



An-Najah NATIONAL UNIVERSITY  
FACULTY OF COMPUTER ENGINEERING

---

# Housewares

---

A Web/Mobile Application to Management Housewares Stores

**Authors**

Essa Abbadi  
Ahmad Qorom

**Supervisor**

Eng. Haya Samaana

January 15, 2023

# Acknowledgements

*“First and foremost, we’d want to express our gratitude to Eng. Haya Samaana, who assisted us in completing these tasks. She gave us great guidance and supported us during difficult times. Her enthusiasm and assistance made a significant contribution to the project’s success. Furthermore, we would like to express our gratitude to all of the professors who assisted us by offering advice. We’d also want to express our gratitude to our families and friends for their support. We would not have been able to finish this project without their help. Last but not least, we’d want to express our gratitude to everyone who assisted and encouraged us to work on this project.”*

# Disclaimer

This report was written by students (Essa Abbadi and Ahmad Qorom) at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. Therefore, it has not been amended or modified as a consequence of assessment, with the exception of editorial adjustments, and it may include grammatical and content problems. The opinions stated, as well as any conclusions or suggestions, are purely those of the students. An-Najah National University accepts no responsibility or liability for the effects of this report being used for something other than what it was intended for.

# Contents

Housewares

Essa Abbadi Ahmad Qorom ..... 1  
Eng. Haya Samaana ..... 1  
January 15, 2023 ..... 1

**List of Figures ..... 3**

**Chapter 1 Abstract ..... 4**

**Chapter 2 ..... 5**

**Introduction ..... 5**

**2.1 Background ..... 5**

**2.2 Problem Statement ..... 5**

**2.3 Significance ..... 5**

**2.4 Objectives and Scope ..... 6**

**Chapter 3 ..... 7**

**Constraints and Earlier Coursework ..... 7**

**3.1 Constraints Limitations ..... 7**

**3.2 Earlier Coursework ..... 7**

**Chapter 4 Literature Review ..... 9**

**4.1 One Store apps ..... 9**

**4.2 Other fields apps ..... 9**

**4.3 How is Housewares Better? ..... 9**

**Chapter 5 Methodology ..... 10**

**5.1 Architecture & Technologies Utilized ..... 10**

5.1.1 ReactJS .....	10
5.1.2 Material UI .....	10
5.1.3 ExpressJS .....	11
5.1.4 Mongoose .....	11
5.1.5 MongoDB Atlas .....	11
5.1.6 Firebase .....	12
5.1.7 Others .....	12
<b>5.2 Features &amp; Implementation .....</b>	<b>12</b>
5.2.1 Home Page .....	12
5.2.2 User Account .....	14
5.2.3 Store Page .....	17
5.2.4 Categories pages (eg. Kitchen page) .....	19
5.3.1 Payment method and Cart page .....	22
5.4.1 Admin Page .....	27
5.4.1.1 Main section(الرئيسية) .....	27
5.4.1.2 Statistics about the store .....	28
5.4.1.3 Add product .....	29
5.4.1.4 List of Products .....	30
5.5 Chat and Notification .....	33
5.6 Wish List Page .....	35
<b>Chapter 6 .....</b>	<b>37</b>
<b>Results &amp; Discussion .....</b>	<b>37</b>
<b>6.1 Final Application .....</b>	<b>37</b>
<b>6.2 Project Outcomes .....</b>	<b>37</b>
<b>Chapter 7 .....</b>	<b>38</b>
<b>Conclusion &amp; Future Work .....</b>	<b>38</b>
<b>7.1 Summary .....</b>	<b>38</b>
<b>7.2 Future Work .....</b>	<b>38</b>
<b>Bibliography .....</b>	<b>39</b>

# List of Figures

Figure 1 : Home Page .....	13
Figure 2 : Stores Ratings Chart .....	13
Figure 3 : Best Products in Home Page .....	14
Figure 4 : Login Form .....	14
Figure 5 : Sign Up Form .....	15
Figure 6 : Forget-password Interface .....	15
Figure 7 : Confirmation Code .....	16
Figure 8 : Login Front-end Validation Example .....	16
Figure 9 : Sign Up Front-end Validation Example .....	17
Figure 10 : Store Page .....	17
Figure 11 : Location of the Store .....	18
Figure 12 : Last Week Ratings .....	18
Figure 13 : Last Month Ratings .....	19
Figure 14 : Rate the Store .....	19
Figure 15 : Filters in Categories Page .....	19
Figure 16 : Kitchen Page Products .....	20
Figure 17 : Filter by Color .....	20
Figure 18 : Arrange By filter recommends .....	21
Figure 19 : Filter by size .....	21
Figure 20 : Filter by name of the Store .....	21
Figure 21 : Product Description Page .....	22
Figure 22 : Similar Products and Ratings Section .....	22
Figure 23 : Cart Page .....	23
Figure 24 : Address Form .....	23
Figure 25 : Choosing method of payment .....	24
Figure 26 : Method of Payment Page .....	24
Figure 27 : Review of Order Page .....	25
Figure 28 : Order Done .....	25
Figure 29 : Two Purchases Example (1) .....	26
Figure 30 : Two Purchases Example (2) .....	26
Figure 31 : Two Purchases Example (3) .....	27
Figure 32 : Admin Main Section .....	27
Figure 33 : List of Store Products .....	28
Figure 34 : Delete Button .....	28
Figure 35 : Statistics section .....	29
Figure 36 : Form to add Product .....	29
Figure 37 : Order Products Descending by name .....	30
Figure 38 : filter Products by category and select them .....	30
Figure 39 : edit Product Form .....	31
Figure 40 : add Variation Form .....	32
Figure 41 : Edit Variation Form .....	32
Figure 42 : Customer Send a Message to Store .....	33
Figure 43 : notification on messages .....	34
Figure 44 : Store replied to the Customer .....	34
Figure 45 : Notification removed .....	35
Figure 46 : Notifications from the Admin .....	35
Figure 47 : Wish List Page .....	35

