, Login, Footwear, Reminders, and About. Below the navigation is a section titled 'Latest Streetwear Releases' with a sub-instruction: 'These are the latest and hottest streetwear items that have been dropped recently!'. A 'NOW' button is visible above a grid of 18 streetwear items. To the right of the grid is a large image of two people wearing bright yellow and green jackets. Below the grid, there's another row of items and a caption: 'Trapstar Drop March 11th'. At the bottom, a red 'See More' button is present.

New tag/icon to display items are new. Subtle feature that I added to improve aesthetic of website

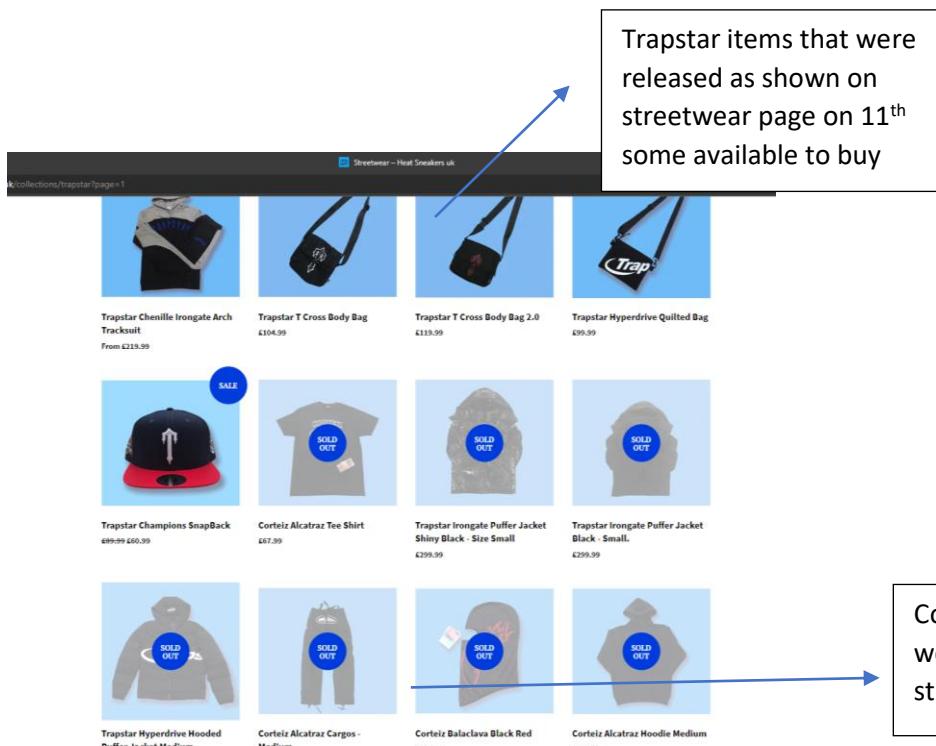
Streetwear section is highlighted on navbar as it is the feature currently in use, using an active class to do this. Also, dropdown feature from streetwear removed as only one category will now be used

Gives release information/date to inform the user so they can be ready for the release

Image block now different, not 6 products just the two from the two most popular streetwear brands right now.

This page is linked from the home page and can be accessed from the navigation bar.

Upon clicking the see more page, this website is loaded up:



TEST: **SUCCESS**

A lot of the clothes are already sold out, showing the competitiveness of people trying to cop the products, further highlighting the popularity of the items and furthering the need for my website so users can access these items quicker. See more button link will be updated when needed for example if there's another website with cheaper prices.

Richard Chinedu: This is very quick and efficient page that can be accessed very quickly from the nav bar and the pictures on the page are clear. The information provided especially for the trapstar drop helped me as I did not know when they were releasing, and I really wanted to get a jacket from there. The see more button was very useful as it took me directly to the page where I could buy my desired items. Overall, I am very happy with this page!

About Page

This is a simple page I decided to implement into my website and is there for if the user wishes to see information about the website and owner and when it was created and why it was created. It is an idea taken from the inspiration of many websites that have a 'About Us' section.

The code was pretty easy to implement:

```

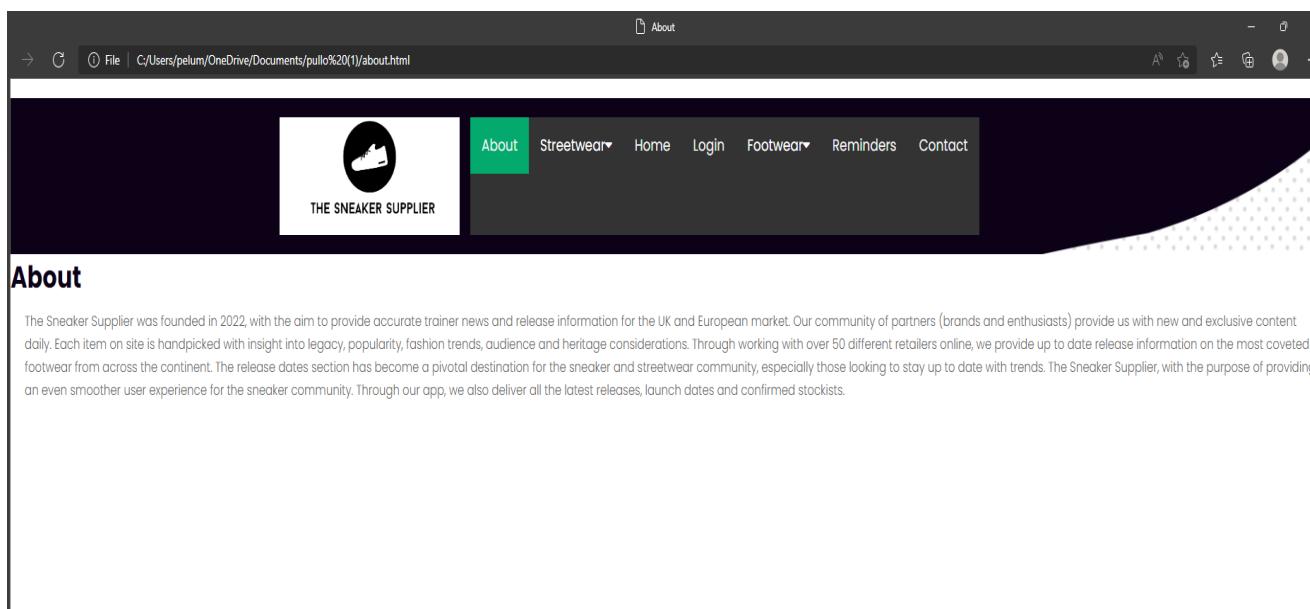
40 <div class="row">
41   <div class="col-sm-3">
42     <div class="logo"><a href="#"></a></div>
43   </div>
44   <div class="topnav" id="myTopnav">
45     <a href="about.html" class="active">About</a> →
46     <div class="dropdown">
47       <button class="dropbtn">Streetwear<i class="fa fa-caret-down"></i>
48     </button>
49     <a href="lateststreetwearrelease.html">Streetwear</a>
50     <a href="home.html">Home</a>
51     <a href="login.html">Login</a>
52   </div>
53   <div class="dropdown">
54     <button class="dropbtn">Footwear<i class="fa fa-caret-down"></i>
55   </button>
56   <div class="dropdown-content">
57     <a href="bestsellingsneaker.html">Best Selling Sneakers</a>
58     <a href="latestsneakerrelease.html">Latest Sneaker Releases</a>
59     <a href="upcomingsneakerrelease.html">Upcoming Sneaker Releases</a>
60   </div>
61   </div>
62   <a href="reminders.html">Reminders</a>
63   <a href="contact.html">Contact</a>
64
65 <a href="javascript:void(0);" class="icon" onclick="myFunction()">#9776;</a>
66 </div>
67   </div>
68 </div>
69 </div>
70 </div>
71   <h1 class="new_text"><strong>About</strong></h1>
72   <p>The Sneaker Supplier was founded in 2022, with the aim to provide accurate trainer news and release information for the UK and European market. Our community of partners (brands and enthusiasts) provide us with new and exclusive content daily. Each item on site is handpicked with insight into legacy, popularity, fashion trends, audience and heritage considerations. Through working with over 50 different retailers online, we provide up to date release information on the most coveted footwear from across the continent. The release dates section has become a pivotal destination for the sneaker and streetwear community, especially those looking to stay up to date with trends. The Sneaker Supplier, with the purpose of providing an even smoother user experience for the sneaker community. Through our app, we also deliver all the latest releases, launch dates and confirmed stockists.
77

```

Highlights about feature on navbar to show we are actively on this page

About us description in paragraph tags with about as heading in bold using strong tags

Display:



Favouriting Items and adding them to the reminders list

This is the final vital piece of code that I need to implement and one of the most important as it is key to the purpose of my actual website and why I decided to make it in the first place

Firstly, I implemented heart icons/buttons onto the item cards of all my products, the code for this can be seen below. Bear in mind that this same line of code would be implemented on every page, I have decided to only display one screenshot on the best-selling sneaker page:

Shoes

C:\Users\ure/Desktop/pullo%20(1)/pullo%20(1)/bestsellingsneaker.html

| | | |
|--|--|--|
| Jordan 5 "Racer Blue"  ★★★★★ £200  | Adidas Yeezy Slide "Onyx"  ★★★★★ £70  | Air Jordan 3 Retro "Cardinal Red"  ★★★★★ £160  |
| Nike Air Force 1 White  ★★★★★ £100  | Air Jordan 11 Retro "Cool Grey"  ★★★★ £160  | Nike SB Dunk Low "White Black"  ★★★ £ 89.99  |

Activate Windows
Go to Settings to activate Windows.

Buy Now

```
<div class="collection_section layout_padding">
    <div class="container">
        <h1 class="new_text"><strong>Best Selling</strong></h1>
        <p class="consectetur_text">These are some of the most popular releases that have sold the most recently!</p>
    </div>
</div>
<div class="layout_padding gallery_section">
    <div class="container">
        <div class="row">
            <div class="col-sm-4">
                <div class="best_shoes">
                    <p class="best_text">Jordan 5 "Racer Blue" </p>
                    <div class="shoes_icon"></div>
                    <div class="star_text">
                        <div class="left_part">
                            <ul>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                            </ul>
                        </div>
                        <div class="right_part">
                            <div class="shoes_price">£<span style="color: #ff4e5b;">200</span></div>
                            <i class="fa fa-heart-o add-to-cart"></i>
                        </div>
                    </div>
                </div>
            </div>
            <div class="col-sm-4">
                <div class="best_shoes">
                    <p class="best_text">Adidas Yeezy Slide "Onyx" </p>
                    <div class="shoes_icon"></div>
                    <div class="star_text">
                        <div class="left_part">
                            <ul>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                                <li><a href="#"></a></li>
                            </ul>
                        </div>
                        <div class="right_part">
                            <div class="shoes_price">£<span style="color: #ff4e5b;">70</span></div>
                            <i class="fa fa-heart-o add-to-cart"></i>
                        </div>
                    </div>
                </div>
            </div>
            <div class="col-sm-4">
                <div class="best_shoes">
                    <p class="best_text">Air Jordan 3 Retro "Cardinal Red" </p>
                </div>
            </div>
        </div>
    </div>

```

```

ontact1.js      registerr.html      style.css*      experiemnt.html      latestsneakerrelese.html      bestsellingsneaker.html -+


<div class="left_part">
        <ul>
            <li><a href="#"></a></li>
            <li><a href="#"></a></li>
            <li><a href="#"></a></li>
            <li><a href="#"></a></li>
            <li><a href="#"></a></li>
        </ul>
    </div>
    <div class="right_part">
        <div class="shoes_price">&lt;span style="color: #ff4e5b;">100</span></div>
        <i class="fa fa-heart-o add-to-cart"></i>
    </div>
</div>
</div>
<div class="col-sm-4">
    <div class="best_shoes">
        <p class="best_text">Air Jordan 11 Retro "Cool Grey"</p>
        <div class="shoes_icon"></div>
        <div class="star_text">
            <div class="left_part">
                <ul>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                </ul>
            </div>
            <div class="right_part">
                <div class="shoes_price">&lt;span style="color: #ff4e5b;">160</span></div>
                <i class="fa fa-heart-o add-to-cart"></i>
            </div>
        </div>
    </div>
</div>
<div class="col-sm-4">
    <div class="best_shoes">
        <p class="best_text">Nike SB Dunk Low "White Black"</p>
        <div class="shoes_icon"></div>
        <div class="star_text">
            <div class="left_part">
                <ul>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                </ul>
            </div>
            <div class="right_part">
                <div class="shoes_price">&lt;span style="color: #ff4e5b;">89.99</span></div>
                <i class="fa fa-heart-o add-to-cart"></i>
            </div>
        </div>
    </div>
</div>


```

As shown the heart icons were implemented successfully with the correct colours thanks to my CSS designs. Now my next step would be to create functions in JavaScript that would allow me to click on a heart button and toggle it so it changes to a different colour, and after this was done it would add this to the user's reminders list successfully, the code below shows me trying to do this:

```
var noti = document.querySelector('h2');
var select = document.querySelector('.select');
var button = document.getElementsByTagName('button1');

for(var but of button1){
    but.addEventListener('click', (e)>{
        var add = Number(noti.getAttribute('data-count') || 0);
        noti.setAttribute('data-count', add +1);
        noti.classList.add('zero');

        // image --animation to cart ---
        var image = e.target.parentNode.querySelector('.shoes_icon');
        var span = e.target.parentNode.querySelector('span');
        var s_image = image.cloneNode(false);
        span.appendChild(s_image);
        span.classList.add("active");
        setTimeout(()>{
            span.classList.remove("active");
            span.removeChild(s_image);
        }, 500);

        // copy and paste //
        var parent = e.target.parentNode;
        var clone = parent.cloneNode(true);
        select.appendChild(clone);
        clone.lastElementChild.innerText = "Buy-now";

        if (clone) {
            noti.onclick = ()>{
                select.classList.toggle('display');
            }
        }
    })
}
}
```

JavaScript files with functions that will be called inside the HTML document

```
@import url('https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css');

*{
    margin: 0;
    padding: 0;
    color: #454545;
}

h2{
    width: 3%;
    position: relative;
    top: 60px;
    left: 90%;
    cursor: pointer;
}

h2:before{
    content: attr(data-count);
    color: white;
    position: absolute;
    right: 16px;
    font-size: 15px;
    text-align: center;
    top: -12px;
    width: 20px;
    height: 20px;
    background: red;
    border-radius: 50%;
    opacity: 0;
}

h2.zero:before{
    opacity: 1;
}

section best_shoes:nth-child(1)>span.active{
    animation: first 0.5s ease-in;
    z-index: 2;
    display: block;
}

section best_shoes:nth-child(2)>span.active{
    animation: second 0.5s ease-in;
    z-index: 2;
    display: block;
}

section best_shoes:nth-child(3)>span.active{
    animation: third 0.5s ease-in;
    z-index: 2;
    display: block;
}

section best_shoes:nth-child(4)>span.active{
    animation: four 0.5s ease-in;
    z-index: 2;
    display: block;
}

No issues found
```

```
        }
    section best_shoes:nth-child(4)>span.active{
        animation: four 0.5s ease-in;
        z-index: 2;
        display: block;
    }
    @keyframes first{
        to{
            width: 30px;
            height: 20px;
            left: 1290px;
            top: -70px;
        }
    }
    @keyframes second{
        to{
            width: 30px;
            height: 20px;
            left: 948px;
            top: -70px;
        }
    }
    @keyframes third{
        to{
            width: 30px;
            height: 20px;
            left: 606px;
            top: -70px;
        }
    }
    @keyframes four{
        to{
            width: 30px;
            height: 20px;
            left: 265px;
            top: -70px;
        }
    }
}

.select{
    width: 60%;
    height: 580px;
    padding: 5%;
    background: #222;
    position: absolute;
    top: -1000px;
    left: 20%;
    transition: 0.5s;
    overflow-y: auto;
    margin: auto;
}

No issues found
```

```
.select{
    width: 60%;
    height: 580px;
    padding: 5%;
    background: #222;
    position: absolute;
    top: -1000px;
    left: 20%;
    transition: 0.5s;
    overflow-y: auto;
    margin: auto;
}
.select.display{
    top: 10px;
}
.select div{
    width: 100%;
    height: 200px;
    display: flex;
    padding: 5px;
    border: 1px solid white;
    position: relative;
    margin: 5px auto;
}
.select shoes_icon{
    width: 300px;
    height: 100%;
}
.select div p{
    padding: 35px;
    color: white;
}
.select shoe_price,
.select div button1 {
    position: absolute;
    left: 45%;
    top: 50%;
    color: white;
}
.select div button1{
    left: 60%;
    top: 55%;
}
.select div span{
    display: none;
}
```

```

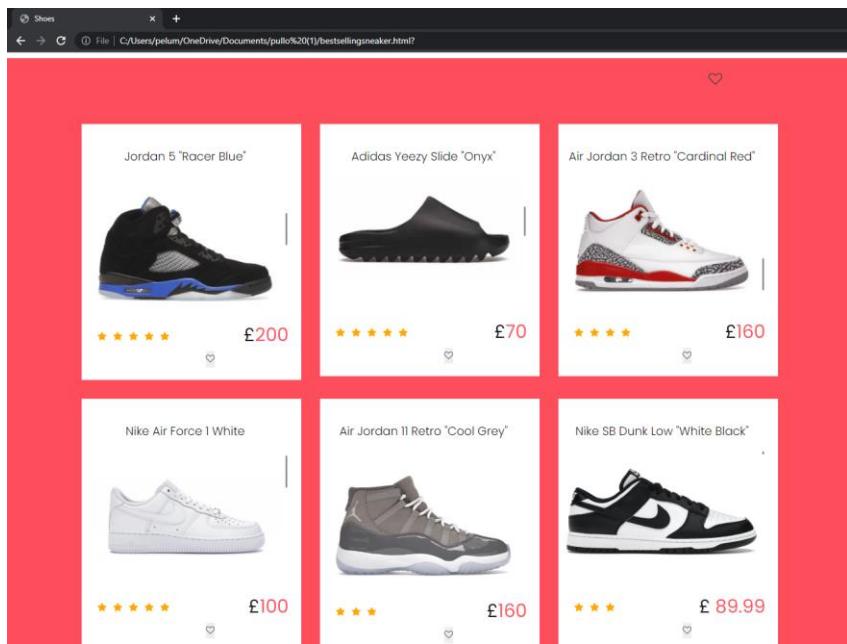
<!--
-->

<section>
  <div class="container">
    <h1 class="new_text"><strong>Best Selling</strong></h1>
    <p class="consectetur_text">These are some of the most popular releases that have sold the most recently!</p>
    <h2><i class=" fa fa-heart-o add-to-cart"></i></h2>
  </div>

  <div class="layout_padding gallery_section">
    <div class="container">
      <div class="row">
        <div class="col-sm-4">
          <div class="best_shoes">
            <p class="best_text">Jordan 5 "Racer Blue" </p>
            <div class="shoes_icon"></div>
            <div class="star_text">
              <div class="left_part">
                <ul>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                  <li><a href="#"></a></li>
                </ul>
              </div>
              <div class="right_part">
                <div class="shoes_price">£<span style="color: #ff4e5b;">200</span></div>
                <span></span>
                <button class="button button1"><i class=" fa fa-heart-o add-to-cart"></i></button>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div class="col-sm-4">
    <div class="copyright">2022 All Rights Reserved. <a href="https://html.design">The Sneaker Supplier</a>
    <div class="select">
      <!-- Javascript files -->
      <script src="js/jquery.min.js"></script>
      <script src="js/popper.min.js"></script>
      <script src="js/bootstrap.bundle.min.js"></script>
      <script src="js/jquery-3.0.0.min.js"></script>
      <script src="js/plugin.js"></script>
      <!-- sidebar -->
      <script src="js/jquery.mCustomScrollbar.concat.min.js"></script>
      <script src="js/custom.js"></script>
      <!-- javascript -->
      <script src="js/owl.carousel.js"></script>
      <script src="https://cdnjs.cloudflare.com/ajax/libs/fancybox/2.1.5/jquery.fancybox.min.js"></script>
      <script src="js/myTopnav.js"></script>
      <script type="text/javascript" src="js/main.js"></script>
      <script>
        window.addEventListener('load', function() {
          var loader = document.getElementById('loader');
          loader.style.display = 'none';
        });
      </script>
    </div>
  </div>

```

Unfortunately, after I ran my code after coding up the JavaScript functions and attempting to call them, the code failed to implement properly. Nothing happened when clicking the heart buttons, which were meant to add to the favourites list, and when clicking the favourites icon to show the favourites list. Also, I tried to create an animation as you can see in my CSS that would make the item drag into the favourites icon to improve the aesthetic, but it also failed to implement.