# Chapter 1

## Abstract

We all know the importance of competition specially in e-commerce markets as it makes each market show the best of its products, prices, and services here comes our app which makes a competitive environment between housewares Stores. The app allows customers to see the best stores, products and discounts to help them choose the best quality and price while also help stores to add and edit products easily with different colors and sizes and show them some statistics about their performance and rating to keep them in the competitive seen. It also provides chatting between customers and stores with notification system for chatting between them. All of this is contained within one app.

Housewares is created using the JavaScript-based ReactJS framework for the web and React Native for mobile devices. Additionally, it has a NodeJS application programming interface (API) that supports the database management and core application functionality. It employs two different types of databases: a NoSQL database (MongoDB) is used to store the application's core data (products, accounts, ratings, and payments), and Firebase Firestore is used to handle real-time functionality like chat messages and notifications.

# Chapter 2

## Introduction

### 2.1 Background

Housewares Stores can show their best in competitive environment and they can learn from their mistakes like pricing a normal product a high price will defiantly lower their rating. Moreover, Customers can see the best Stores and choose the best value for price products. Also give their opinion about the product/store quality or service.

### 2.2 Problem Statement

Every Housewares Store faces 2 main problems:

- **Customers Reach:** Housewares Stores are known only to their clients that are close to them or hear about them from a friend. Because they don't have a web or mobile store. So involving in our app will make their store known to all app clients whether they are close or not.
- **Reputation:** If Store have good products and prices in general. Its rating will be higher of course and it will be seen to all customers and vice versa.

On the other hand Customers gets 2 main advantages:

- **Having Products in one place:** Customers can easily see and compare all products from different Housewares Stores by going to the related Section and Search or Filter the products.
- **Choosing the best product:** Customers can see the ratings on all Stores and Products. So they can easily choose the best product that suits their needs.

### 2.3 Significance

For Customers choosing the best product became more and more importance specially in

these days where the Salaries are lowered in a lot of majors. Moreover Housewares Stores need an competitive environment to show their products and services to wide area of people.

## 2.4 Objectives and Scope

This app isn't a e-commerce app for a market; it's an Management system that doesn't hide the rating of Stores to be seen as equal. The goal of this program is to build competitive environment that provide values to both Customers and Stores. Using Different Features:

1. Chat Messages Feature: This feature helps provide a unified way for Stores and Customers to communicate with each other by exchanging messages, and supports one-to-one messages.
2. Notification Feature: This feature is two parts. First notifications for messages, if you have unread messages it will give you a notification about it. Second push notifications using firebase. which are notificatios from admin to stores about discount days, holidays and so.
3. Wishlist Feature: This allows the Customers to Store products in Wishlist for buying later or when discounts comes to them.
4. Different Variations of Product Feature: This feature allows the Stores to Store different Sizes and Colors for the same Product.
5. Cart and Checkout Feature: This feature allows Customers to Store all Products they need to buy in cart and then buy them in checkout later if they need.
6. Stores Pages: each store has its own page which tells the customers more about the store and allow them to rate it.
7. Admin Page: using this page the Store can see different Statistics about their stores and see all of its products .Therefore can add, edit products with different variations or make a discount on multiple products at the same time like on holidays.

# Chapter 3

## Constraints and Earlier Coursework

### 3.1 Constraints Limitations

#### 3.1.1 Lack of suitable information about housewares Stores

Housewares Stores provide us with little information about their Stores. They prefer to keep the majority of information secret. Which create a struggle for us but we have over come it eventually.

#### 3.1.2 Time Limit.

It took a long time to construct the entire program, which included learning technologies, searching for a topic, creating the user interface, and implementing it on both sides. A time wasn't enough to perform search of best libraries to handle some features. The majority of the features agreed upon with the project's teacher were completed.

### 3.2 Earlier Coursework

- Object-Oriented Programming (JAVA):  
Although this project does not use Java, the synchronization concepts were initially developed there. However, NodeJS also used them, and the lessons acquired were extremely helpful in building many of the app's features.
- Database Design Management:  
This application's database was created entirely with NoSQL technology (No Structured Query Language). Along with database design principles, crucial ideas including dependencies, keys, and indexes were covered in this course.
- Web Development:  
The front-end employed HTML, CSS, and JavaScript, whereas JavaScript The back-end made use of a library named NodeJS. All of these languages were taught to me in this course. On the other hand, this course discussed (APIs) and how and why they are created.
- Software Engineering:  
The creation of this program benefited greatly from the meticulous consideration

given to each idea, including software requirements, architecture, user acceptability, and version control.

- **Critical Thinking Scientific Research:**  
To prepare for creating this application, which was done in this course, it needed a lot of research across websites. However, it was the first location where the skill of using latex and creating scientific publications was shown.

# Chapter 4

## Literature Review

Housewares aims to create a user-friendly and comfortable management experience for both the Stores and the customers. However, it is not the first application that was created to achieve the same purpose, there are some other different applications that are already live but on one Store only without Management or in different fields:

### 4.1 One Store apps

There are a lot of apps and projects that create an environment for one store to place its products and allows customers to buy from it with ease. But it laxes the competition between different stores as it has only one store.

### 4.2 Other fields apps

There are also apps that manage the stores in fields other than Housewares like food applications for example ”يمني“. Our application focus on Housewares stores to make it the best app for managing Housewares Stores.

### 4.3 How is Housewares Better?

Housewares was created keeping in mind both the advantages and disadvantages of existing applications. The user interface of Housewares was designed to be as straightforward and user-friendly as possible. Additionally, it has the majority of the crucial features that other programs provide as well as what they don't. Finally, we made every effort to resolve any problems that arose with other programs.

# Chapter 5

## Methodology

### 5.1 Architecture & Technologies Utilized

#### 5.1.1 ReactJS

React is considered a flexible free open-source front-end JavaScript library that is used to develop fast and interactive web applications[1]. The library was created by Meta (formerly Facebook), and it has a very large community as well as a collection of libraries and reusable parts.

We decided to use React for many different reasons:

1. React provides JSX syntax, which is a simple, intuitive, and easy-to-use syntax. Although it shares many similarities with HTML, the primary difference is that JavaScript code may be injected into the front-end design, giving the developer additional control over the component.
2. One of React's best features is its support for reusable components. Instead of having to design a layout from scratch each time, you can just create it once and import it everywhere it is required.
3. A huge array of completely free libraries. This is one more crucial aspect of React that enables programmers to benefit from the labors of more seasoned programmers by using their libraries. Although it may not always be beneficial, this supports the goal of completing tasks as quickly as possible.

#### 5.1.2 Material UI

Material-UI is simply a React library that allows the developers to import and use different pre-built components to create a clean user interface in React applications [2]. Since the developers no longer have to completely rewrite every component, Material UI saves a lot of time.

We decide to use Material UI for many different reasons:

1. Every component in the library offers a variety of examples showing how to use and alter it, which is one sign that the library is properly documented.

2. Numerous new components and bug fixes are regularly added to Material UI. With Material UI Lab, you may still access the new features that haven't yet been added to the core library [3].
3. Consistency in look and openness for innovation are two of Material UI's key features.

### 5.1.3 ExpressJS

Express is regarded as a straightforward and adaptable web server/application framework for NodeJS. It gives developers a wide range of functionality for both web and mobile apps and makes it simple and quick to design and create a web application. [4]

We used ExpressJS in our project for many different reasons:

1. Express makes it easy and rapid to construct NodeJS web apps.
2. Connections to databases like MongoDB and MySQL may be made quickly and easily.
3. It enables the developers to easily construct a scalable server using the REST API.

### 5.1.4 Mongoose

According to their website, Mongoose offers a simple, schema-based approach to modeling application data. Type casting, validation, query construction, business logic hooks, and many more features are all built-in. [5]

We used Mongoose for many different purposes:

1. Given that documents include distinct sets of fields with various data types, MongoDB is a denormalized NoSQL database, which makes it naturally schema-less. To ensure that documents adhere to a predefined structure with predefined data types, Mongoose specifies a schema for data models. For developers who came from SQL databases like MySQL, this makes working with data much simpler.
2. Mongoose features built-in schema definition validation, which is very time-efficient.
3. Returning updated documents or query results is a lot simpler using Mongoose.

### 5.1.5 MongoDB Atlas

MongoDB Atlas, according to their website, is an integrated suite of cloud database and data services to accelerate and simplify how you build with data. [6] Developers may build and maintain a NoSQL database in the cloud using MongoDB Atlas.

We used MongoDB Atlas for many different reasons, including:

1. The demand for a NoSQL database with excellent performance and fast response times.
2. MongoDB Atlas databases provide various layers of security and are very scalable.

### 5.1.6 Firebase

Firebase, according to their website, is a platform developed by Google for creating mobile and web applications. It was originally an independent company. [7] Developers can build applications for iOS, Android, and the web using the Firebase platform, which also offers tools for tracking analytics, reporting, and fixing app bugs as well as marketing and product testing.

We used Firebase for many different reasons, including:

1. controls all data in the database in real time. As a result, transferring data to and from the database is simple and rapid. It is therefore what we are employing to develop chat messaging.
2. You can provide a login for any social networking site with only a few lines of code.

### 5.1.7 Others

There are multiple other technologies used, including:

1. **React Router** which is used to build a single-page web application with navigation. [8]
2. **bcrypt** which is used to hash user passwords. [9]
3. **axios** which is used to do API calls from the client side. [10]

## 5.2 Features & Implementation

The application provides many features and a lot to talk about. This section will focus on the application main features and the role of each library in the implementation.

### 5.2.1 Home Page

When accessing the website for the first time as a customer, you will be able to view the main home page for the website. The main homepage provides the main categories about the products in many stores, and we provide the ability to navigate to see the categories of the home like kitchen page.