I tried to click on the favourite's icon and the heart buttons on the shoes but nothing was happening, indicating that there was an issue with my code.

TEST: **FAIL**

Second Prototype/Version after End-User Review

Now that I have finished the programming, I have asked my main stakeholder to test the current version of my website and see if it meets all his/end-user requirements. He reviewed and said:

Richard: I have tried your website and I have to say I am very impressed with my experience on the site. The aesthetic is very pleasing and is a perfect balance of simple and exciting. The navigation through the website was very easy and efficient and when trying to go and cop a pair of shoes, it was very quick when trying to get to the website due to the item I wanted already being on the home page. All I had to do was to click on the link to the page via the see more page and then it took me to the page where I had to favourite the item, and after favouriting it I clicked on the favourite's icon at the top of that page and all the items I favourited were successfully loaded into the favourites list and then the links were then loaded when clicking the buy button. I clicked on one of the links, in this case to stockx as it was the best price right now as I saw from the Sneaker supplier website as it was at the top of all the other links. Also, when trying to contact the website, when inputting my data, they were entered and sent successfully which was a bit of a add on feature that I really appreciated. The last great thing was when creating my account and logging in it was flawless and very successful, notifying my when my user was created and when going to the log in page, I entered those same details, and I was logged in successfully and taken to the home page. It differently meets a lot of my initial requirements. However, upon my time on the website I did encounter some issues and also some things that I would recommend the website can improve upon:

- When contacting the website, it would be better to have an alert when the user has clicked send to have that assurance that the website owners have received the message
- Logging out successfully after logging in
- The dropdown navigation feature was not working and if this was fixed then it would make the site even more efficient than it already is as the only way I could get to these footwear sections was under the see more buttons on the carousel etc.
- After favouriting items function is not working

After listening to my end user's answer, I have decided to try and implement the features that he has requested. I have also decided to add some validation to my contact form to ensure data entered is sensible and feasible, and that data is at least entered before trying to send a message. This is so empty spaces don't fill up my database and so a set sequence of values with the same format are stored in my database for me to see:

After every improvement I make, I ask the end-user to review them to check for his satisfaction, as my end-user represents all user's satisfaction. This is important to try and get the best possible version I can get within my capacity.

1. 2nd Attempt/Version at the dropdown navigation bar

After analysing my code carefully, I found where I went wrong. It was in my navbar.css file for my design for the navigation bar. Both the overflow properties in the .topnav and in the .dropdown had been set to hidden and the display in the dropdown-content had been set to list-item. I changed the display to none, and I took away the overflows by making them a comment. I also added another overflow button under dropdown: hover .dropdown-content and set it to visible so the user could see the contents when hovered over. These are all the changes that I believed would solve the problem. My edited code can be viewed below:

COMMENT,
INDICATED BY /*
*/

Overflow set to visible so when mouse hovers over the dropdown button the contents are displayed.

```
/* Show the dropdown menu when the user moves the mouse over the dropdown button */  
.dropdown:hover .dropdown-content {  
    overflow: visible;  
    display: block;  
}
```

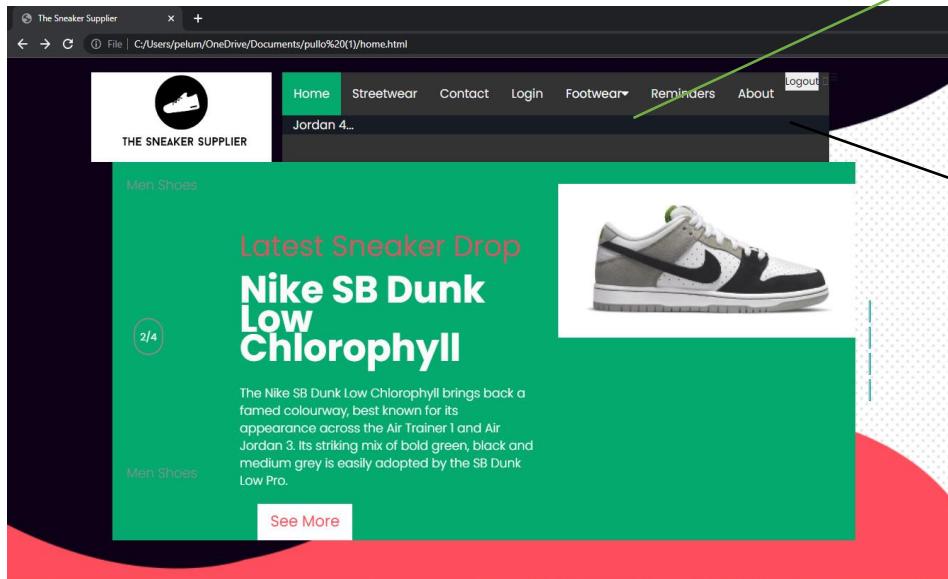
```
/* Style the dropdown content */  
.dropdown-content {  
    display: none;  
    position: absolute;  
    background-color: #f9f9f9;  
    min-width: 160px;  
    box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);  
    z-index: 1;  
}
```

Changed property from list-item to none. So you can't see drop down content unless you hover as stated in the comment above screenshot

```
.dropdown {  
    float: left;  
/* overflow: hidden; */  
}
```

The overflow was set to hidden before, then I removed it by adding comment tags which means it wouldn't be implemented into the CSS

```
.topnav {  
    background-color: #333;  
/* overflow: hidden; */  
}
```



Not hovering,
footwear
dropdown
not activated



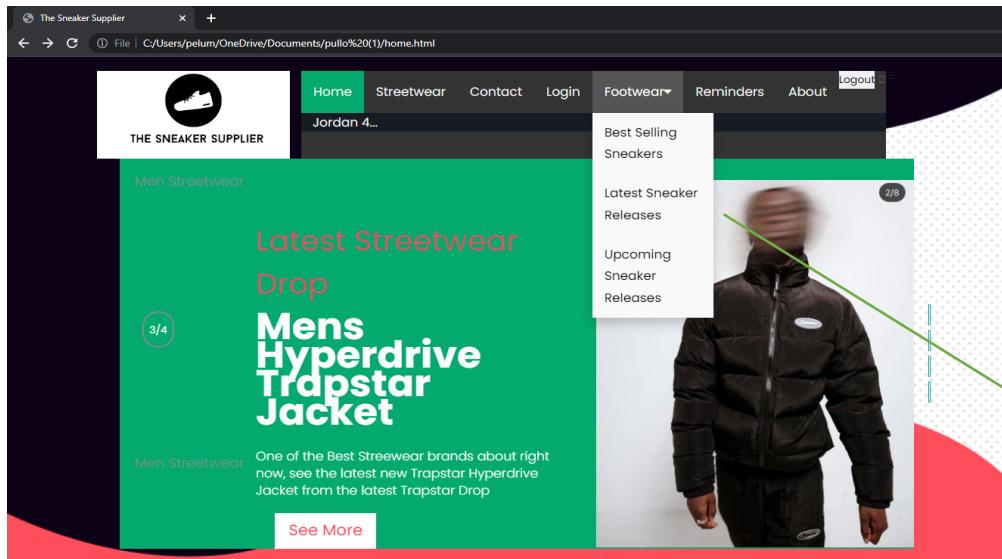
After inputting
the Jordan 4 shoe
name and
pressing enter,
nothing
happened. **TEST:
FAIL**

Yeezy March

The New Yeezy drops for the upcoming month of March

New

The first screenshot shows me not hovering over the footwear dropdown button. Note that I took away the dropdown feature for streetwear as there was only one page, I made use of which was the latest streetwear release



When footwear is
hovered over,
subcategories are shown
and when clicked will
take them to the desired
page

Yeezy March

The New Yeezy drops for the upcoming month of March

New

The second screenshot holds me hovering and the categories displaying.

From this, I have concluded that the dropdown works successfully, and after testing the feature, I can hail this test a **SUCCESS**.

Richard Chinedu: I am happy with the improvement shown of the website and with this fully functioning menu at the top it will be significantly easier to navigate to find the shoe that I want.

2. Website Contact and Validation

To improve my code and meet my criteria and requirements in previous stages, I decided to improve the quality of my contact form by adding validation. This was a requirement that I was to put in from my validation table in my design. Using patterns from my previous validation for example my log in page, coding up the validation for my contact form in conjunction with saving to the database was not that complicated to do

Below shows the new updated JavaScript form for my contact form saved as contact1.js, referenced and linked to at the bottom of my contact form html page, both on the home page and on the separate html page shown below:

```
.. <-- bestsellingsneaker.html 1   # favourite.css.css   # responsive.css   <-- latests :: II ⏪ ⏴ ⏵ ⏶ style.css 8   JS ms
C: > Users > pelum > OneDrive > Documents > pullo (1) > js > contact1.js > validate

16
17
18 // reference your database
19 var contactFormDB = firebase.database().ref("contactForm");
20
21 document.getElementById("contactForm").addEventListener("submit", submitForm);
22
23
24 function validate(){
25     var name = document.getElementById("name").value;
26     var email = document.getElementById("emailid").value;
27     var message = document.getElementById("msgContent").value;
28     var error_message = document.getElementById("error_message");
29
30     error_message.style.padding = "10px";
31
32     var text;
33     if(name.length < 5){
34         text = "Please Enter valid Name";
35         error_message.innerHTML = text;
36         return false;
37     }
38
39     if(email.indexOf("@") == -1 || email.length < 6){
40         text = "Please Enter valid Email";
41         error_message.innerHTML = text;
42         return false;
43     }
44     if(message.length <= 10){
45         text = "Please Enter More Than 10 Characters";
46         error_message.innerHTML = text;
47         return false;
48     }
49     alert("Form Submitted Successfully!");
50     return true;
51 }
52
53
54 function submitForm(e) {
55     e.preventDefault();
56
57     var name = getElementVal("name");
58     var emailid = getElementVal("emailid");
59     var msgContent = getElementVal("msgContent");
60
61     saveMessages(name, emailid, msgContent);
62
63 }
```

Validation function that I added to improve this section of code

As shown in the comment, this is the line referencing my database so the values can save there. The line below is a method that attaches an event handler to an element. The event is described in the parameters, and so is the function that will be performed when that event occurs. In this case in my HTML form the div section under the id contactForm will perform the submitForm function when the submit button is clicked

This section describes my own personal validation for each field. For example, the form will not be submitted if the name is less than 5 characters, and will return this error message

If all validation criteria is met, the form is submitted and an alert message should be displayed.

The Sneaker Supplier

File | C:/Users/pelum/OneDrive/Documents/pullo%20(1)/contact.html?

Home Streetwear Contact Login Footwear Reminders About

Jordan 4...

Contact Us

Contact Now

Please Enter valid Name

ee

ss

Please include an '@' in the email address. 'ss' is missing an '@':

Good work!

Send

After trying to submit with a name under 5 characters, this message is displayed. This shows a **SUCCESS** for the name validation

If the precise correct format for an email is not entered, this validation message is now displayed. Email validation test: **SUCCESS**

This screenshot shows a contact form titled 'Contact Now'. It has two input fields: 'Name' (containing 'ee') and 'Email Address' (containing 'ss'). A validation message above the email field says 'Please include an "@" in the email address. "ss" is missing an "@"'. Below the fields, a success message says 'Good work!'. A red box highlights the validation message for the email field, with an arrow pointing to it from the explanatory text on the right.

The Sneaker Supplier

File | C:/Users/pelum/OneDrive/Documents/pullo%20(1)/contact.html?

Contact Us

Contact Now

Please Enter valid Name

Full Name

Email Address

Please fill out this field.

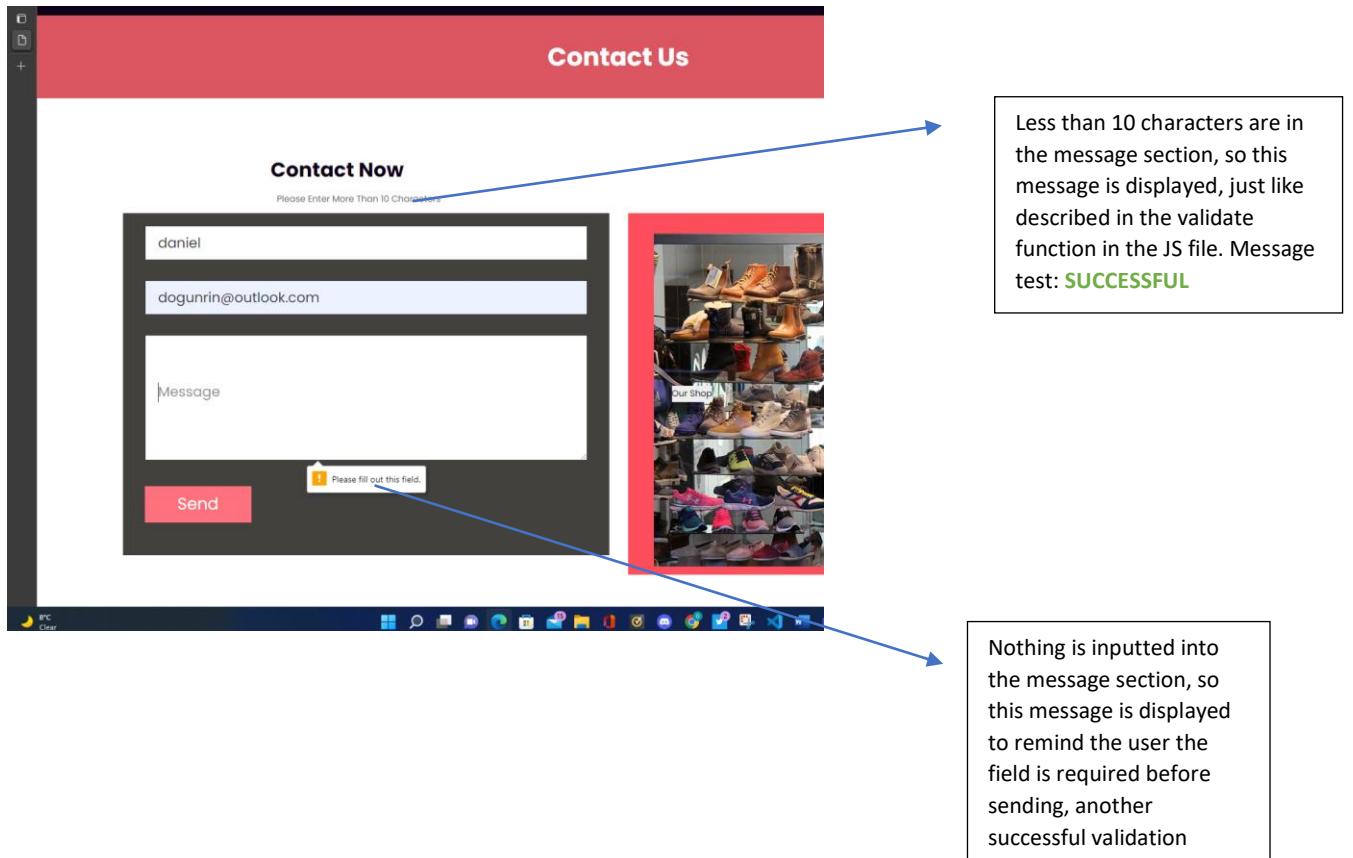
Message

Send

If nothing is inputted into any fields this is displayed, showing a **SUCCESSFUL** validation for when nothing is inputted to the required fields

This screenshot shows a contact form with three input fields: 'Name', 'Email Address', and 'Message'. The 'Email Address' field has a validation message 'Please fill out this field.' A red box highlights this message, with an arrow pointing to it from the explanatory text on the right. The 'Name' and 'Message' fields are empty.