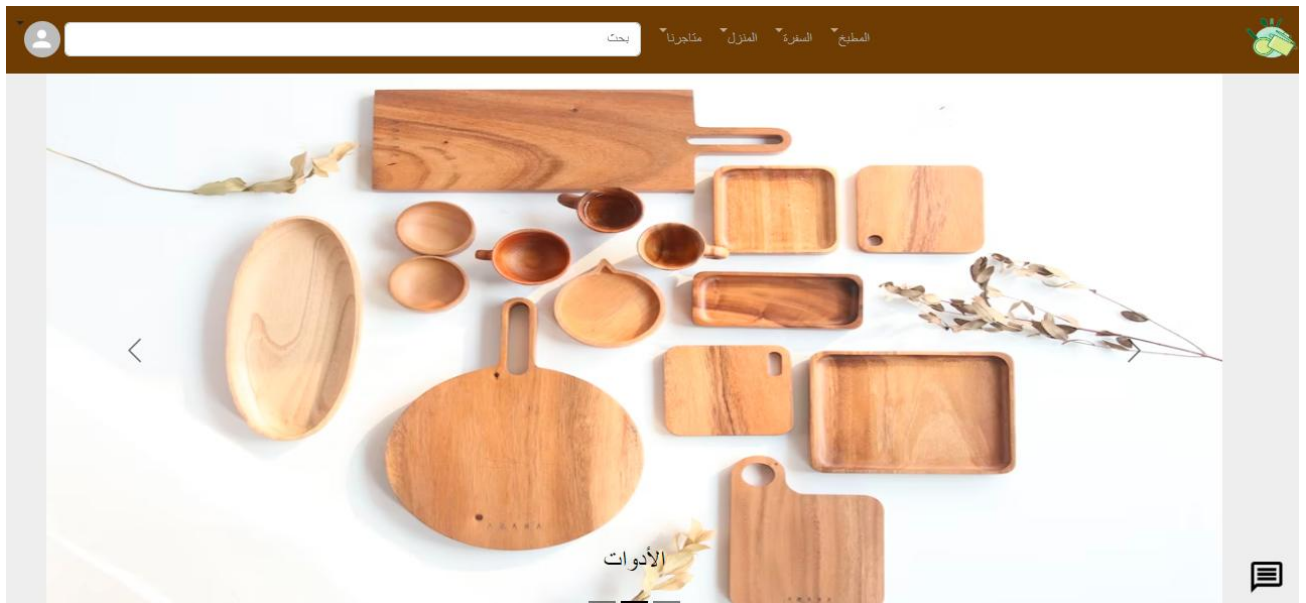


Figure 1: Home Page

Also, we show in it the stores that our site host them, and it shows statistic to make comparison between stores in ratings by evaluation.



Figure 2: Stores Ratings Chart

We also provided in Homepage some product recommendation's like (أحدث المنتجات، أفضل المنتجات، انتقل بمطبخك الى مستوى من التطور، الترتيب اساس في السفره) Which we get from the Database depending on Rating, create time and so.

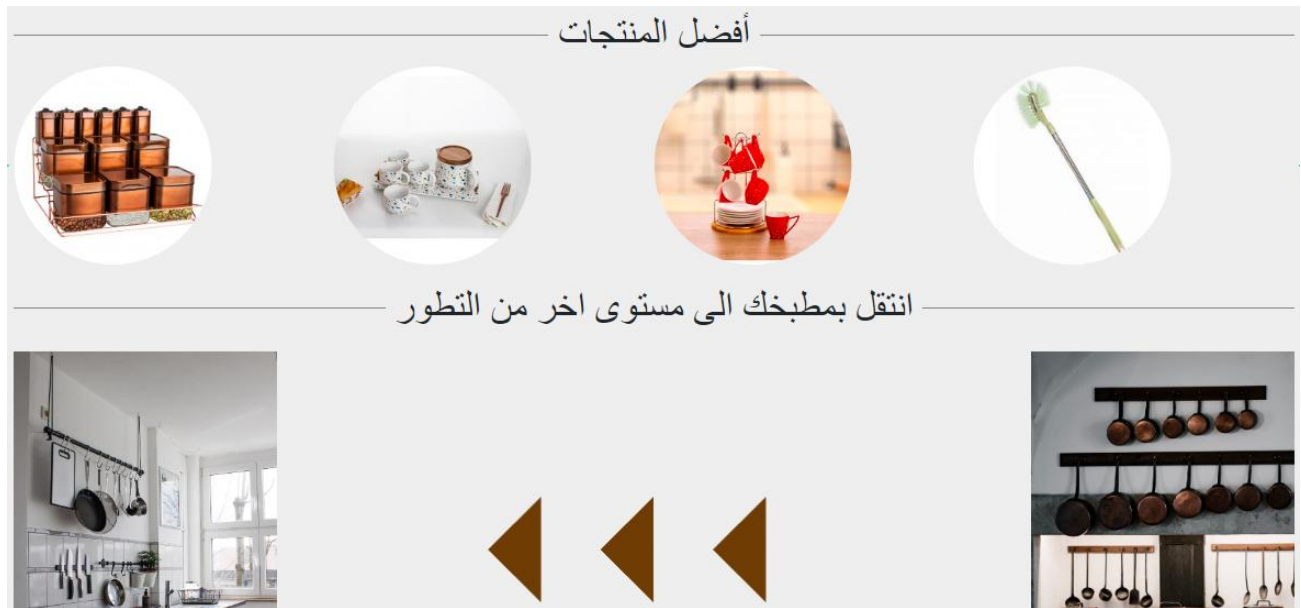


Figure 3: Best Products in Home Page

## 5.2.2 User Account

### Sign Up / Login Pages

Since our application provide to navigate between pages and give the ability for users to give their feedback about the stores and products, we provide the Sign Up/Login Pages.