PARTIAL SUCCESS is the category of success that I have deemed for this improved section, as I have successfully added the validation to the contact form however the alert message after the page was submitted failed to implement, but everything else worked well. Regardless, this is a significant improvement from my original version

Richard Chinedu: Yes, I agree this has improved the websites quality and I will be using this contact form section more now. However, the alert message is not vital but is still needed in my opinion

3. Favouriting Items

After successfully improving my code with my contact form validation and my functional topnav bar after I reviewed it with my end-user, I then tried to improve the functionality of the favouriting items feature, below is code showing me trying to implement this into the system. My plan was to after a user clicks the heart icon on an item, the heart icon in the top of the page would set its visible index to 1 to show the user now has one item in his favourites list. Then if the user wishes to access and view his list, he could do this by clicking that heart icon with the index and a dropdown page from above will be displayed with all the items and a button saying where to buy. When a user clicks the button, he was supposed to be shown links the best websites that are currently selling that item, below is the code for me attempting to do this:

```

C:\> Users > pelum > OneDrive > Documents > pullo (1) > bestsellingsneaker.html > body.main.layout > section > div.layout.padding.gallery_section >
164     <ul>
165         <li><a href="#"></a></li>
166         <li><a href="#"></a></li>
167         <li><a href="#"></a></li>
168     </ul>
169 </div>
170 <div class="right_part">
171     <div class="shoes_price"><span style="color: #ff4e5b;">160</span></div>
172     <span></span>
173     <button><i class="fa fa-heart-o add-to-cart"></i></button>
174 </div>
175 </div>
176 </div>
177 <div class="col-sm-4">
178     <div class="best_shoes">
179         <p class="best_text">Nike SB Dunk Low "White Black"</p>
180         <div class="shoes_icon"></div>
181         <div class="star_text">
182             <div class="left_part">
183                 <ul>
184                     <li><a href="#"></a></li>
185                     <li><a href="#"></a></li>
186                     <li><a href="#"></a></li>
187                 </ul>
188             </div>
189             <div class="right_part">
190                 <div class="shoes_price"><span style="color: #ff4e5b;">89.99</span></div>
191                 <span></span>
192                 <button><i class="fa fa-heart-o add-to-cart"></i></button>
193             </div>
194         </div>
195     </div>
196 </div>
197 </div>
198 </div>
199 </div>
200 </div>
201 </div>
202 </div>
203 </div>
204 <div class="select">
205     <div class="section_footer">
206         <div class="container">
207             <div class="wall_section">
208                 <div class="row">
209                     <div class="col-sm-6 col-lg-2">

```

Heart icon feature
to be clicked

Div. section referenced to
in CSS and JavaScript where the
favourites list will be displayed

```

File Edit Selection View Go Run Terminal Help
C:\> bestsellingsneaker.html 1 > main.js # favourite.css # responsive.css latestsneakerrelease.html # style.css contact.html
C:\> Users > pelum > OneDrive > Documents > pullo (1) > bestsellingsneaker.html > body.main.layout > section > div.layout.padding.gallery_section > div.container > div.row >
75
76
77
78 <section>
79     <div class="container">
80         <h1 class="new_text"><strong>Best Selling</strong></h1>
81         <p class="consecutetor_text">These are some of the most popular releases that have sold the most recently</p>
82         <i class="fa fa-heart-o add-to-cart"></i></h2>
83     </div>
84
85     <div class="layout_padding_gallery_section">
86         <div class="row">
87             <div class="col-sm-4">
88                 <div class="best_shoes">
89                     <p class="best_text">Jordan 5 "Racer Blue" </p>
90                     <div class="shoes_icon"></div>
91                     <div class="star_text">
92                         <div class="left_part">
93                             <ul>
94                                 <li><a href="#"></a></li>
95                                 <li><a href="#"></a></li>
96                                 <li><a href="#"></a></li>
97                             </ul>
98                         </div>
99                         <div class="right_part">
100                             <div class="shoes_price"><span style="color: #ff4e5b;">200</span></div>
101                             <span></span>
102                             <button><i class="fa fa-heart-o add-to-cart"></i></button>
103                         </div>
104                     </div>
105                 </div>
106                 <div class="col-sm-4">
107                     <div class="best_shoes">
108                         <p class="best_text">Adidas Yeezy Slide "Onyx" </p>
109                         <div class="shoes_icon"></div>
110                         <div class="right_part">
111                             <div class="shoes_price"><span style="color: #ff4e5b;">70</span></div>
112                             <span></span>
113                             <button><i class="fa fa-heart-o add-to-cart"></i></button>
114                         </div>
115                     </div>
116                 </div>
117                 <div class="col-sm-4">
118                     <div class="best_shoes">
119                         <p class="best_text">Air Jordan 3 Retro "Cardinal Red" </p>
120                         <div class="shoes_icon"></div>
121
122

```

Heading icon referenced to
in CSS and JavaScript where
index of amount of items in
list will be displayed. When
clicked on the favourites list
should drop down

```

File Edit Selection View Go Run Terminal Help
C:\> bestsellingsneaker.html 1 > main.js # favourite.css # responsive.css latestsneakerrelease.html # style.css
C:\> Users > pelum > OneDrive > Documents > pullo (1) > bestsellingsneaker.html > body.main.layout > section > div.layout.padding.gallery_section > div.container > div.row > main.js >
1 var notif = document.querySelector('#notif');
2 var select = document.querySelector('.select');
3 var button = document.getElementById('button');
4 for(var but = button; but != null; but = but.previousElementSibling)
5     but.addEventListener('click', ()=>{
6         var add = Number(notif.getAttribute('data-count')) || 0;
7         notif.setAttribute('data-count', add + 1);
8         notif.classList.add('zero');
9
10        // image --animation to cart -----
11        var image = e.target.parentNode.querySelector('img');
12        var span = e.target.parentNode.querySelector('span');
13        var s_image = image.cloneNode(false);
14        span.appendChild(s_image);
15        span.classList.add('active');
16        setTimeout(()=>{
17            span.classList.remove('active');
18            span.removeChild(s_image);
19        }, 500);
20
21        // copy and paste --
22        var parent = e.target.parentNode;
23        var clone = parent.cloneNode(true);
24        select.appendChild(clone);
25        clone.lastElementChild.innerText = "Where to Buy";
26
27        if (clone) {
28            notif.onclick = ()=>{
29                select.classList.toggle('display');
30            }
31        }
32    }
33
34
35

```

Variables
initiated, class id
from HTML page

Function for when heart
button is clicked, the
item is added to list and
index is incremented

Animation to
add image to
cart

When header favourites icon is
clicked, favourites list
displayed

```

File Edit Selection View Go Run Terminal Help
bestsellingsneaker.html  JS main.js # favourite.css # responsive.css latestbreakerreleas
C:\> Users > pelum > OneDrive > Documents > pullo (1) > css > # favourite.css > h2:before
1 @import url('https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.m
2 *{
3   margin: 0;
4   padding: 0;
5   color: #454545;
6 }
7 h2{
8   width: 3%;
9   position: relative;
10  top: 60px;
11  left: 90%;
12  cursor: pointer;
13 }
14 h2:before{
15   content: attr(data-count);
16   color: white;
17   position: absolute;
18   left:90%;
19   font-size: 15px;
20   text-align: center;
21   top: 60px;
22   width: 3%;
23   height: 50px;
24   background: black;
25   border-radius: 50%;
26   opacity: 0;
27 }
28 h2.zero:before{
29   opacity: 1;
30 }
31
32
33 img{
34   width: 300px;
35   height: 180px;
36 }
37
38 button{
39   padding: 5px;
40   background: red;
41   border: none;
42   outline: none;
43   font-weight: bold;
44   color: #fafafa;
45   cursor: pointer;
46 }
47
48
49 <section><div><h2><span>

```

Data count from javascript, index will increase when user favourites an item.
 Design for the heading heart icon/favourites icon and also designs the index, e.g colour of index number background will be black.

Favourites heart button is designed like this

```

File Edit Selection View Go Run Terminal Help
bestsellingsneaker.html  JS main.js # favourite.css # responsive.css latestbreakerreleas
C:\> Users > pelum > OneDrive > Documents > pullo (1) > css > # favourite.css > h2:before
40 section div:nth-child(1)>span.active{
41   animation: first 0.5s ease-in;
42   z-index: 2;
43   display: block;
44 }
45
46 section div:nth-child(2)>span.active{
47   animation: second 0.5s ease-in;
48   z-index: 2;
49   display: block;
50 }
51
52 section div:nth-child(3)>span.active{
53   animation: third 0.5s ease-in;
54   z-index: 2;
55   display: block;
56 }
57
58 section div:nth-child(4)>span.active{
59   animation: four 0.5s ease-in;
60   z-index: 2;
61   display: block;
62 }
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96

```

CSS design for animation when the product is added to the favourites list, would show the product visibly entering the favourites list icon

```
File Edit Selection View Go Run Terminal Help
bestsellingsneaker.html  main.js  # favourite.css  responsive.css  latestsneakerrelease.html  style.css
C:\Users\pelum\OneDrive\Documents>pullo (1)\css\favourite.css> h2:before
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
}
@keyframes four{
    to{
        width: 30px;
        height: 20px;
        left: 265px;
        top: -70px;
    }
}

.select{
    width: 60%;
    height: 580px;
    padding: 5px;
    background: #2222;
    position: absolute;
    top: -90px;
    left: 20%;
    transition: 0.5s;
    overflow-y: auto;
    margin: auto;
}
.select.display{
    top: 10px;
}
.select div{
    width: 100%;
    height: 200px;
    display: flex;
    padding: 5px;
    border: 1px solid white;
    position: relative;
    margin: 5px auto;
}
.select div img{
    width: 300px;
    height: 100%;
}
.select div p{
    padding: 35px;
    color: white;
}
.select div h6,
.select div button{
    position: absolute;
    left: 45%;
```

The select section in the HTML page should be designed like this at the with a black background, this is for when the user clicks the favourites icon, and the favourites list is displayed

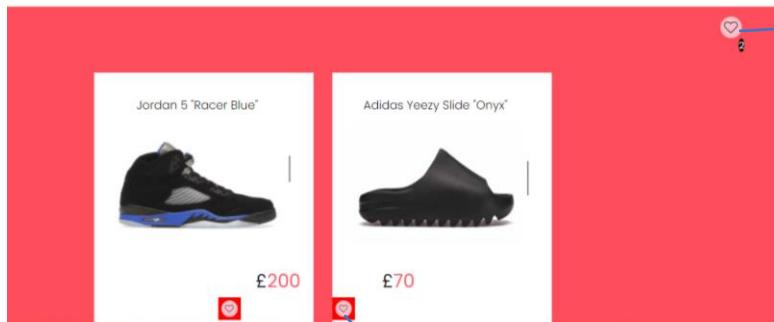
Favourites list should be displayed from the top

Image and other components should be displayed and put on this favourites list when displayed as well

```
143
144
145
146
147
148
149
150
151
152
153
154
}
.select div button{
    left: 60%;
    top: 55%;
}
.select div span{
    display: block;
```

Best Selling

These are some of the most popular releases that have sold the most recently!



The favourites list index is updated once the user clicks the favourite icon, this screenshot displays that and as you can see, I favourited these two items and the favourites list index displays 2

These are the favourite buttons on each item and were designed by CSS. When clicked, they should be added into a favourites list

I tried my hardest in the code to implement the favourites list and display it after the user clicks the header icon, but it was not displaying with the items. My original plans were to implement links to the bestselling pages where the user can buy these shoes that are displayed on this favourite display page but since this was not working I implemented the links in another way in my next version:

Review:

Richard Chinedu: When clicking the icons, I was satisfied with the fact that those items appeared to have been going to my favourites list as the number kept increasing the more, I clicked on the heart icon but when I clicked on it, nothing happened. This was underwhelming for me. However, the use of icons to make the website look more aesthetically pleasing was brilliant here. To improve, rather than trying more to get the favourites list to show, you could click on the item and be able to access its links and information from there.

Final Version/Prototype After End-User Reviews

Contact Alert

```

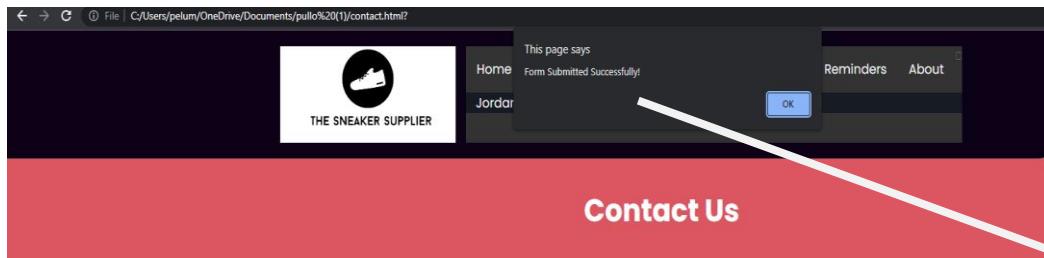
C:\Users\pelum>OneDrive>Documents>pullo(1)>j>JS contact.js>validate
38
39     if(email.indexOf("@") == -1 || email.length < 6){
40         text = "Please Enter valid Email";
41         error_message.innerHTML = text;
42         return false;
43     }
44     if(message.length <= 10){
45         text = "Please Enter More Than 10 Characters";
46         error_message.innerHTML = text;
47         return false;
48     }
49     alert("Form Submitted Successfully!");
50     return true;
51 }

52
53
54 function submitForm(e) {
55     e.preventDefault();
56
57     var name = getElementVal("name");
58     var emailid = getElementVal("emailid");
59     var msgContent = getElementVal("msgContent");
60
61     saveMessages(name, emailid, msgContent);
62     // Show alert
63     document.querySelector(".alert").style.display = "block";
64
65     // Hide alert after 3 seconds
66     setTimeout(()=>{
67         document.querySelector(".alert").style.display = "none";
68     },3000)
69
70     // reset the form
71     document.getElementById("contactForm").reset();
72
73 }
74
75
76 const saveMessages = (name, emailid, msgContent,) => {
77     var newContactForm = contactFormDB.push();
78
79     newContactForm.set({
80         name: name,
81         emailid: emailid,
82         msgContent: msgContent,
83     });
84 }
85

```

Alert implemented

Before:



Contact Now

Nice Website!

Send

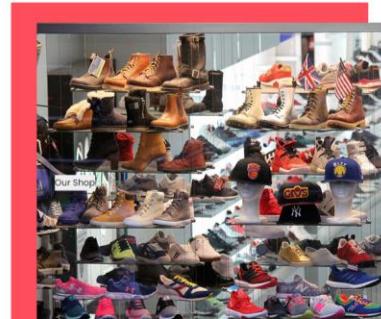


Alert function working after user clicks send, after the user clicks ok on the alert, the form will reset and the details will save to the database

After:

Contact Now

[Send](#)



Page successfully resets after clicking ok on the alert and normal contact form procedure in database continues as normal. This improvement and test was a **SUCCESS**

End-User: I am satisfied with this improvement

Links and information on items

```


Nike SB Dunk Low "Paisley"







- !\[\]\(images/star-icon.png\)
- !\[\]\(images/star-icon.png\)
- !\[\]\(images/star-icon.png\)



$89.99



<form>
<button class="buy_text" formaction="https://stockx.com/">Buy Now</button>
</form>

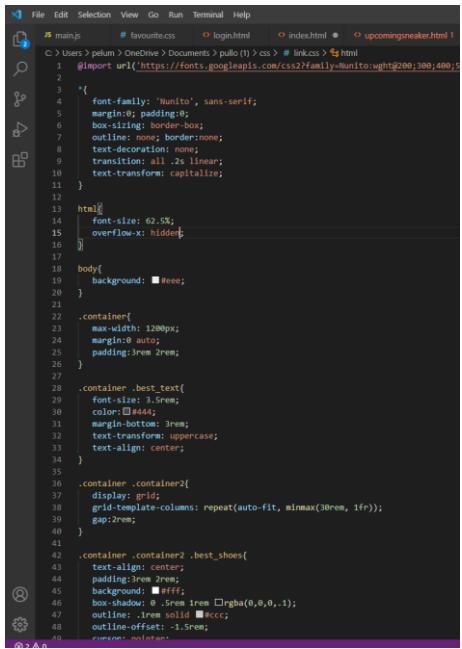

```

```
File Edit Selection View Go Run Terminal Help upcomingnikeair.html - Visual Studio Code
main.js # favourite.css login.html index.html upcomingnikeair.html # link.css script.js home.html contact.html
C:\Users\pelum\OneDrive\Documents\pullo (1)\upcomingnikeair.html\html\body.main.layout>body.main.layout>div.products-preview>div.previewactive
214     </form>
215   </div>
216 </div>
217 </div>
218 <div class="products-preview">
219   <div class="preview active" data-target="p-1">
220     <i class="fas fa-times"></i>
221     
222     <h3>Air Jordan 4 "CANVAS" </h3>
223     <div class="stars">
224       <i class="fas fa-star"></i>
225       <i class="fas fa-star"></i>
226       <i class="fas fa-star"></i>
227       <i class="fas fa-star"></i>
228       <i class="fas fa-star-half-alt"></i>
229     <span>( 250 )</span>
230   </div>
231   <p>Coming soon on the Nike app on May 31, 2022 8:00AM. As of now, check them out for the best price right now below</p>
232   <div class="price">$200</div>
233   <div class="buttons">
234     <a href="https://stockx.com/air-jordan-4-retro-blank-canvas-w" class="buy">buy now</a>
235   </div>
236 </div>
237 <div class="preview active" data-target="p-2">
238   <i class="fas fa-times"></i>
239   
240   <h3>Jordan 4 "INFLAMED" </h3>
241   <div class="stars">
242     <i class="fas fa-star"></i>
243     <i class="fas fa-star"></i>
244     <i class="fas fa-star"></i>
245     <i class="fas fa-star"></i>
246     <i class="fas fa-star"></i>
247     <i class="fas fa-star"></i>
248     <i class="fas fa-star-half-alt"></i>
249     <span>( 250 )</span>
250   </div>
251   <p>Coming soon on the Nike app on June 24, 2022 8:00AM. As of now, check them out for the best price right now below</p>
252   <div class="price">$200</div>
253   <div class="buttons">
254     <a href="https://stockx.com/air-jordan-4-retro-infrared" class="buy">buy now</a>
255   </div>
256 </div>
257 <div class="preview active" data-target="p-3">
258   <i class="fas fa-times"></i>
259   
260   <h3>Adidas Yeezy 450 "Cinder" </h3>
261 </div>
```