On the login page, we designed two input fields to take the user email and password, to manage the user to login to his account, this is for security requirements, and we provided also another type of login, it's sign in with google, the user can sign in through his google account.

Also, the login page has two links, one link used for if the user forgets his password, he will press this link to navigate to forget password page, and the second link 'هل تمتلك حساب؟ انشأ حسابك', this for create new account.

Figure 4: Login Form

we make the password in database **hashed** by using **bcrypt** library like this:

```
password: "$2a$05$0J/GFhkU0YiTEy6eSxgNEOnIQEvqZ3Mf3Wwxu7DUbD6NN8NSaBTG"
```

On the sign-up page, there are 4 input fields that were designed with all of the stylistic requirements in mind. We also included one link at the bottom, to take the user to the Login page if it is already registered in our website.



The sign-up form is titled "إنشاء حساب" (Create Account) and features a lock icon at the top. It contains four input fields: "اسم المستخدم" (Username), "البريد الإلكتروني" (Email), "كلمة السر" (Password), and "كرر كلمة السر" (Repeat Password). A large orange button labeled "إنشاء حساب" (Create Account) is positioned below the fields. At the bottom right, there is a link that says "تملك حساب؟ سجل الدخول" (Have an account? Log in).

Figure 5: Sign Up Form

We also provided forget-password feature, if registered user forgets his password, he can make new password by using forget password interface, the user will write his email in input field, then he will receive a code to his email to make confirmations.



The forget-password interface is titled "نسيت كلمة السر!!" (Forgot Password!!) and features a lock icon at the top. It contains one input field labeled "البريد الإلكتروني" (Email). A large orange button labeled "إعادة تعيين كلمة السر" (Reset Password) is positioned below the field.

Figure 6: Forget-password Interface



node-123456789-node@outlook.com  
to me ▾

# Your Code number is : 643691

Figure 7: Confirmation Code

These two pages also feature two types of verification. The first is front-end validation, which can indicate incorrect user input and block the input button. This is handy for providing rapid feedback to the user as they enter data into the form. Second, back-end verification to check that this isn't a fraudulent attempt and that the user of the original email is attempting to access an account.

Let's talk a little bit about the importance of front-end validation, it is essential to validate that fields entered as required, else the user cannot click to enter button, it is also used to make sure that the password field match confirm field.

Figure 8: Login Front-end Validation Example



  
**إنشاء حساب**

اسم المستخدم  
مطلوب

البريد الإلكتروني  
مطلوب

كلمة السر  
مطلوب

كرر كلمة السر

إنشاء حساب

تمتلك حساب؟ سجل الدخول

Figure 9: Sign Up Front-end Validation Example

### 5.2.3 Store Page

After the user is logged in and go to one of the Stores Pages like: “الروافد”, he can see more information about the store and its products. Also, he can see the rating of this site and rate it himself.



Figure 10: Store Page

Moreover, he can see other information about the store like the location in map for this Store.