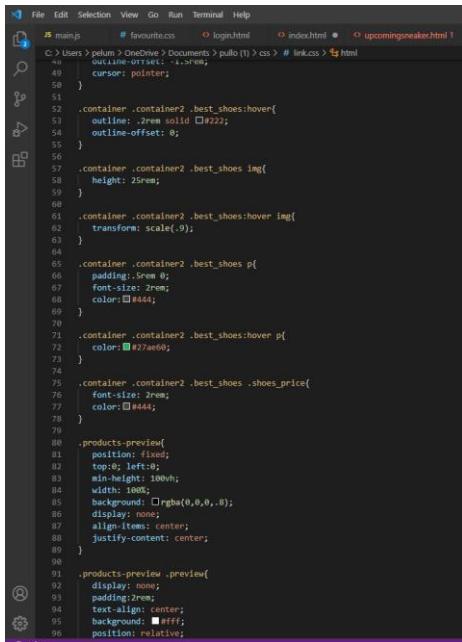
```
261   <h3>Adidas Yeezy 450 "Cinder" </h3>
262   <div class="stars">
263     <i class="fas fa-star"></i>
264     <i class="fas fa-star"></i>
265     <i class="fas fa-star"></i>
266     <i class="fas fa-star"></i>
267     <i class="fas fa-star-half-alt"></i>
268     <span>( 250 )</span>
269   </div>
270   <p>Coming soon on the Confirmed app on March 24, 2022 7:00AM. As of now, check them out for the best price right now below
271     <a href="https://www.offspring.co.uk/view/content/noSuchPage">Offspring</a>
272     <a href="https://apps.apple.com/us/app/id1500115440?mt=8">Confirmed</a>
273   </p>
274   <div class="price">$10</div>
275   <div class="buttons">
276     <a href="https://stockx.com/adidas-yeezy-450-cinder" class="buy">buy now</a>
277   </div>
278 </div>
279 <div class="preview active" data-target="p-4">
280   <i class="fas fa-times"></i>
281   
282   <h3>Adidas Yeezy V2 "Core" </h3>
283   <div class="stars">
284     <i class="fas fa-star"></i>
285     <i class="fas fa-star"></i>
286     <i class="fas fa-star"></i>
287     <i class="fas fa-star"></i>
288     <i class="fas fa-star-half-alt"></i>
289     <span>( 250 )</span>
290   </div>
291   <p>Coming soon on the Confirmed app on March 12, 2022 7:00AM. As of now, check them out for the best price right now below
292     <a href="https://apps.apple.com/us/app/id1500115440?mt=8">Confirmed</a>
293   </p>
294   <div class="price">$200</div>
295   <div class="buttons">
296     <a href="https://stockx.com/adidas-yeezy-boost-350-v2-core-black-white" class="buy">buy now</a>
297   </div>
298 </div>
299 </div>
300 </div>
```

```
File Edit Selection View Go Run Terminal Help upcomingnikeair.html - Visual Studio Code
main.js # favourite.css login.html index.html upcomingnikeair.html # link.css script.js home.html contact.html contact.html
C:\Users\pelum\OneDrive\Documents\pullo (1)\upcomingnikeair.html\html\body.main.layout>body.main.layout>div.products-preview>div.previewactive>img
301   <div class="preview active" data-target="p-5">
302     <i class="fas fa-times"></i>
303     
304     <h3>Adidas 350 V2 "Dazzling Blue" </h3>
305     <div class="stars">
306       <i class="fas fa-star"></i>
307       <i class="fas fa-star"></i>
308       <i class="fas fa-star"></i>
309       <i class="fas fa-star"></i>
310       <i class="fas fa-star-half-alt"></i>
311     <span>( 250 )</span>
312   </div>
313   <p>Coming soon on the Confirmed app on March 17, 2022 7:00AM. As of now, check them out for the best price right now below
314     <a href="https://apps.apple.com/us/app/id1500115440?mt=8">Confirmed</a>
315   </p>
316   <div class="price">$200</div>
317   <div class="buttons">
318     <a href="https://stockx.com/adidas-yeezy-boost-350-v2-dazzling-blue" class="buy">buy now</a>
319   </div>
320 </div>
321 <div class="preview active" data-target="p-6">
322   <i class="fas fa-times"></i>
323   
324   <h3>Nike SB Dunk Low "Paisley" </h3>
325   <div class="stars">
326     <i class="fas fa-star"></i>
327     <i class="fas fa-star"></i>
328     <i class="fas fa-star"></i>
329     <i class="fas fa-star"></i>
330     <i class="fas fa-star-half-alt"></i>
331     <span>( 250 )</span>
332   </div>
333   <p>Coming soon on the SNKRS app on April 5, 2022 8:00AM. As of now, check them out for the best price right now below
334     <a href="https://www.nike.com/gh/launch/notfound">SNKRS</a>
335   </p>
336   <div class="price">$89.99</div>
337   <div class="buttons">
338     <a href="https://stockx.com/nike-sb-dunk-low-pro-paisley-brown" class="buy">buy now</a>
339   </div>
340 </div>
341 </div>
342 </div>
343 </div>
344 </div>
345 </div>
346 <div class="section_footer">
347   <div class="container">
348     <div class="mail_section">
```

CSS:



```
File Edit Selection View Go Run Terminal Help
main.js # favourite.css login.html index.html upcomingsneaker.html
C:\>Users\pelum>OneDrive\Documents>pullo()>css>#linkcss>html
1 @import url('https://fonts.googleapis.com/css?family=Munito:wght@200;300;400;500;600;700;800;900');
2 *
3 {
4   font-family: 'Munito', sans-serif;
5   margin: 0; padding: 0;
6   box-sizing: border-box;
7   outline: none; border: none;
8   text-decoration: none;
9   transition: all .2s linear;
10  text-transform: capitalize;
11 }
12 html{
13   font-size: 62.5%;
14   overflow-x: hidden;
15 }
16
17 body{
18   background: #eeee;
19 }
20
21 .container{
22   max-width: 1200px;
23   margin: 0 auto;
24   padding: 0rem 2rem;
25 }
26
27 .container .best_text{
28   font-size: 3.5rem;
29   color: #444;
30   margin-bottom: 3rem;
31   text-transform: uppercase;
32   text-align: center;
33 }
34
35 .container .container2{
36   display: grid;
37   grid-template-columns: repeat(auto-fit, minmax(30rem, 1fr));
38   gap: 2rem;
39 }
40
41 .container .container2 .best_shoes{
42   text-align: center;
43   padding: 0rem 2rem;
44   border: 1px solid #ccc;
45   box-shadow: 0 .5rem 0 0 rgba(0,0,0,.1);
46   outline: 1px solid #ccc;
47   outline-offset: -1.5rem;
48   cursor: pointer;
49 }
```



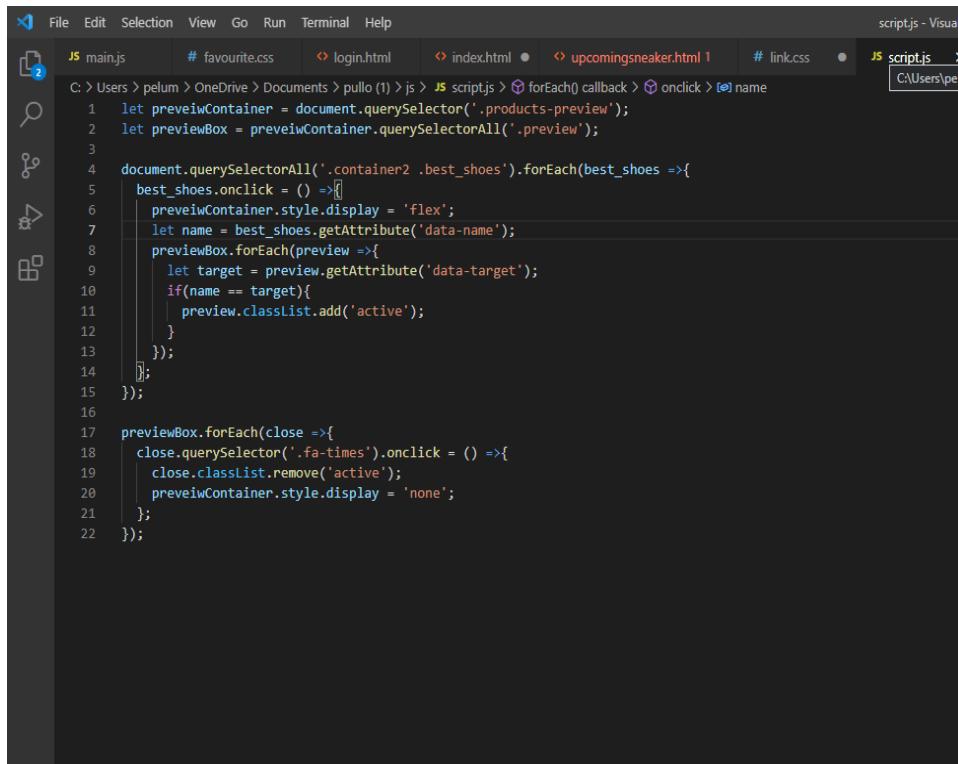
```
File Edit Selection View Go Run Terminal Help
main.js # favourite.css login.html index.html upcomingsneaker.html
C:\>Users\pelum>OneDrive\Documents>pullo()>css>#linkcss>html
50 }
51
52 .container .container2 .best_shoes:hover{
53   outline: 2rem solid #222;
54   outline-offset: 0;
55 }
56
57 .container .container2 .best_shoes img{
58   height: 25rem;
59 }
60
61 .container .container2 .best_shoes:hover img{
62   transform: scale(.9);
63 }
64
65 .container .container2 .best_shoes p{
66   padding: 5rem 0;
67   font-size: 2rem;
68   color: #444;
69 }
70
71 .container .container2 .best_shoes:hover p{
72   color: #27ae60;
73 }
74
75 .container .container2 .best_shoes .shoes_price{
76   font-size: 2rem;
77   color: #444;
78 }
79
80 .products-preview{
81   position: absolute;
82   top: 8%; left: 8%;
83   min-height: 10vh;
84   width: 100px;
85   background: #fff;
86   display: none;
87   align-items: center;
88   justify-content: center;
89 }
90
91 .products-preview .preview{
92   display: none;
93   padding: 2rem;
94   text-align: center;
95   background: #fff;
96   position: relative;
```

```
File Edit Selection View Go Run Terminal Help
main.js # favourite.css login.html index.html ● ○ upcomingnikeer.html
C > Users > pelum > OneDrive > Documents > pullo(1) > css > # link.css > 4.html
96
97     position: relative;
98     margin: 2rem;
99     width: 40rem;
100 }
101 .products-preview .preview.active{
102     display: inline-block;
103 }
104 .products-preview .preview img{
105     height: 30rem;
106 }
107 .products-preview .preview .fa-times{
108     position: absolute;
109     top: 1rem; right: 1.5rem;
110     cursor: pointer;
111     color: #444;
112     font-size: 4rem;
113 }
114 .products-preview .preview .fa-times:hover{
115     transform: rotate(90deg);
116 }
117 .products-preview .preview h3{
118     color: #444;
119     padding: 0.5rem 0;
120     font-size: 1.5rem;
121 }
122 .products-preview .preview .stars{
123     padding: 0 0 0 1.5rem;
124     font-size: 1.7rem;
125 }
126 .products-preview .preview .stars i{
127     color: #27ae60;
128 }
129 .products-preview .preview .stars span{
130     color: #999;
131 }
132 .products-preview .preview p{
133     line-height: 1.5;
134     padding: 0 0 0 1.5rem;
135     font-size: 1.6rem;
136 }
```

```
File Edit Selection View Go Run Terminal Help
main.js # favourite.css login.html index.html ● ○ upcomingnikeer.html
link.css script.js
C > Users > pelum > OneDrive > Documents > pullo(1) > css > # link.css > 4.html
141     line-height: 1.5;
142     padding: 0;
143     font-size: 1.6rem;
144     color: #777;
145 }
146 .products-preview .preview .price{
147     padding: 0;
148     font-size: 2.5rem;
149     color: #27ae60;
150 }
151 .products-preview .preview .buttons{
152     display: flex;
153     gap: 1.5rem;
154     flex-wrap: wrap;
155     margin-top: 1rem;
156 }
157 .products-preview .preview .buttons a{
158     flex: 1 16rem;
159     padding: 0;
160     font-size: 1.8rem;
161     color: #444;
162     border: 1px solid #444;
163 }
164 .products-preview .preview .buttons a.cart{
165     background: #444;
166     color: #fff;
167 }
168 .products-preview .preview .buttons a.cart:hover{
169     background: #111;
170 }
171 .products-preview .preview .buttons a.buy:hover{
172     background: #444;
173     color: #fff;
174 }
175 @media (max-width:991px){
176     html{
177         font-size: 55px;
178     }
179 }
180 }
181 @media (max-width:768px){
182     .products-preview .preview img{
183         height: 25rem;
184     }
185 }
186 @media (max-width:450px){
187     html{
188         font-size: 50px;
189     }
190 }
```

```
188     }
189 }
190 @media (max-width:768px){
191     .products-preview .preview img{
192         height: 25rem;
193     }
194 }
195 @media (max-width:450px){
196     html{
197         font-size: 50px;
198     }
199 }
```

JavaScript:



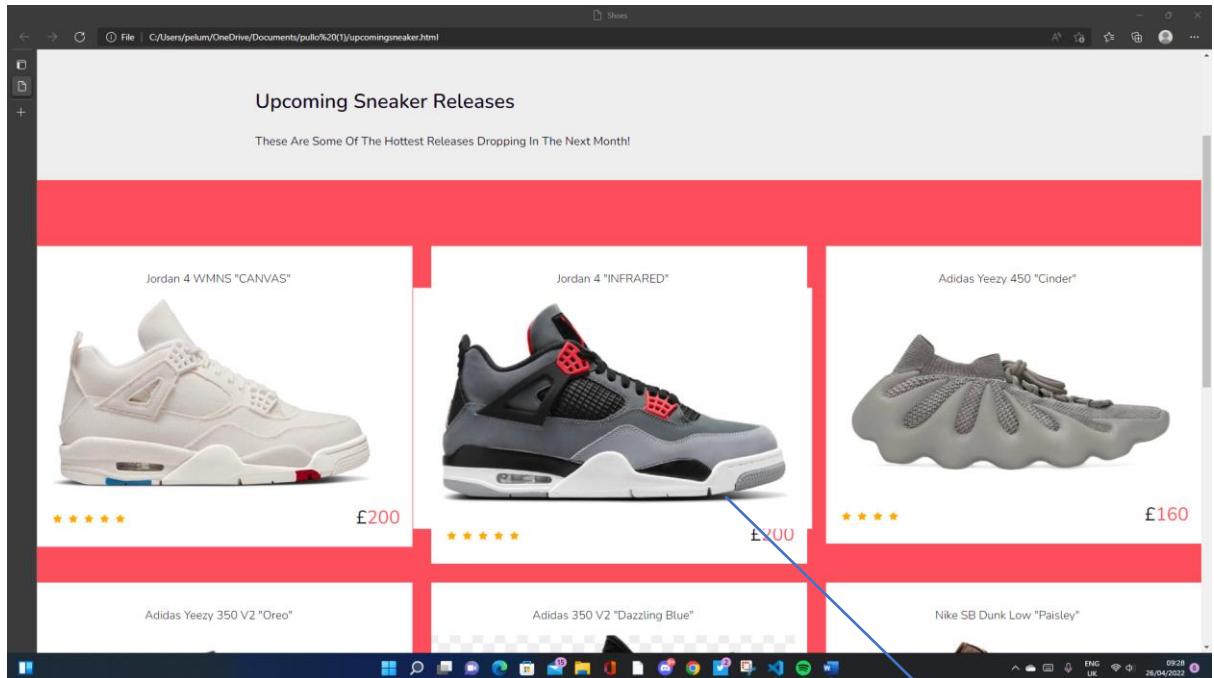
```

File Edit Selection View Go Run Terminal Help
JS main.js # favourite.css login.html index.html upcomingsneaker.html link.css script.js C:\Users\pelum
C: > Users > pelum > OneDrive > Documents > pullo () > js > JS script.js > forEach() callback > onclick > name
1 let preveiwContainer = document.querySelector('.products-preview');
2 let previewBox = preveiwContainer.querySelectorAll('.preview');
3
4 document.querySelectorAll('.container2 .best_shoes').forEach(best_shoes =>{
5   best_shoes.onclick = () =>{
6     preveiwContainer.style.display = 'flex';
7     let name = best_shoes.getAttribute('data-name');
8     previewBox.forEach(preview =>{
9       let target = preview.getAttribute('data-target');
10      if(name == target){
11        preview.classList.add('active');
12      }
13    });
14  });
15 });
16
17 previewBox.forEach(close =>{
18   close.querySelector('.fa-times').onclick = () =>{
19     close.classList.remove('active');
20     preveiwContainer.style.display = 'none';
21   };
22 });

```

Display:

Hover over item feature, when user hovers over a product, picture will enlarge. Nice aesthetic effect. Implemented in style.css file so all product pages will have this feature. CSS for this is shown below