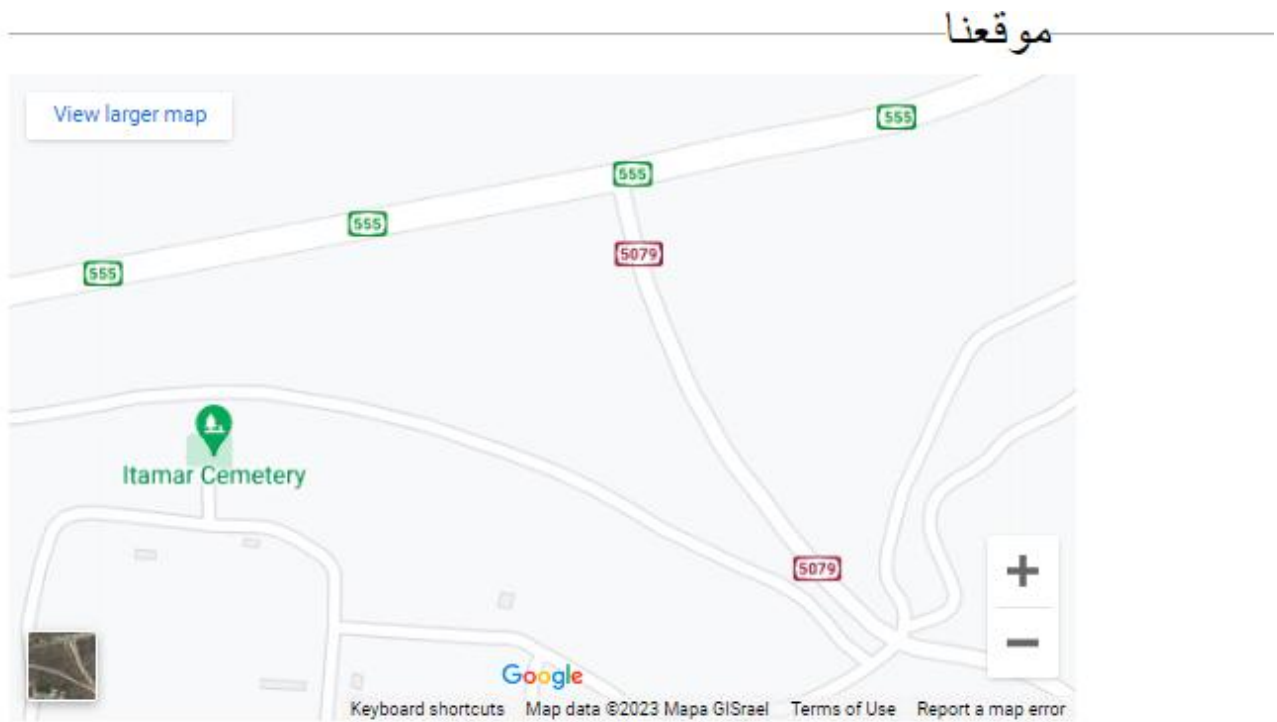


Figure 11: Location of the Store

We provided the Rating for the Store by storing the Ratings of all stores each day in another database. Then show it as a chart that can be altered using period filter (الاسبوع السابق، الشهر السابق، السنة السابقة), the user can filter the rating of the store by sub periods like last week.



Figure 12: Last Week Ratings

Also, he can filter by last month:



Figure 13: Last Month Ratings

Also, he can give a rating of the store by starts as following:



Figure 14: Rate the Store

### 5.2.4 Categories pages (eg. Kitchen page)

If the user wants to navigate to products page without specifying store name, he can navigate to categories pages like (المطبخ، السفارة، المنزل), in kitchen page for example, there are products for kitchen and filters to help Customer to get the best product for himself.

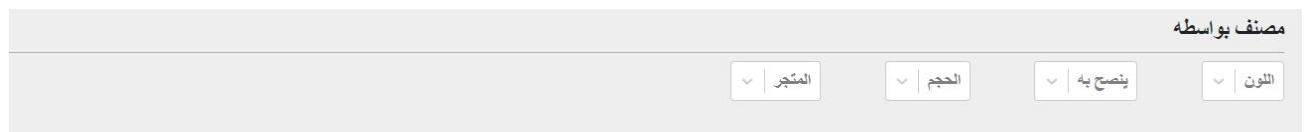


Figure 15: Filters in Categories Page

There are 4 filters, filter by color, size, store, recommends. And the same page has list of products as this image shows:



Figure 16: Kitchen Page Products

The user can retrieve the best product for himself by using filters, for example if the user selects the color white in filters, he will see only the products with color white and if he selects green, he will see only green and so on.

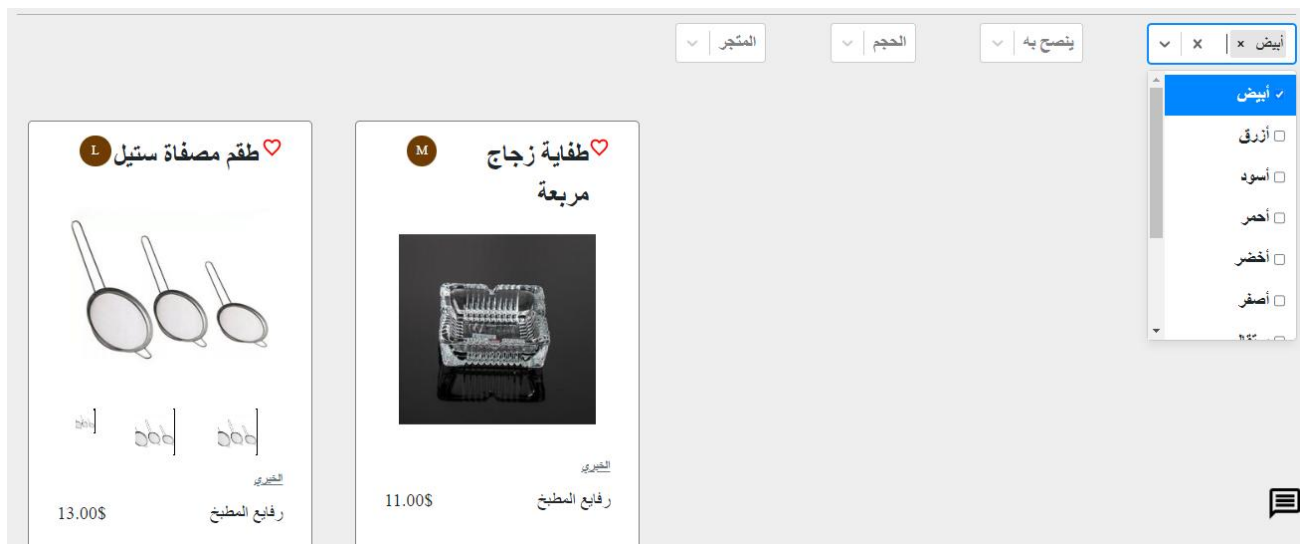


Figure 17: Filter by Color

Moreover, the user can filter the products depending on recommends (ينصح به). Which arrange products on the price: price from high to low, or from low to high, customers ratings or newest.