When item is hovered over it becomes larger, this piece of code works

```

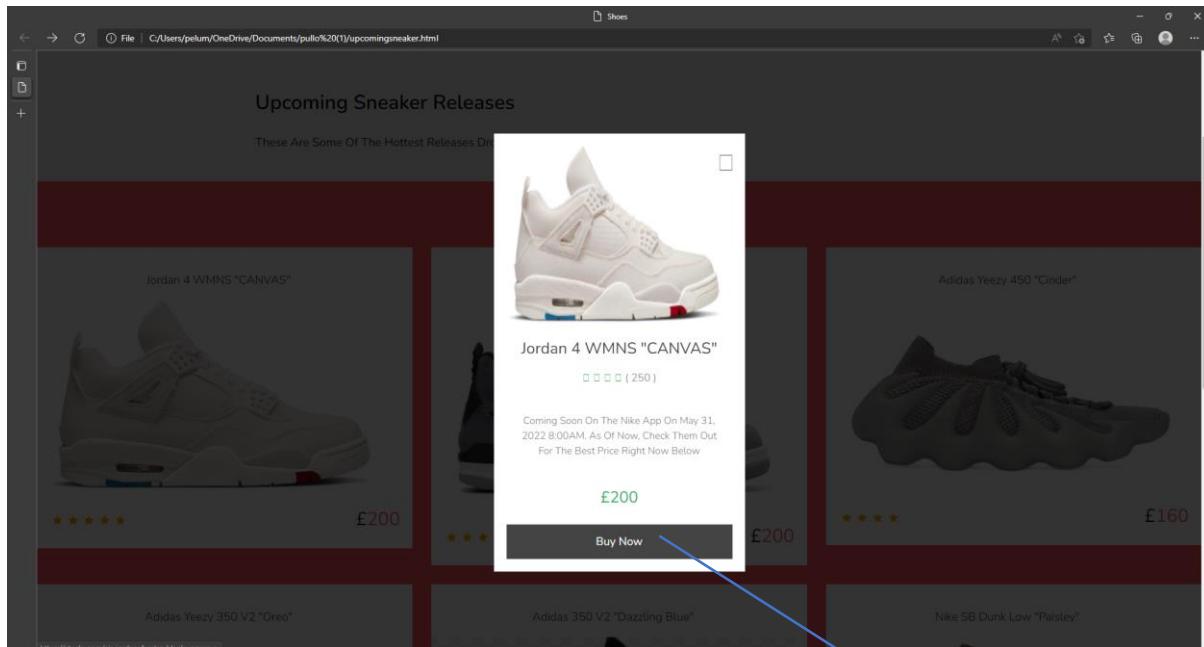
802 .header_main {
803   width: 100%;
804   height: auto;
805   background: #0c0116;
806 }
807 }
808
809 .shoes_icon img:hover{
810   transform: scale(1.1);
811 }
812 }
813 .fa-heart-o{
814   padding: 6px;
815   border-radius: 50%;
816   background: pink;
817   cursor: pointer;
818 }
819
820 .toggle-fa-heart-o{
821   background: orange;
822   color: white;
823 }
824
825
826 .error_message{
827   width: 100%;
828   background: green;
829   padding: 10px 20px;
830   border-radius: 5px;
831   text-align:center;
832   font-size: 18px;
833   font-weight:900;
834   display:none;
835 }
836 }

```

CSS code section for hover feature shown above

Pop ups will include item picture, release information and links to where it is dropping. Will also display its price and stockx link.

After clicking on item of choice:



Clicking buy now:

This link will take user to stockx, the number 1 website for purchasing shoes for a reseller price

Jordan 4 Retro Blank Canvas (W) - DQ4909-100

StockX

Search for brand, color, etc.

Browse News About Help Login Sign Up

Home / Sneakers / Air Jordan / 4 / Jordan 4 Retro Blank Canvas (W)

Jordan 4 Retro
Blank Canvas (W)

100% Authentic Condition: New

Size: All

Buy or Bid Size 7W £300

Sell or Ask Size 11.5W £351

Last Sale: £253 ▾ -£9 (-3.6%)

View Asks View Bids View Sales

Related Products

Jordan 4 Retro Shimmer (W) Jordan 4 Retro Wahlburgers Jordan 4 Retro Manila Jordan 4 Retro Silt Red Splatter (W) Jordan 4 Retro Starfish (W)

We use cookies to understand how you use our site and to improve your experience. By continuing to use our website or by closing this notice you acknowledge that you agree to our use of cookies and other tracking technologies. To learn more, please click on Privacy Policy.

Privacy Policy

This is the cancel icon so you can cancel the pop up and go back to the product page after viewing.

Buy Now

Coming Soon On The Confirmed App On March 24, 2022 7:00AM. As Of Now, Check Them Out For The Best Price Right Now Below Offspring Confirmed

£160

Two retail links for the websites that are releasing this shoe on the day described in the pop up

Screenshot of the App Store Preview for the adidas CONFIRMED app:

<https://apps.apple.com/us/app/adidas-confirmed/id1508115448?mt=8>

adidas CONFIRMED on the App Store

Store Mac iPad iPhone Watch AirPods TV & Home Only on Apple Accessories Support

App Store Preview

This app is available only on the App Store for iPhone.

adidas CONFIRMED 4.4+
Shop latest Adidas sneakers
adidas
#147 in Shopping
4.5+ 2.2K Ratings
Free

iPhone Screenshots

NEVER MISS A DROP
CONFIRMED
INSIDER STORIES
NO TWO AQUAS ARE THE SAME
ADIDAS ORIGINALS BY HUMAN MADE
EXCLUSIVE PRODUCTS

Screenshot of the OFFSPRING website showing a 404 error page:

It looks like the page you are looking for no longer exists!
Click [here](#) to visit the homepage

OFFSPRING

SHOP NEW IN 10% STUDENT DISCOUNT SHOP SALE

NEWIN MEN WOMEN BRANDS SALE FEATURED

Get in on the latest news, reviews & products:
Get the weekly OFFSPRING newsletter

Women Men

Email

[View our Privacy Policy](#)

Website images for the shoes on that link. The shoe has already been released so he website doesn't load with anything and for adidas, it directs the user to download the confirmed app which is where the shoe will be releasing, usually conducted by a raffle.

adidas Yeezy 450
Cinder

100% Authentic Condition: New

Size: All

Buy or Bid Size 4.5 £159

Sell or Ask Size 14 £203

Last Sale: £184 ▲ £4 (2.2%)

View Asks View Bids View Sales

Related Products

- adidas Yeezy 450 Cinder
- adidas Yeezy 450 Cinder
- adidas Yeezy 450 Resin
- adidas Yeezy 450 Sulfur
- adidas Yeezy Boost 350

Stockx link for these shoes. All items will have a stockx link linking to their own item and this will show the price that the item is selling for for every size. This is useful for resellers if they're selling a specific size or for casual buyers if they need a specific size.

| Item | Release Date | Price | Action |
|-------------------------------|---|--------|---------|
| Jordan 4 WMNS "CANVAS" | Coming Soon On The Nike App On May 31, 2022 8:00AM. As Of Now, Check Them Out For The Best Price Right Now Below | £200 | Buy Now |
| Jordan 4 "INFRARED" | Coming Soon On The Nike App On June 24, 2022 8:00AM. As Of Now, Check Them Out For The Best Price Right Now Below | £200 | Buy Now |
| Adidas Yeezy 450 "Cinder" | Coming Soon On The Confirmed App On March 24, 2022 7:00AM. As Of Now, Check Them Out For The Best Price Right Now Below Offspring Confirmed | £160 | Buy Now |
| Adidas Yeezy 350 V2 "Oreo" | Coming Soon On The Confirmed App On March 12, 2022 7:00AM. As Of Now, Check Them Out For The Best Price Right Now Below Confirmed | £200 | Buy Now |
| Adidas 350 V2 "Dazzling Blue" | Coming Soon On The Confirmed App On March 17, 2022 7:00AM. As Of Now, Check Them Out For The Best Price Right Now Below Confirmed | £200 | Buy Now |
| Nike SB Dunk Low "Paisley" | Coming Soon On The SNKRS App On April 5, 2022 8:00AM. As Of Now, Check Them Out For The Best Price Right Now Below SNKRS | £89.99 | Buy Now |

Button clicked to go see the item on stockx, streetwear items won't be included as my streetwear brands are not on stockx

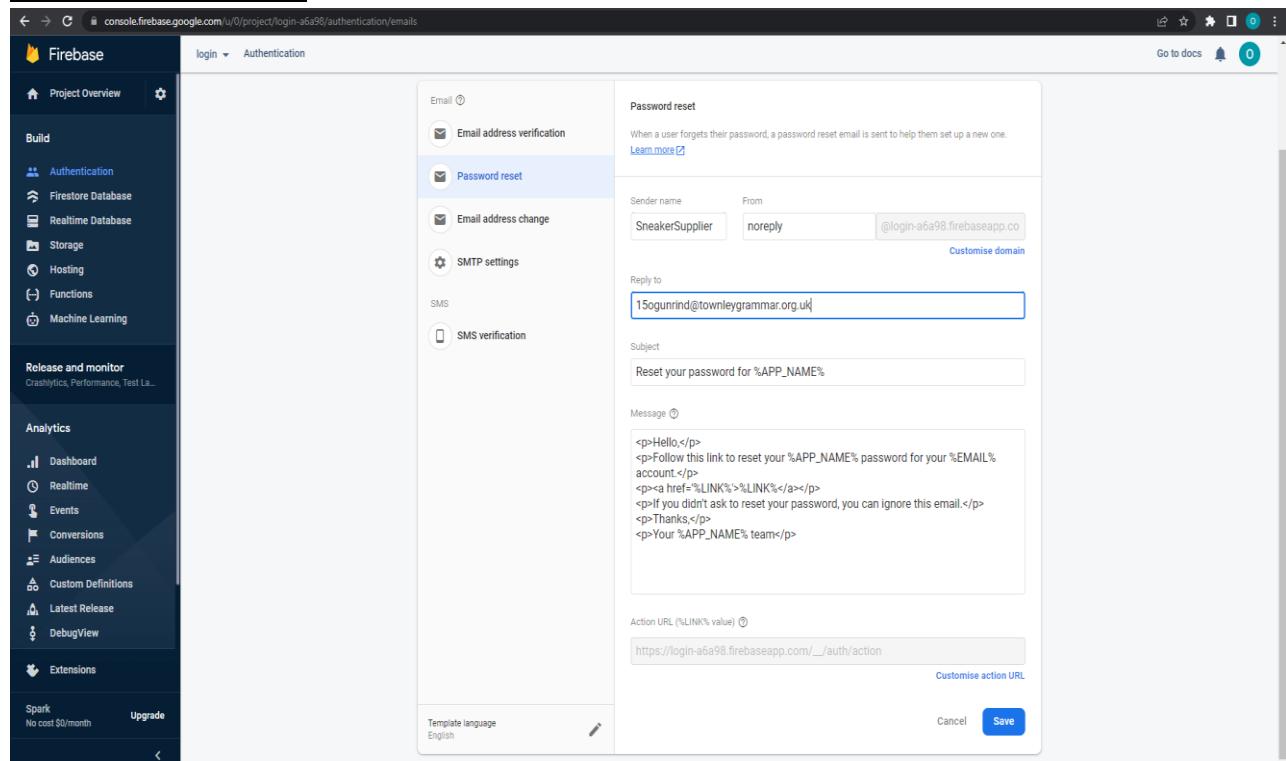
Link taking user to location where they can purchase item

Release date and information for the user of when and where it is dropping.

Richard Chinedu: I am very satisfied with this feature...

TEST REVIEW: SUCCESS

Resetting password



The screenshot shows the Firebase console's Authentication settings page. On the left, there's a sidebar with Project Overview, Build (Authentication selected), Release and monitor, Analytics, and Extensions. Under Authentication, it lists Email address verification, Password reset (selected), Email address change, SMTP settings, and SMS verification. The main area shows the configuration for the Password reset template. It includes fields for Sender name (SneakerSupplier), From (noreply), Reply to (15ogunrind@townleygrammar.org.uk), Subject (Reset your password for %APP_NAME%), and Message (HTML content). The message content is:

```
<p>Hello,</p>
<p>Follow this link to reset your %APP_NAME% password for your %EMAIL% account.</p>
<p><a href="%LINK%">%LINK%</a></p>
<p>If you didn't ask to reset your password, you can ignore this email.</p>
<p>Thanks,</p>
<p>Your %APP_NAME% team</p>
```

Action URL (%LINK% value) is set to https://login-a6a98.firebaseio.com/_/auth/action. There are Save and Cancel buttons at the bottom.

This is the format for how the email are going to be sent using firebase. The name will be SneakerSupplier and it will show the sender being the no reply email, indicating that the user should not respond to the email. All replies however will still be sent to my school email and the email content and paragraph are to tell the user what to do is displayed below along with the link of where you are to reset your password

Display for resetting password page, corresponds with log in page theme so it keeps the consistency of the aesthetic of the website.

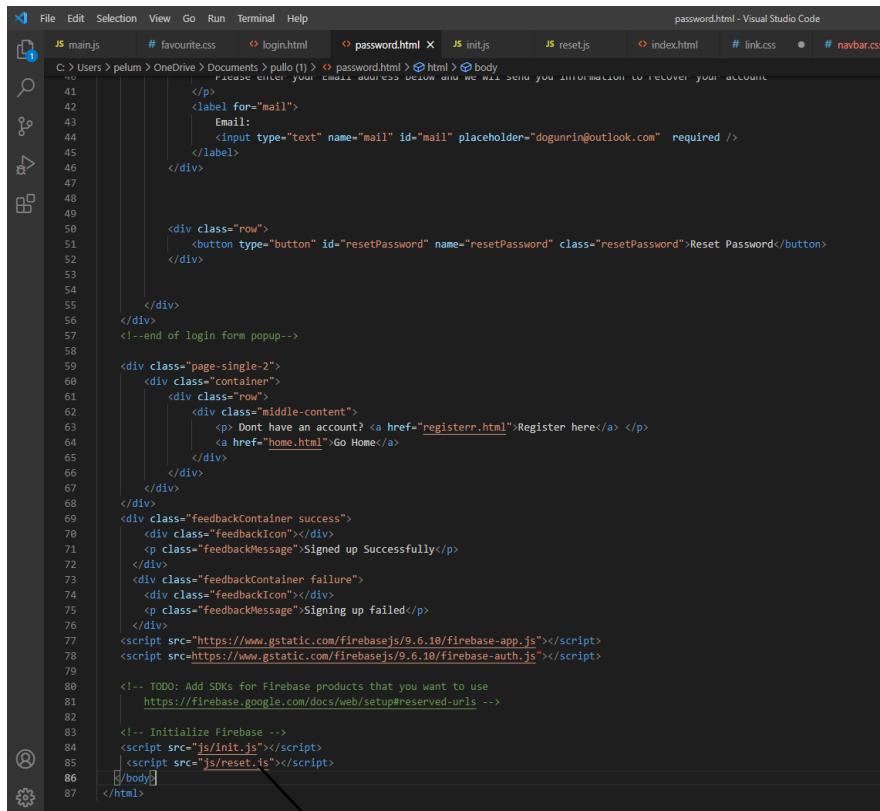
drive/Documents/pullo%20(1)/password.html

The screenshot shows a web page with the following elements:

- Logo:** A black oval containing a white sneaker icon.
- Title:** "THE SNEAKER SUPPLIER" centered below the logo.
- Text:** "Reset Password" in a grey bar at the top.
- Form:** An input field labeled "Email:" with the placeholder "dogunrin@outlook.com".
- Buttons:** A "Reset Password" button below the email input.
- Text:** "Please enter your Email address below and we wil send you information to recover your account".
- Links:** "Dont have an account? Register here" and "Go Home".
- Annotations:**
 - A callout box points to the title: "Title of the page, shows the purpose".
 - A callout box points to the "Information the informing the user how to reset their password and how the process will go": "Information the informing the user how to reset their password and how the process will go".
 - A callout box points to the "Links back to the register page and home page if the user ended up here accidentally": "Links back to the register page and home page if the user ended up here accidentally".

```
C:\> Users > pelum > OneDrive - Documents > pullo (1) > password.html > html > body
20   <link rel="stylesheet" href="css/plugins.css">
21   <link rel="stylesheet" href="css/style.css">
22
23 </head>
24 <body>
25
26   <div id="preloader">
27     
28     <div id="status">
29       <span></span>
30       <span></span>
31     </div>
32   </div>
33   <!--end of preloading-->
34   <!--login form popup-->
35   <div class="Login-wrapper" id="login-content">
36     <div class="Login-content">
37       <h3>Reset Password</h3>
38       <div class="row">
39         <p>
40           Please enter your Email address below and we wil send you information to recover your account
41         </p>
42         <label for="mail">
43           Email:
44           <input type="text" name="mail" id="mail" placeholder="dogunrin@outlook.com" required />
45         </label>
46       </div>
47
48
49       <div class="row">
50         <button type="button" id="resetPassword" name="resetPassword" class="resetPassword">Reset Password</button>
51       </div>
52   </div>
53   <!--end of login form popup-->
54
55   <div class="page-single-2">
56     <div class="container">
57       <div class="row">
58         <div class="middle-content">
59           <p> Dont have an account? <a href="register.html">Register here</a> <a href="home.html">Go Home</a> </p>
60         </div>
61       </div>
62     </div>
63   </div>
```