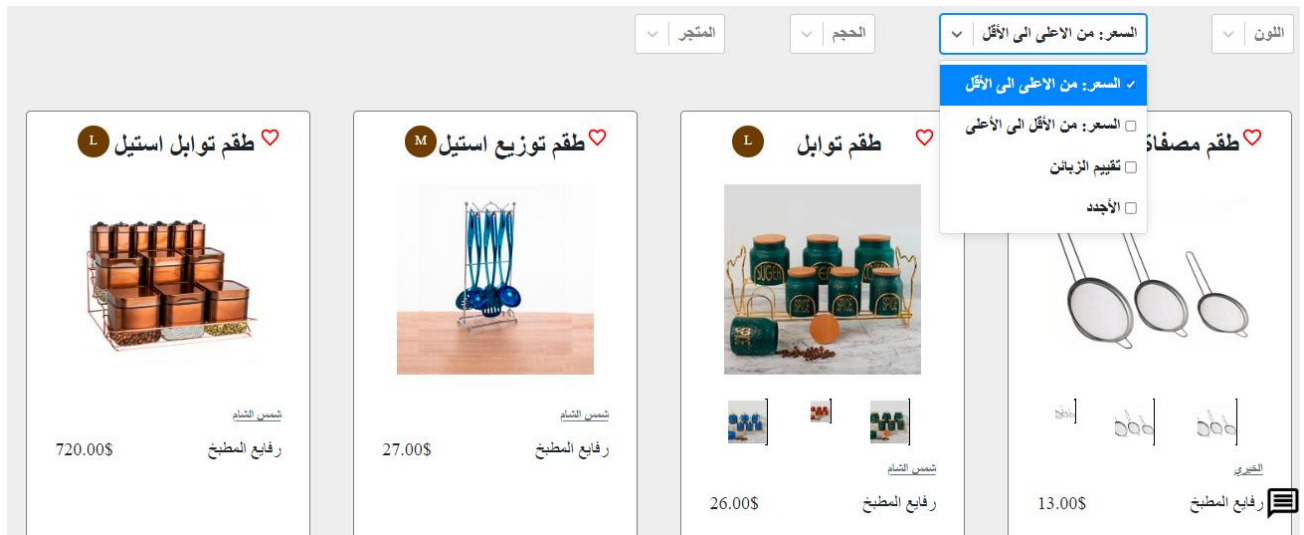


Figure 18: Arrange By filter recommends

And the user can filter depending on the size of the products:

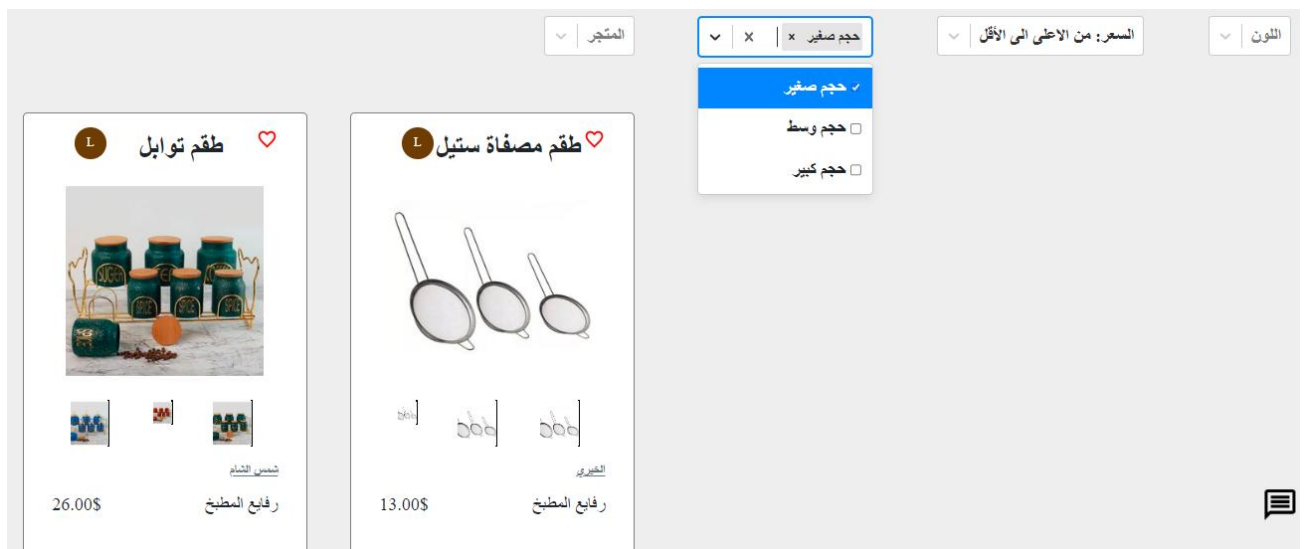


Figure 19: Filter by size

Also, the user can filter depending on the name of the store.