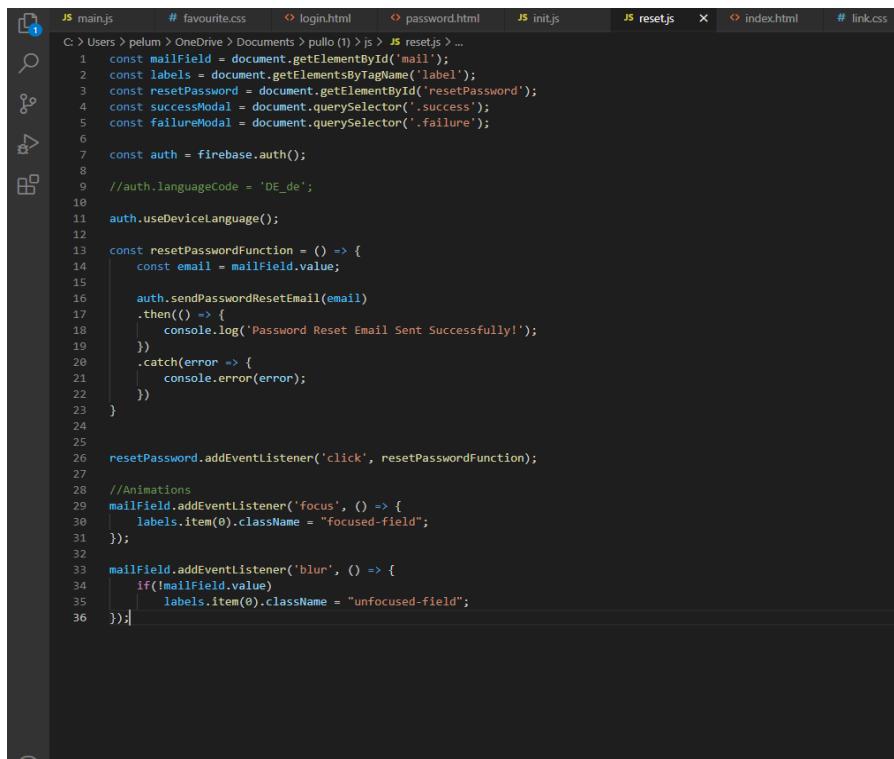
HTML IMPLEMENTATION :



```
C:\> Users > pelum > OneDrive > Documents > pulo () > password.html > html > body
    Please enter your email address below we will send you information to recover your account
41     </p>
42     <label for="mail">
43         Email:
44         <input type="text" name="mail" id="mail" placeholder="dogunrin@outlook.com" required />
45     </label>
46   </div>
47
48
49   <div class="row">
50     <button type="button" id="resetPassword" name="resetPassword" class="resetPassword">Reset Password</button>
51   </div>
52
53
54   </div>
55 </div>
56 <!-- end of login form popup-->
57
58 <div class="page-single-2">
59   <div class="container">
60     <div class="row">
61       <div class="middle-content">
62         <p> Dont have an account? <a href="register.html">Register here</a> </p>
63         <a href="home.html">Go Home</a>
64       </div>
65     </div>
66   </div>
67
68 </div>
69 <div class="feedbackContainer success">
70   <div class="feedbackIcon"></div>
71   <p class="feedbackMessage">Signed up Successfully</p>
72 </div>
73 <div class="feedbackContainer failure">
74   <div class="feedbackIcon"></div>
75   <p class="feedbackMessage">Signing up failed</p>
76 </div>
77 <script src="https://www.gstatic.com/firebasejs/9.6.10/firebase-app.js"></script>
78 <script src="https://www.gstatic.com/firebasejs/9.6.10.firebaseio-auth.js"></script>
79
80 <!-- TODO: Add SDKs for Firebase products that you want to use
81      https://firebase.google.com/docs/web/setup#presigned-urls -->
82
83 <!-- Initialize Firebase -->
84 <script src="js/init.js"></script>
85 <script src="js/reset.js"></script>
86 </body>
87 </html>
```

External files linking
to the JavaScript
pages displayed
below

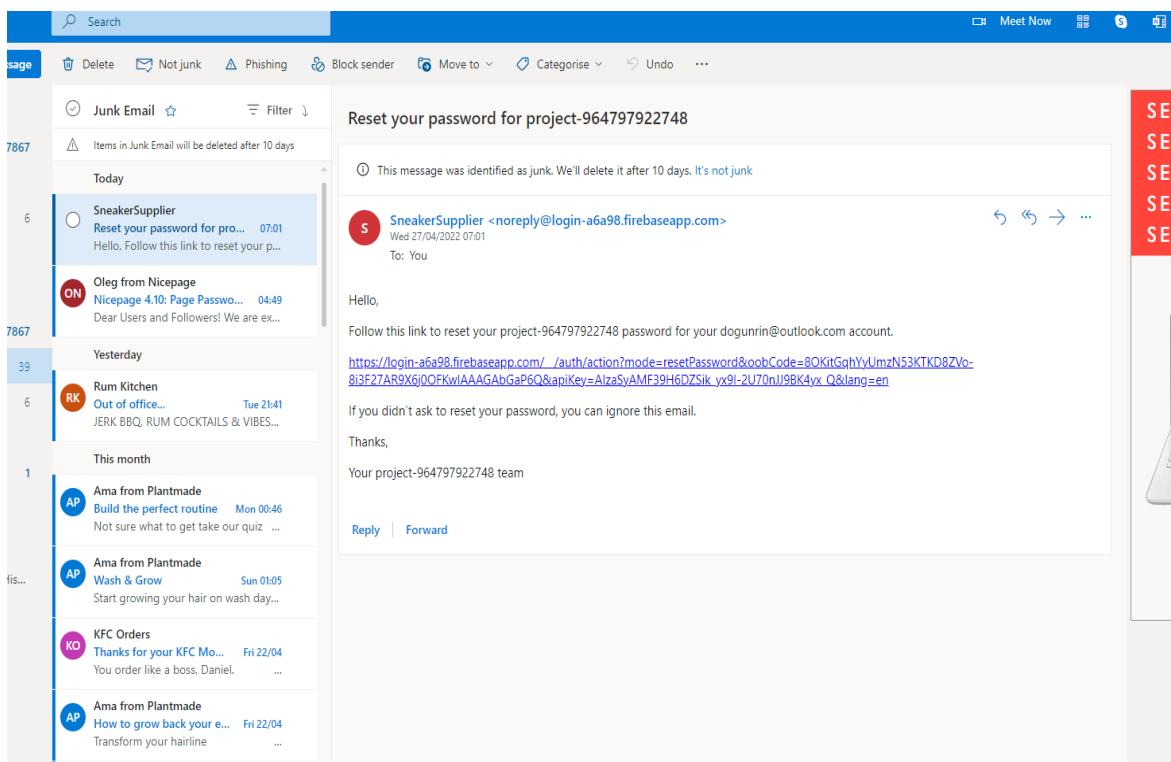
The actual javascript for trying to implement the reset password function:



```
JS main.js      # favourite.css    < login.html    < password.html    JS init.js    JS reset.js    < index.html    # link.css
C:\Users\pelum>OneDrive>Documents>pullo (1)>js> JS reset.js > ...
1 const mailField = document.getElementById('mail');
2 const labels = document.getElementsByTagName('label');
3 const resetPassword = document.getElementById('resetPassword');
4 const successModal = document.querySelector('.success');
5 const failureModal = document.querySelector('.failure');
6
7 const auth = firebase.auth();
8
9 //auth.languageCode = 'DE_de';
10
11 auth.useDeviceLanguage();
12
13 const resetPasswordFunction = () => {
14   const email = mailField.value;
15
16   auth.sendPasswordResetEmail(email)
17     .then(() => {
18       console.log('Password Reset Email Sent Successfully!');
19     })
20     .catch(error => {
21       console.error(error);
22     })
23 }
24
25
26 resetPassword.addEventListener('click', resetPasswordFunction);
27
28 //Animations
29 mailField.addEventListener('focus', () => {
30   labels.item(0).className = "focused-field";
31 });
32
33 mailField.addEventListener('blur', () => {
34   if(mailField.value)
35     labels.item(0).className = "unfocused-field";
36 })
```

The alert function failed to implement after clicking the reset password button but a reset email was still sent successfully if the user checked their email. The link worked successfully and the user was able to reset their password and it updated in the database, shown below:

TEST



Reset your password for project-964797922748

This message was identified as junk. We'll delete it after 10 days.

SneakerSupplier <no-reply@login-a6a98.firebaseio.com>
Wed 27/04/2022 07:01
To: You

Hello,

Follow this link to reset your project-964797922748 password for your dogunrin@outlook.com account.

https://login-a6a98.firebaseio.com/_auth/action?mode=resetPassword&oobCode=8OKtGghYUmxN53KTKD8ZV0-8i3F27AR9X6j0OFkwIAAAGAbGaP6Q&apiKey=AIzaSyAMF39H6D2Sik_yx9l-2U70nj9BK4yx_Q&lang=en

If you didn't ask to reset your password, you can ignore this email.

Thanks,

Your project-964797922748 team

As shown, the reset email sent immediately and successfully to my email with the correct format that I styled before, with the name being from SneakerSupplier and t including my email in the paragraph. The link was successfully included and when clicked on:

https://www.sneakersupplier.com/auth/password/reset?token=8UkitGqnYyUmzN53TKD8ZVo-8j5F27AR9X6j0fKwIAAAAGAbGaPbQ&apiKey=AizasyAMF59H6D2Stk_y9I-ZU70njj9bK4yx_Q8t

The image shows a password reset interface. At the top, it says "Reset your password" and "for dogunrin@outlook.com". Below that is a "New password" input field with a small circular icon containing a dot to its right. At the bottom is a blue rectangular button labeled "SAVE".

This page was displayed where you could easily change your password

?oobCode=80KitGqhYyUmzN53TKD8ZVo-8i3F27AR9X6j0OFKwlAAAGAbGaP6Q&apiKey=AlzaSyAMF39H6DZSik_yx9I-2U70nJJ9BK4yx_Q&lang=en

Reset your password
for **dogunrin@outlook.com**

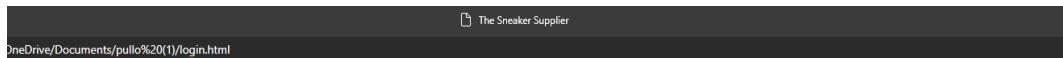
New password
 

SAVE

This shows me inputting my new changed password in, eye toggled so you can see what I am resetting it to so you can see the test is successful when checking the new password in the database



Alert saying the password was changed displays when clicking okay and to check this I tried to log in with my old database and my new database:



THE SNEAKER SUPPLIER

Login

Email:

Password:

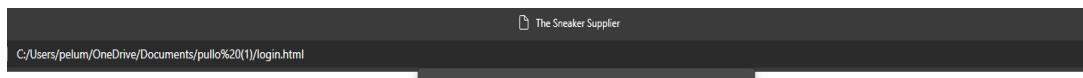
Remember me

[Forgot password ?](#)

[Log In](#)

[Dont have an account? Register here](#)

[Go Home](#)



THE SNEAKER SUPPLIER

Login

Email:

Password:

Remember me

[Forgot password ?](#)

[Log In](#)

[Dont have an account? Register here](#)

[Go Home](#)

This shows that my old password does not successfully log in to my system anymore. To fully view this as a success, I also tried to log in with my new password which was shown before:

The Sneaker Supplier

e/Documents/pullo%20(1)/login.html



THE SNEAKER SUPPLIER

Login

Email: Password:

Remember me
[Forgot password?](#)

[Don't have an account? Register here](#)

[Go Home](#)

New password

An arrow points from the "New password" box to the "Log In" button.

The Sneaker Supplier

OneDrive/Documents/pullo%20(1)/login.html

This page says

user logged in successfully

THE SNEAKER SUPPLIER

Login

Email: Password:

Remember me
[Forgot password?](#)

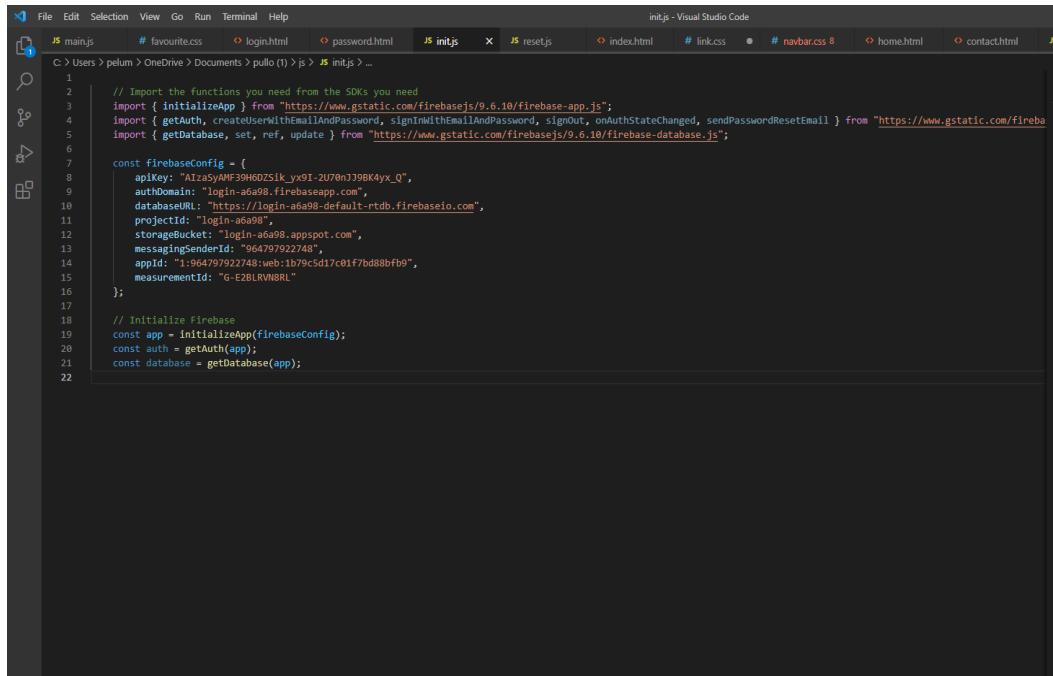
[Don't have an account? Register here](#)

[Go Home](#)

After changing my password using my reset password function after using this page from the forgot password link on my home page, it was successfully changed and allowed me to login with these new password and disallowed my old password. This proves this test of changing user details a

SUCCESS

Firebase initialisation which was referenced in the reset password HTML page



```

File Edit Selection View Go Run Terminal Help
JS main.js # favourite.css login.html password.html JS init.js x JS reset.js index.html # link.css . # navbar.css 8 home.html contact.html JS
C: > Users > pelum > OneDrive > Documents > pull (1) > js > JS init.js > ...
1 // Import the functions you need from the SDKs you need
2 import { initializeApp } from "https://www.gstatic.com/firebasejs/9.6.10/firebase-app.js";
3 import { getAuth, createUserWithEmailAndPassword, signInWithEmailAndPassword, signOut, onAuthStateChanged, sendPasswordResetEmail } from "https://www.gstatic.com/firebasejs/9.6.10/firebase-auth.js";
4 import { getDatabase, set, ref, update } from "https://www.gstatic.com/firebasejs/9.6.10.firebaseio.js";
5
6 const firebaseConfig = {
7   apiKey: "AIzaSyAMP39H60ZSik_yx9I-2U70nJJ9Bk4yx_Q",
8   authDomain: "login-a6a98.firebaseio.com",
9   databaseURL: "https://login-a6a98.firebaseio.com",
10  projectId: "login-a6a98",
11  storageBucket: "login-a6a98.appspot.com",
12  messagingSenderId: "964797922748",
13  appId: "1:964797922748:web:1b79c5d17c0f7bd88fb9",
14  measurementId: "G-E2BLRVN8RL"
15 };
16
17 // Initialize Firebase
18 const app = initializeApp(firebaseConfig);
19 const auth = getAuth(app);
20 const database = getDatabase(app);
21
22

```

End-User: Even though this is a feature that I did not ask for and expect, it was very useful to use and very successfully implemented

4 - Final Testing

4.1 - Test Plan

To do a final test of my app, I will follow my test plan created in the design section using the suggested tests and expected outcome to see if my app is successful in each of these tests. Each of the tests will have the test data used and the outcome that is expected for that set of data. I will then analyse if this test was successful or unsuccessful and assign it a pass or fail with an explanation. I will then present evidence for the test – these will be references to page numbers in which I have tested these during my development. Some of the tests from the original plan may need to be altered due to the changes I have had to make during the development, for example, the user creates an account with an email now instead of the originally proposed idea of a username – the test is still virtually the same, but now needs to be changed to test an email rather than a username.

| Test no. | Test Data | Expected Outcome | Pass or fail | Evidence |
|----------|--|---------------------------|------------------------------|------------|
| T1 | Email: dogunrin@outlook.com | The website should create | Pass Data is successfully | Page 86-96 |

| | | | | |
|----|--|---|--|---|
| | ok.com Password: Ogunrin123 | the account and save the data to the database | saved, and account is created | |
| T2 | Email: dogunrin@outlook.com Password: Ogunrin123 | Website will tell user account with email already exists, won't create a new account | Pass Email must be unique or else account is not created | Page 86-96 |
| T3 | | Upon clicking create account user will be alerted that the password must meet the criteria and the account will not be created unless they do | Pass Password must be at least 6 characters message will be displayed as alert if criteria not met | Page 86 |
| T4 | Name: Daniel Ogunrin Email: dogunrin@outlook.com Message: Fantastic Website! | Values in contact message fields will be saved to the database only if validation criteria are met | Pass Data saved to database successfully, validation working as user can't send message without criteria met or if there is nothing in one of the required fields | Page 152-156 |
| T5 | Navigating through the website from the log in page | Easy navigation through website and when clicking on category or see more button, taken straight to the HTML page. | Pass All buttons and features that can take user to separate page works successfully and easy scrolling with scrollbar easy to use | Everywhere in my code where I click on links to take me to different pages, e.g clicking 'Go Home' successfully takes you to home page, shown in my development |
| T6 | Jordan 4 | Enter desired product and enter | Fail Search field and button implemented but does not display products upon searching | Page 152 |
| T7 | Click on Jordan 5 Retro Blue | Favouriting feature should only work if user is logged in | Partial Success Favourite/heart icon was implemented on all items and when clicked the favourites list seemed to be updating as shown by the index next to the | Page 155-159 |

| | | | | |
|-----|--|---|--|--------------|
| | | | heart icon at the top of the page, but upon clicking the icon to show the favourites list it did not work. Because of this I can deem this a partial success. Also the function to save only when logged in did not work either. | |
| T8 | Items favourited should display | Favourites list displayed when icon clicked | Fail List of favourited items failed to display when clicking the favourites list icon | Page 142-149 |
| T9 | | Pop up when shoe releases | Fail System failed to implement | Page.. |
| T10 | Yeezy Dazzling Blue, Dunk Chlorophyll, Trapstar, Corteiz | Carousel will move as desired and display most popular products with links to their pages | Pass Carousel moves automatically displaying key information | Page 108 |
| T11 | Dunk Black White on the home page | All the hottest products with their pictures should be displayed on the home page | Pass Home page was stacked with hot products and their pictures were all displayed alongside information | Page 110 |
| T12 | Email: dogunrin@outlook.com Password: random password | Will log in successfully or if not matching will say incorrect details try again | Pass Log in function worked successfully and if data is not matching when compared to data in the database, log in will fail and display incorrect details | Page 86-96 |
| T13 | | Log in or create account screen will be loaded after clicking on certain link on each page | Pass If user already had an account and was on the register page. There would be a link to take them to the log in page which worked successfully and there was also a link on the log in page to the register page which also worked | Page 95 |
| T14 | | User will be logged out and loaded back into home page or will be taken to screen where password can be changed | Partial Success If the user wished to change their password, they could click forgot password on the home page and they would be taken to the page to reset the password and will input their details and an email would be sent | Page 171 |
| T15 | | The app should return to the | Fail Logout function failed to | Page... |

| | | | | |
|-----|---------------------------------------|---|---|--------------|
| | | login/create account screen and the database should store all the same The information it had before the user logged out | implement when trying to click the button | |
| T16 | Click on Adidas 350 to display pop up | Upon clicking a product, a pop up should display where the user can click links for the product to go and purchase them | Pass Clicking on the item would display pop up with release information and links to where you can buy it. Links worked successfully | Page 161-169 |
| | | | | |

4.2 - Usability Testing

To test the usability of the app, the best way to do this is by asking the users of the app. Since I have been developing it, I know exactly how it is supposed to work and therefore know how to use all the features and what each of the buttons do etc. To test the usability, I need to present the app to people who have not seen it before and assess how easy they found the app to use, whether everything still worked with potentially unexpected inputs and whether they would use it in the future and if it helps them – since this is the main goal. To do this, I asked a group of people into shoes of different ages to use the app and answer a questionnaire. I allowed the users to use the app freely (i.e., I did not give them any plan to follow and allowed them to use it how they feel fit). At the end of each questionnaire, I asked each person if they could give any extra feedback and assess what they think worked well, what didn't work so well, what was difficult, what was straight forward and what they felt may be missing and may be a useful feature in further development.

User Review Questionnaire

These are the questions that I have formed and sent to ask people who have used the website to review their experience on the website

Review

Form description



Did you find it straight forward to create an account for the app?

Yes

No

Did you find it straight forward to log in to the app?

Yes

No

Does the home screen show useful features and information when opening the app?

Yes

No

Was it easy to navigate the app?

Yes

No

[Questions](#) [Responses](#) [Settings](#)

Did you find the favouriting items feature useful and was it easy to use?

Yes - and it was easy to use

No

Yes - but it was difficult to use

Was information on product release times useful and accurate?

Yes

No

Were you able to log out successfully?

Yes

No

Were the links to websites where you can buy useful - and why?
Long answer text

Was there a good variety of products and were they all popular products

Yes - All the hottest products that I desire were there

Was there a good variety of products and were they all popular products

Yes - All the hottest products that I desire were there

Yes - there was some popular products that you could have added

No - but the products that were there were popular

No

Did you make use of the contact screen? If so was it helpful?

No, I did not use it.

Yes, I used it - it was helpful

Yes, I used it - it was not helpful

Did you change any of your personal information? If so, was it easy to do?

No, I did not change them

Yes, I did - it was easy

Yes, I did - it was difficult

How would you rate the attractiveness of the design of the app?

How would you rate the attractiveness of the design of the app?

1 2 3 4 5

Very Unattractive Very Attractive

How would you rate the ease of use of the app?

1 2 3 4 5

Hard to use Easy to use

How would you rate the usefulness of the app?

1 2 3 4 5

Useless Very Useful

Would you use this website in the future? If so, how regularly?

Every day
 A few times a week
 Once a month
 No I would not use it

No I would not use it

Did the app crash/close unexpectedly while you were using it? If so, when?

Short answer text

Rate the overall app out of 10

1 2 3 4 5 6 7 8 9 10

Please list any other feedback you have (e.g. what features do you feel are missing, what was especially good/helpful, what was particularly bad/difficult, did anything not work etc.)

Long answer text

Results:

Did you find it straight forward to create an account for the app?

8 responses

 Copy

● Yes
● No

100%

Did you find it straight forward to log in to the app?

8 responses

 Copy

● Yes
● No

100%

Does the home screen show useful features and information when opening the app?

8 responses

 Copy

● Yes
● No

100%

Was it easy to navigate the app?

8 responses

 Copy

● Yes
● No

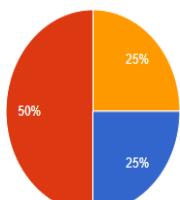
100%

Did you find the favouriting items feature useful and was it easy to use?

8 responses

 Copy

● Yes - and it was easy to use
● No
● Yes - but it was difficult to use



Was information on product release times useful and accurate?

Copy

8 responses

100%

● Yes
● No

Were you able to log out successfully?

Copy

8 responses

100%

● Yes
● No

Were the links to websites where you can buy useful - and why?

8 responses

Yes, took me straight to a website where i could purchase the items and the best price link was at the top.

They were very helpful, quick and efficient and took me straight to the purchase page. I utilised it a lot

Were the links to websites where you can buy useful - and why?

8 responses

Yes, took me straight to a website where i could purchase the items and the best price link was at the top.

They were very helpful, quick and efficient and took me straight to the purchase page. I utilised it a lot

Yes they were very helpful abd saved me time in having to search for the product myself

Yes, all website links worked successfully and took me to places where I could buy these shoes, some places where I didn't know sold the shoes before.

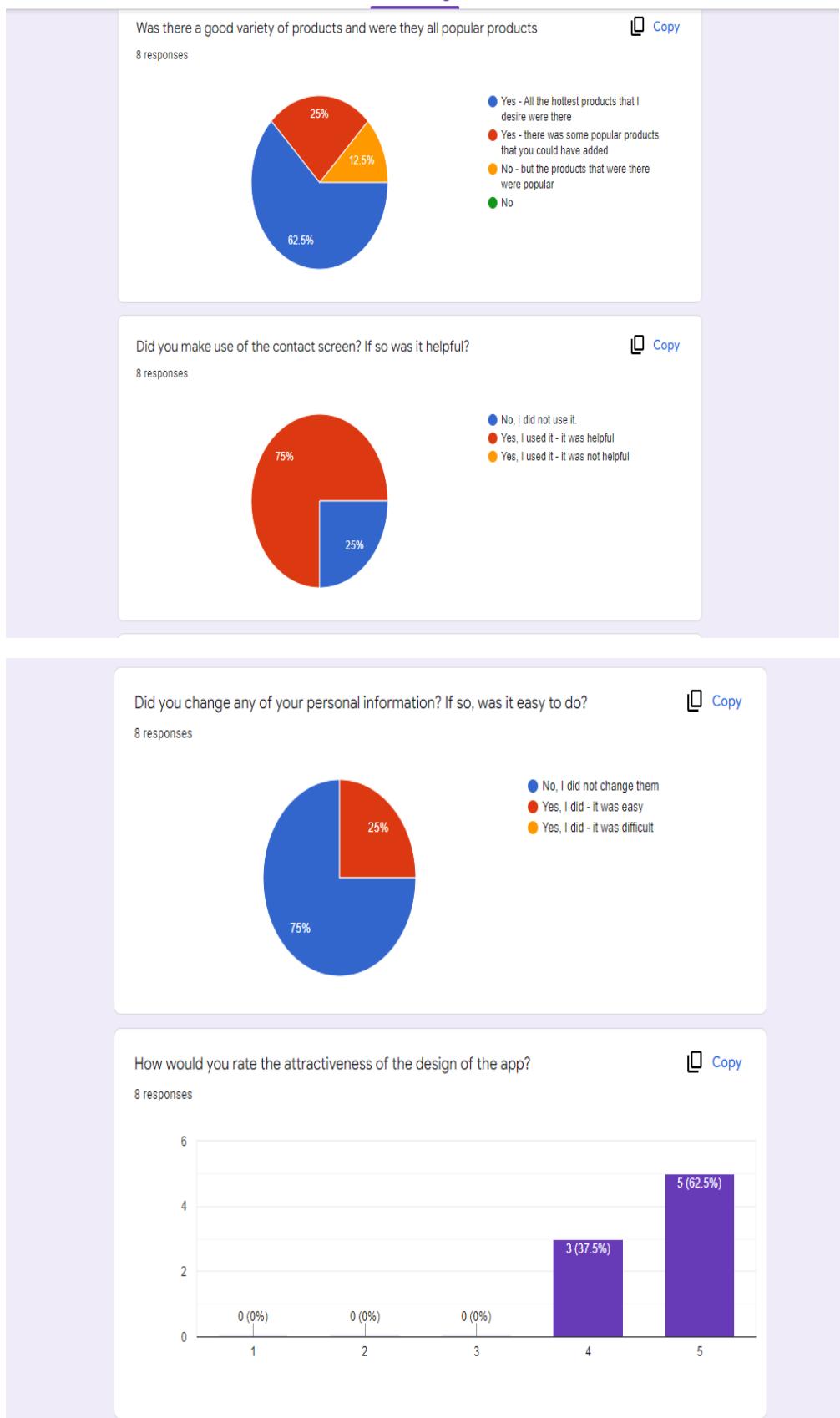
When I clicked on the product that I wanted the item information and links were displayed which was a very useful feature. When clicking the buy now button it was taken to stockx where I could buy it off resell price, but also the retail store links if I manage to get them in time. Very useful as it saves time when wanting to buy a shoe as I don't have to search myself

Very useful because it tells me where things are selling at retail and if not it gives me a site where i can buy for a reseller price.

Yes

Yes

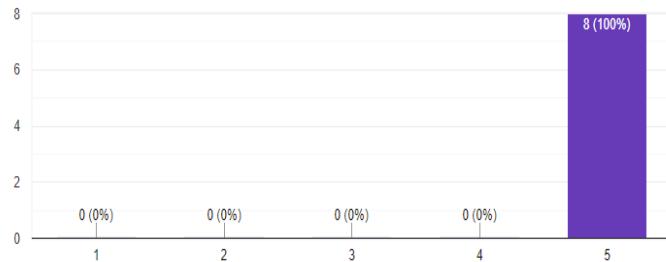
Yes because as a reseller I can quickly view the current resale price of a product and can use this information to mold prices for this shoe for my business it also helps me sell items for correct items as I can check the average price it's selling at from these websites.



How would you rate the ease of use of the app?

 Copy

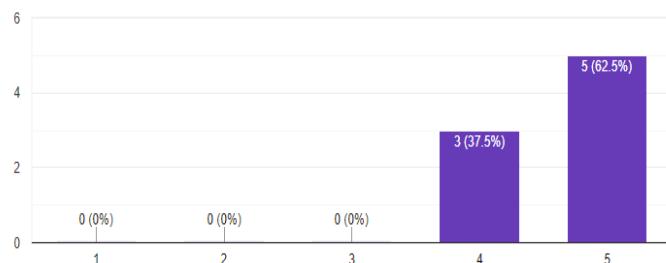
8 responses



How would you rate the usefulness of the app?

 Copy

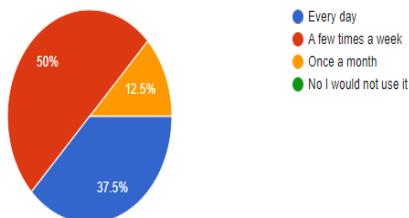
8 responses



Would you use this website in the future? If so, how regularly?

 Copy

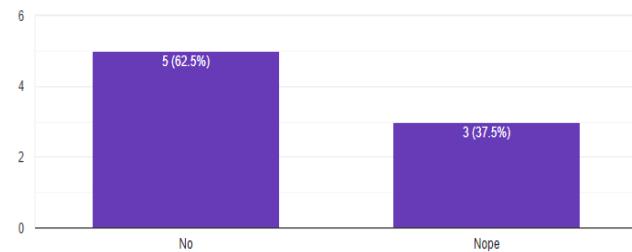
8 responses



Did the app crash/close unexpectedly while you were using it? If so, when?

 Copy

8 responses



Rate the overall app out of 10

8 responses

| Rating | Count | Percentage |
|--------|-------|------------|
| 1 | 0 | (0%) |
| 2 | 0 | (0%) |
| 3 | 0 | (0%) |
| 4 | 0 | (0%) |
| 5 | 0 | (0%) |
| 6 | 0 | (0%) |
| 7 | 0 | (0%) |
| 8 | 4 | (50%) |
| 9 | 2 | (25%) |
| 10 | 2 | (25%) |

Please list any other feedback you have (e.g. what features do you feel are missing, what was especially good/helpful, what was particularly bad/difficult, did anything not work etc.)

6 responses

Add more streetwear brands

More streetwear brands, logout function to work, notification and reminder system when entering your email to your email

Notifications would be helpful if I ever forget the time that a shoe is releasing, personalised system as well so only products I really want show up on the home page

Links on products were useful, product rating in stars were good information, release dates were useful to use and carousel feature was nice aesthetically.

More streetwear products, everything else was brilliant

Links to product websites feature was brilliant and easy to use. Home page was stacked and showed all useful information without me having to waste time navigating through the site. Efficient

4.3 – Client Sign-off

I, as the client, am happy with the development and final testing.

Name: Richard Chinedu

Signed: _____

Date: 04/04/2022

5 – Evaluation

5.1 –Success Criteria

5.1.1 – Evaluation of Success Criteria

| Criteria No. | Success Criteria | Met, not met or partially met? |
|--------------|--|--------------------------------|
| SC1 | User should be able to create an account and should be prompted to fill in all the required data in a valid format | Met |
| SC2 | When the user creates an account, the data should be stored in the correct database | Met |
| SC3 | User should be able to successfully login with their correct email and password and allow multiple attempts to login | Met |
| SC4 | User should be allowed to enter a contact message to the company securely | Met |
| SC5 | The app should save the inputted data to a database of messages. The message should be saved with the users email and name to show which users message reading it is | Met |
| SC6 | The app should have many visual icons in the menu | Met |
| SC7 | The app home page should be able to display all most important information and products on that day | Met |

| | | |
|------|---|--|
| SC8 | The app should have a top navigation bar and every time the user presses on one of the features: Contact, Reminders, Footwear categories, streetwear Categories and Home to allow easy navigation through app | Met |
| SC9 | Links to page and displays page when the user clicks the see more button on a product | Met |
| SC10 | User should be able to receive reminders and notifications after favouriting a desired item, either on the home page or to their email (if possible) | Not met |
| SC11 | User should be able to favourite items if they are logged in, and favourite multiple items if they want to and all of these items should be added to the reminders list | Partially met – users were able to favourite products |
| SC12 | User should be able to search the product database and most relevant products matching the search appear | Not met/Partially met |
| SC13 | If there is no search result for a product item, the user should be prompted to try the search again and should be informed that there are no matching results | Not met |
| SC14 | The user should be able to access the settings menu and change any of the features they wish to regard them personal details and these will be edited in the database | Partially met- no settings menu was implemented but a forgot password feature was implemented to change the users' personal details, which also changed them in the database |
| SC15 | The user should be able to log out of the app with all data remaining saved in the database | Not met |

| | | |
|------|--|-----|
| SC16 | User should be able to access links and information to each product so they can purchase and see release information | Met |
|------|--|-----|

All of evidence for the meeting of these success criteria are in my test table and highlighted there. The corresponding test that is similar to the test number in the table are the evidence of these success criteria being met.

5.1.2 – Unmet and Partially Met Criteria

During the development, due to the time constraints and issues with implementation, some of the success criteria were only partially met, or were not met at all:

- SC10 and SC11(not met/partially met) – Both of these criteria were to do with the favouriting of specific product items and then these items being added to a reminders list and then receiving notifications about these items. The problem came with trying to implement the favourites/reminders list system which as shown in the testing I tried to do but just did not work. Also the code for trying to favourite an item and this only working if the user was logged in failed to implement too. This could be due to a lack of experience coding in the languages of HTML, CSS, and JavaScript. As these personal reminders list was failing to implement, it then prohibited me from trying to make a notification system. The user login was working but there was no way of linking these favoured items to the user account, there was no way of sending these personalised notifications out as the favoured items weren't saved to the users specific database and therefore there was no way of knowing what notifications to send to who, so the notification system was of no use and I failed to implement this.
- If I had more time to develop this and make this work I would've spent more time learning in-depth more about all of these languages and learning about my database software. By doing this I would be able to link favourite items to specific users in a database and the favourites list that wasn't working, I would be able to develop it so it could work and this would allow me to build my automatic notification system.
- SC12 and SC13 (not met/partially met) – These criteria were to do with the user using the search engine function on the website to search for the desired products and then it is displayed to the user if products in the database match or are similar to that search. The search engine and icon were successfully implemented however when typing in a product and then pressing enter the website loaded back into itself and nothing happened. Reasons for this failing to work was that I did not have enough time and experience coding in these languages so even after researching and trying to learn, it was very difficult to implement and make work. In the end I decided that it was so necessary because as long as the user can navigate through

the app easily and quickly from page to page to find these products then the search function didn't hold much significance. Also because there were not that many products on the website that there was no need to create a whole database for them and the user would probably not need to search for what they need anyways because everything is easy to find as only the most popular items are on this website and all navbar features are labelled so the user would know where the items that they desire are by using these other features that I implemented. Also a lot of items desired items (based on popularity shown by rating stars) are already on the home page so the user may not need to search. Because of this failed search function, SC13 could not be met at all because if nothing happens when the user searches and if no products result from the search then there would be no chance of no matching results happening and then no display of the prompt to try to search again.

- If I had more time to perfect and further develop this website I would implement this system because as much as I believe it was not necessary it would still improve the overall quality of the website and make it more efficient for the user. A way to develop it was by putting keywords and letters in the search function when coding so when a user searches a product if any of these keywords/letters appear in the input then these products will display.
- SC14 (partially met) – This criterion was to do with making the users' experience on the website easier and making a password reset feature for the user when the user forgets his password or wants to reset his password for any other reason. If this feature was not implemented the user would have to create a new account every time they want to log in. This feature was implemented successfully on the log in page as a link under the user log in input fields as this is where it makes more sense to be as the log in page is where the user would realise they have forgotten his password but it would be better to implement this in the actual website on the navigation bar as a settings visual icon and when clicking on it, it would display options to reset user information. This is the criteria I set for myself before and I did not do it like this but the fundamental part of making a reset password feature was implemented successfully.
- In the future if I had more time, I could definitely implement this, but I did not have it as a priority as I had already made the actual forget password function. Due to time constraints and limitations, I only completed features that were of priority. A way to develop this is to add it to the navbar in my HTML code and use a visual icon picture and implement this as a dropdown button. Upon clicking this button, the content will display for the settings and one of the options would be to reset their password and when the user clicks this they would be taken to the page to reset their password
- SC15 (not met) – Criteria is to do with the logout function on my website which failed to implement properly and the button not actually doing anything when clicked on. I was meant to be able to log out after logging in and after logging out all data was meant to be saved from that account. This did not work as data meant to be saved was favourites list and this didn't implement fully as discussed before so

nothing could be saved while logging out. Also when logging out it should've taken me straight back to the log in page but this did not work either

- In the future to make this work I could implement my favourites list and favourites database connected to that users ID personalised to each user. Then I would make this save after logging out and then make the button work by implementing a link to the login page after pressing this button.

Requirements

Evaluation of Requirements

These are the evaluation of the design requirements, user requirements and more that were set in my analysis

| • Requirement no. | Requirement | Met or not met? |
|-------------------|---|--|
| R1 | The game should be suited to any type of demographic, such as resellers and casual buyers | Met – casual buyers can use it and access links to buy the shoes they want and resellers can check resell prices |
| R2 | The website should be easy to navigate around | Met |
| R3 | The app should have multiple useful features, such as favouriting desired items | Met – however not all of them worked |
| R4 | The website should be accessible on all platforms. | Not tested |
| R5 | The first page of the website should give you essential information immediately on the hottest drop that week, with pictures and more for it to look nice | Met |
| R6 | The notifications given on chosen items should be very quick so it gives more of a chance of coping the item | Not met |
| R7 | Users should be able to | Met |

| | | |
|--|--|--|
| | contact us when they run into an issue | |
|--|--|--|

| Requirement no. | Requirement | Met or not met? |
|-----------------|---|--------------------------|
| R8 | Have a specific theme throughout the application | Met |
| R9 | All users should have their own profile to keep track of their progress data | Not met |
| R10 | Internet connection (Software) | Partially met |
| R11 | A platform for a HTML, CSS and Java Editor. FTP client to upload files to web server (Software) | Met – Visual Studio Used |
| R12 | Computer/Laptop (Hardware) | Met, used HP hardware |

| | | |
|-----|-------------------------|---|
| R13 | Touch Screen (Hardware) | Met, these functions were implemented in every page |
| R14 | 2 GB RAM (Hardware) | Met, the required RAM was obtained and used |

| Requirement no. | Requirement | Met or not met? |
|-----------------|--|-----------------|
| R15 | User will be able to login with a username and password | Met |
| R16 | User can search for shoes and clothing items in a product database | Not met |
| R17 | User can create account to receive restocks and reminders | Partially Met |
| R18 | To be able to develop the game I will need to have Visual Studio downloaded onto my computer to be able to code Using HTML, Java and CSS | Met |
| R19 | Program should have a menu to access the login page, reminders, features and homepage. | Met |
| R20 | Use drop down menu with categories under a feature. Display all the shoes and clothing within the sub-category | Met |

| | | |
|-----|---|--|
| R21 | Users are able to log out and log in easily | Partially met – log in works not log out |
| R22 | There should be multiple links for each product (depending on how many sites the product is sold on). Users can click a link successfully to go and buy the shoe that they want | Met |
| R23 | User can reset password. | Met |

| Requirement no. | Requirement | Met or not met? |
|-----------------|--|--|
| R24 | When the website loads up, the first screen should be the home page with a menu of icons with links to each of the features in the site | Met |
| R25 | User should be able to mark sneakers as “favourite” and these foods will appear in a favourites list and will be the shoes that the user gets restock/release notifications about. | Partially met – favourite list failed to implement |
| R26 | User should be able to tap on a product and a pop up of the information about that product will appear | Met |
| R27 | The program should be easy to use and navigate for someone of any age | Met – with the use of visual icons. |

The end user is satisfied that most of his requirements were met successfully.

5.2 –Validation

5.2.1 – Evaluation of validation

| Validation No. | Feature | Validation required | Met, partially met or not met? |
|-----------------------|---------------------|--|---------------------------------------|
| V1 | Login | User should only be allowed to log in if all fields are entered and all required standards are met | Met |
| V2 | Login | User should only be allowed to login with a correct email address and password | Met |
| V3 | Login | User should be able to retry if login failed, the app should not crash | Met |
| V4 | Contact Message | User should only be able to send message when all required fields have been filled | Met |
| V5 | Contact Message | User should meet required field format | Met |
| V6 | Display products | The app should ensure that the user can view the information of one product at a time | Met |
| V7 | Searching for items | App should not crash if no results are found when searching for an item | Not met |
| V8 | Creating account | User must enter info in the required fields and must be in the correct form | Met |

| | | | |
|-----|------------------|---|-----|
| V9 | Creating account | User cannot use an email that has already been used | Met |
| V10 | Creating account | User's password must be at least 6 characters. | Met |

Test evidence for the success of these validations can be found throughout my code, some pages referenced to from the test plan so they are easier to find.

5.2.2 – Unmet or Partially Met Validation

During the development, due to the time constraints and issues with implementation, some of the validation criteria were only partially met, or were not met at all:

- V7 (not met)
- This piece of validation was to ensure that the app should not crash if no results are found when searching for an item. This criterion was not met for the simple reason that the search function of the website was not successfully implemented.
- If the website was to be developed further and the search feature was implemented, it would be essential that this piece of validation being implemented also as the search function cannot be seen as fully successful until this validation is in place.

Analysis and Evaluation of User Review Questionnaire

To assess how successful my website was I had around 8 different users fill in the questionnaire for my review of the website. All 8 of the users asked said that they found it easy to both create and account and login to the account. This is successful since it is important that this is straight forward as to not waste time in the busy of schedule of people using my app, for example, a reseller. All users questioned also said that the app was easy to navigate which again will result in less time wasted and a better overall experience of using the app as a difficult to navigate app will often result in it not making the best use of the features available. As well as this, none of the 8 users found the website to crash or close unexpectedly at any point which makes the users experience much easier, but form a developer point of view, shows that the app is robust and sufficiently validated.

All 8 users said that the websites home screen had useful features and information and instantly displayed these features on the website upon opening it. This shows that my initial goal of making the home screen a page where the user would feel like there is being no time wasted has been successful. To have all necessary products, features, and information all on the home page means the user wouldn't have to navigate through the website and can access to things that he needs from there.

Of the 8 people mentioned, 50% of them said the favouriting items feature was not useful. This was expected because as discussed in other parts of my evaluation the feature failed to implement properly. The usefulness of this feature would've been to have a favourites list for every user which saved in their account but as I was unable to do this, it lowers the usefulness of the favouriting feature. As for the other 50%, they both claimed it was useful but half said it was easy to use and half said it was difficult to use. This suggests that even with the feature not implemented fully the basic part that I did implement of having the heart icon there and upon clicking it, the number of favourited items was shown to the user was appreciated by a few. The people that said it was difficult could be referencing the favourites list not actually showing when clicked which is understandable. This range of answers shows how well my success criteria for this section were meant and it further justifies why I deemed it partially – clearly some people still appreciated the feature.

All the users claimed that they would at least use this app at least once a month and none said they would not use it. The majority, 50% said they would use it a few times a week and 37.5% even stated that they'd use it everyday which shows that while the websites could be improved, there is some usefulness to it that makes it worth using consistently. I can assume that the casual dressers and sneaker wearers will only use this during the times when they want to complete an outfit and buy a pair of clothing but with a reseller, the app would be used way more consistently because they have to check for shoe every day and all the time. Out of 8, all the users rated the attractiveness of the app either a 4 or a 5 which proves to me that the app's design is attractive and appealing to users which would likely make them more likely to use it.

All of the users rated the ease of use of the app a 5. This implies that the app is relatively easy to use and no improvements need to be made in regard to the smoothness of the user experience. The usefulness of the app was rated either a 4 or a 5 by all the users showing me that even without the implementation of the notification system into my website, the website is still very useful for people. This is probably due to the usefulness of the links and the release times. In the future to make all users rate it a 5 I can implement this notification system in addition to everything else.

All 8 people stated that the product release times were useful to them and accurate, highlighting the in-depth research I had to do when obtaining the release dates for these products and then helping the user with this date and time as these dates are hard to find and not explicitly announced by the company all the time. This also showed the users used this as when the product was released at the time stated by me, the product was released on the website. The use of this feature by the user further enforces the usefulness of this website. On the other hand, all 8 people stated that they were not able to log out successfully, a problem that has been stated in my code and addressed in my success criteria. This suggests the implementation of this feature was not successful at all. This was the only clear answer where no was the majority which suggests overall my website was a success but evidently there is still things that were lacking and need further development.

I then asked people if they made use of certain additional features that I added to the website, like the contact section. Out of all 8 people that I asked, 75% said they used it and it was helpful. This suggests that even though it was not a necessary feature to add to the purpose of my website, its implementation of it was very good and very successful. It also shows that a lot of people do leave feedback to website companies, positive or negative. 25% also said they did not use it at all which is expected as it is not the main purpose of the website and just a feature that if you want to use it, is there for use. I then asked if the changing of personal information was easy if they did it or not. 75% did not use the feature which could suggest that it wasn't really needed but forgetting your

password is quite common. 25% did use it and said it was easy to use, further highlighting another successful implementation of code and showing efficiency and ease of use.

I then asked if people thought there was a good variety of products and if they were all the most popular products that they wished to see. The majority said yes there was a good variety and all of their desired products were there on one site, which is good as it fulfils my original purpose of making a website that only had necessary items that are trendy and popular among young people. 25% said some more popular products could've been added but there was still a good variety but 12.5% said there was no variety, but they still identified the products as popular products. This highlights the wants and needs of different types of people which is to be expected as the website can't satisfy everyone and as with any app, different users will expect and want different things out of the app and may find things harder or easier than others, so a spread of answers is to be expected.

I then asked if the website links to pages you can buy were useful, and this was a question where users could write their own answers if they wanted to. All 8 users answered and every single one of them said yes, they were useful which shows the success of this feature and shows people of any demographic within fashion can make use of this website when trying to buy popular items. A lot of people said it was efficient and saved them time which highlights efficiency and ease of use, others said that it even gave them new information on the sites/apps where these products were actually released which suggests a lot of people don't even know basic information on these types of products and this highlights that my website has successfully solved that problem. Finally, resellers said that it was helpful to their resell business and this shows that it appeals to different demographics.

I then asked if people had any feedback to give to make the website better or to just give positive feedback. Out of 8, 6 people responded and many of them said that what they wanted was to have more streetwear items and brands on the website. This suggests my page was too one-sided with the idea of the sneakers and to expand the website and make it better and more unique and successful, the addition of my streetwear would do this. A lot of them also said about a notification system being implemented to remind them when they forgot about a release which shows that my system could be improved a lot.

The overall ratings out of 10 varied between 8 and 10. I think this is quite successful, but to improve this score, the logout and favouriting function of the website needs to be improved as these were the main issues highlighted by the questionnaire.

The overall result of the questionnaire was positive and gave the impression that the website was liked and very successful and worked well but required a more in-depth development which could have possibly been achieved with more time, more knowledge and better planning.

5.5 – Maintenance

Since the code could benefit from further development, it is important to assess the maintenance of the app and how easy it will be for it to be maintained. An example of this is using sensibly named variable, function, class and method names. I feel as though in my code, features have been named appropriately in order to make the code easy to follow or return

to, or if someone else was to view the code, they would be able to work out what different variable will store and what different methods do. For example, the variables used when implementing my log in/register page to be saved to the database. The variables used for the users email and password were Email and password. The function name for creating a new account and then saving the user's information to the database is called `createUserWithEmailAndPassword` – the name says what the method does. I feel as though I have done this consistently through my website and I found no difficulty in remembering what features did when returning to them after some time.

Another feature of making the website easier to maintain is including comments throughout the code. This is very important when coming back to the code after some time or having someone else be able to understand what different sections of code do. My comments are shown in green in my code screenshots. Along with the sensibly named variables and methods, I think that this makes my code very easy to follow, maintain and develop further.

I also made it easier to maintain for me by creating separate files for different languages and images as shown in one of my first screenshots. I made a file for images, for CSS files, for JavaScript files and finally for HTML files. This maintained my website easily when creating it as all similar files were stored and organised in the same place so it was easy for me to find them when I wanted to edit them. When creating a new file I would store it in the folder depending on what type of file it was.

One of the problems I have evaluated with the maintenance of my website comes with repeated code. While I did use methods and variables that were called multiple times within each activity, many of these methods were then repeated very similarly in different activities, for example for my all my product pages all code needed to be repeated and it was the same exact format. The problem with this is that if the code needs to be edited for whatever reason, it will need to be changed every time it is featured rather than just one time in the original method.

If my website was to be implemented as an app, the app may require extra maintenance when a newer version of Android/iOS is released, and the OS is updated. It is important that when a user updates their phone, the app does not stop working because of new feature of the phone's update. To prevent this, it is important to ensure the compatibility of the app and ensure that it works not only on new versions of android when they are release, but still works on the old versions to ensure everyone can make full use of their app.

5.6 – Limitations

My laptop hardware was a big limitation for me when creating my website as the laptop that I used was susceptible to lagging and a lack of storage. My laptop would initially not

allow me to download certain apps because of the software requirements. Because of this I had to obtain another laptop from somewhere else, but it took me some time to obtain and because of this the issue of time constraints arose and I wasn't able to fully implement things that I wanted to.

A further limitation is the fact that I have not used HTML, JavaScript, and CSS in the past a significant amount. Because of this there was some pieces of code that could've been implemented easier than they were and there were features which I failed to implement that I could've implemented having had experience with the programming languages. However, I did learn more on the go whilst doing my project and it was a good learning experience for me.

Another limitation is the fact that the app is only in English. While this would not be very easy to overcome, and would include collaboration with many different people fluent in various languages, it would be a very nice feature to add in further development in order to be more inclusive and allow greater accessibility of the app. In apps, it greatly increases the sales and downloads if the app can offer different language options as it widens the available audience, but people also would appreciate the inclusivity.

The final limitation of my final product would be the failure to implement a notification system and it could lower the use of the website significantly because people forget when it comes to product releases. If people forget they are not likely to use this app for the purpose of using its links to the websites where these items are released and this will lower usability. However, they would still use it for the information on the release dates of these products, so the website would still have some form of usability

5.7 – Improvements

If the app was developed further, there are some improvements that I would make in order to make the app more usable and to improve the functionality and usefulness of the app. The first improvement I would work on is the addition of the notification system which would improve usability as described in my limitations section. However, to implement this notification system as I planned, I would need to improve the favouriting items feature that

I tried to make work in my development. When favouriting items, the item would be added to the favourites list and when clicked on the favourites list should appear and all items that were favourited should display on this list. This should be unique to what each user has favourited, and the data should be saved even when the user logs out of the website. These items would then be the items that the user would receive release reminders/notifications about, so the user should only favourite items they want to receive notifications about. This would improve usefulness greatly as more people will use this website if they are essentially reminded to use this notification system and they will be satisfied knowing they have their own personal list of items that they receive information on, it will give them a sense of security knowing they are highly unlikely to miss any releases with this system in place. To do this I would not have a lot more time and also more experience, so I am secure in my knowledge of these languages.

I would also like to improve the search function of my website, so it actually outputs the desired product that the user entered. Whilst I deemed this as not so necessary, it would be a useful feature in the site which will improve efficiency and functionality. To improve this even further I would add a filter on my search so only the items corresponding to the filter that the user has selected would be displayed. To do this I would need just a bit more knowledge and a large product database.

Another big feature that I want to improve is the logout functionality as this would be for users who after they have logged in, desire to log out for any reason (maybe they do not want to receive notifications anymore). This was a big part that was missed in the creation of my login system and the improvement of this would complete this.

I could also improve my system by adding more visual icons to the system to improve the aesthetic of the system and make it appealing to all people of different types. This would help people that need more visual help and need these icons for any type of reason for example people with dyslexia.

5.8 – Conclusion

Throughout the process of developing the website, it has been interesting to have an insight on the development process of a project and the skills required to successfully document the process in a sensible way. It was also beneficial to understand the importance of time management and prioritising the criteria of the project to be able to produce a successful product that may not meet all the criteria but meets the main requirements. It is a valuable skill to be able to assess a situation based on the time and difficulty and attempt to stick to a schedule so that one can turn in a fully functional product. It was also very beneficial to work with a client and be able to develop the project with a specific proposal in mind. I found it useful to discuss with the client the various stages of the project to get feedback to

check that they are happy with the way the project is being developed. I found it very beneficial for my evaluation to be able to resent the website to real users to get an idea of how they found the site and especially whether they would use the site. It is very difficult as the developer to see what a user will see, so having people evaluate the site for you is incredibly helpful for the final evaluation of the website.

Overall, I think that my website was mostly successful and meets the client's original proposal for a website and also solves the initial problem that I identified at the very start of this project – to create an app/website for people who are not informed about popular sneaker/streetwear releases and needed an efficient website to inform them about it and also a place for people to have quick and efficient ways of accessing websites where these items are sold (through my links). While the website does not do everything that was proposed in the design section and analysis section (such as reminding users when products are released through notifications) and does not meet all the success criteria fully, it does meet the main criteria and runs successfully with no instances of it crashing or closing unexpectedly. The client is happy with the website app I have presented; however, they do agree with the improvements suggested and they agree that with more time, many more useful features to improve the user experience could be added. While the final product does differ slightly from the product suggested in the analysis and design section of the website, it fulfils the client's proposal and I, therefore, see it as successful, as does my client.

5.9 – Client Sign-off

I, as the client, am happy with the evaluation.

Name: Richard Chinedu

A handwritten signature in black ink, appearing to read "R Chinedu".

Signed: _____

Date: 15/04/2022