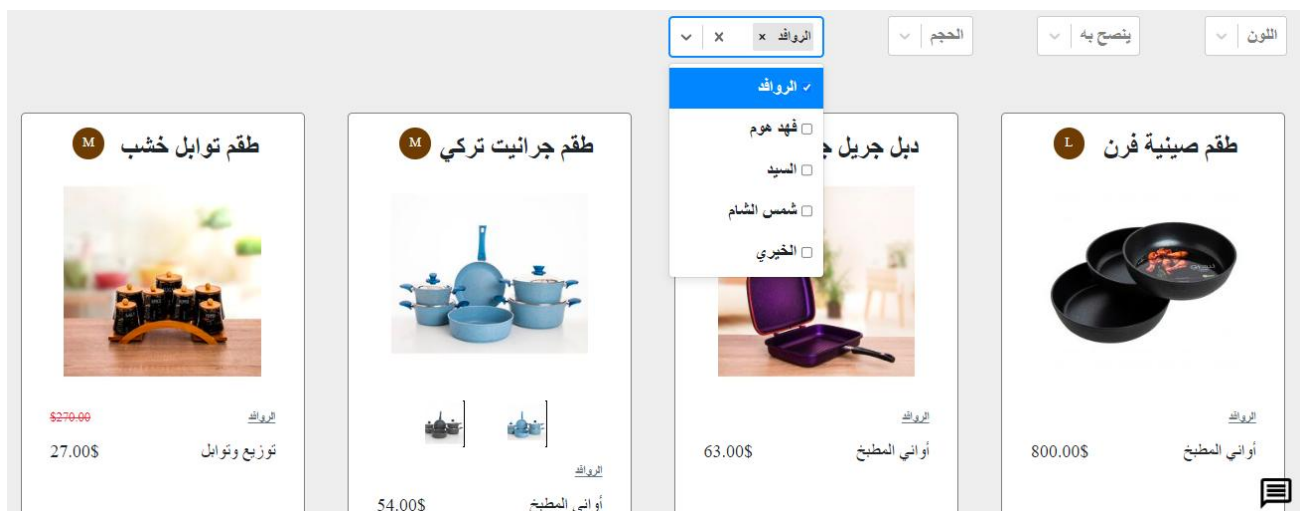


Figure 20: Filter by name of the Store

After the customer select his product, he can click on the product to navigate to product description page. The user will see in it a description about the product like the price of that's product, and he can select the Quantity which he needs to buy. Also, he can choose variants (size, color) he wants.



Figure 21: Product Description Page

In it we provide some recommended products that are similar to the product he selected previously. Also, he can write a review about the product and give his feedback.



Figure 22: Similar Products and Ratings Section

### 5.3.1 Payment method and Cart page.

The user can add any product he wants to buy it to his cart page in the choosing quantity (سلة التسوق),

the cart page will show him the quantity of products and he can increase or decrease the quantity as the user want, and finally the interface will give him a simple button (الدفع), he can press it to navigate to payment page.

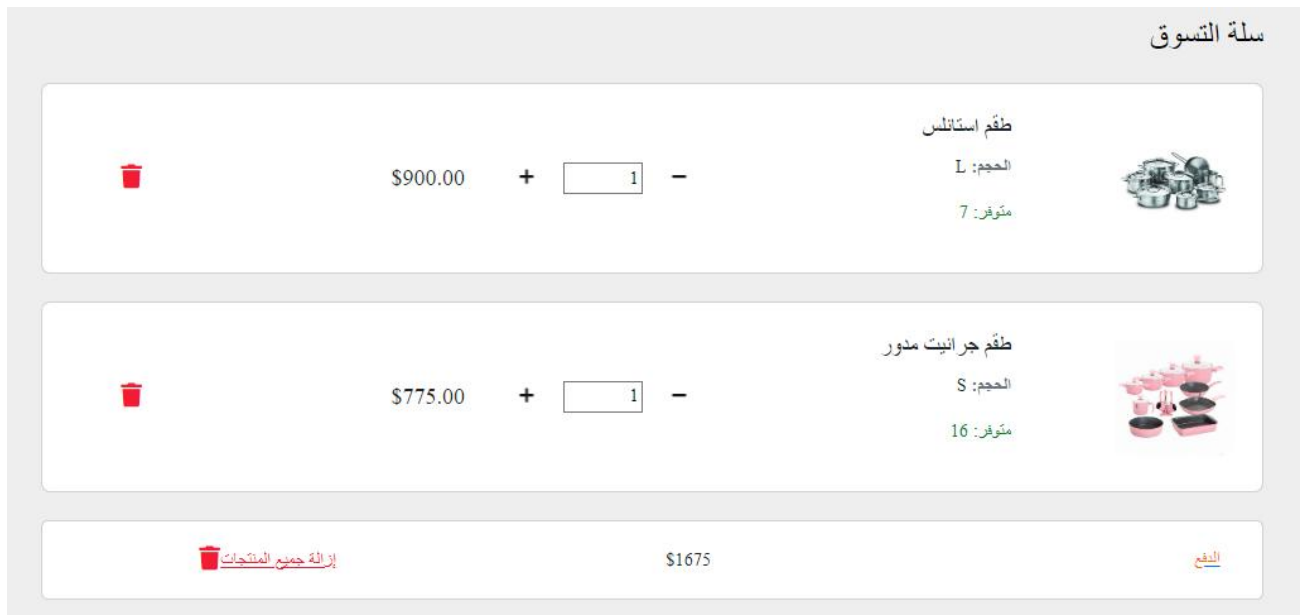


Figure 23: Cart Page

After the user press the pay button (كبسة الدفع), he will go to payment page, the user Then will fill his personal information and his address in it.

Copyright © Houseware 2023

Figure 24: Address Form

After that, the user will press next (التالي), to go to method of payment page, in it will set (يحدد) the payment method from drops down list like this.



Figure 25: Choosing method of payment

Then will press next(التالي), to go to review page (مراجعة الطلب), or he can go back to previous page if he wants to correct some information in address page.



Figure 26: Method of Payment Page

In review page, we will provide the user the summary of the payment and the total price for his cart.

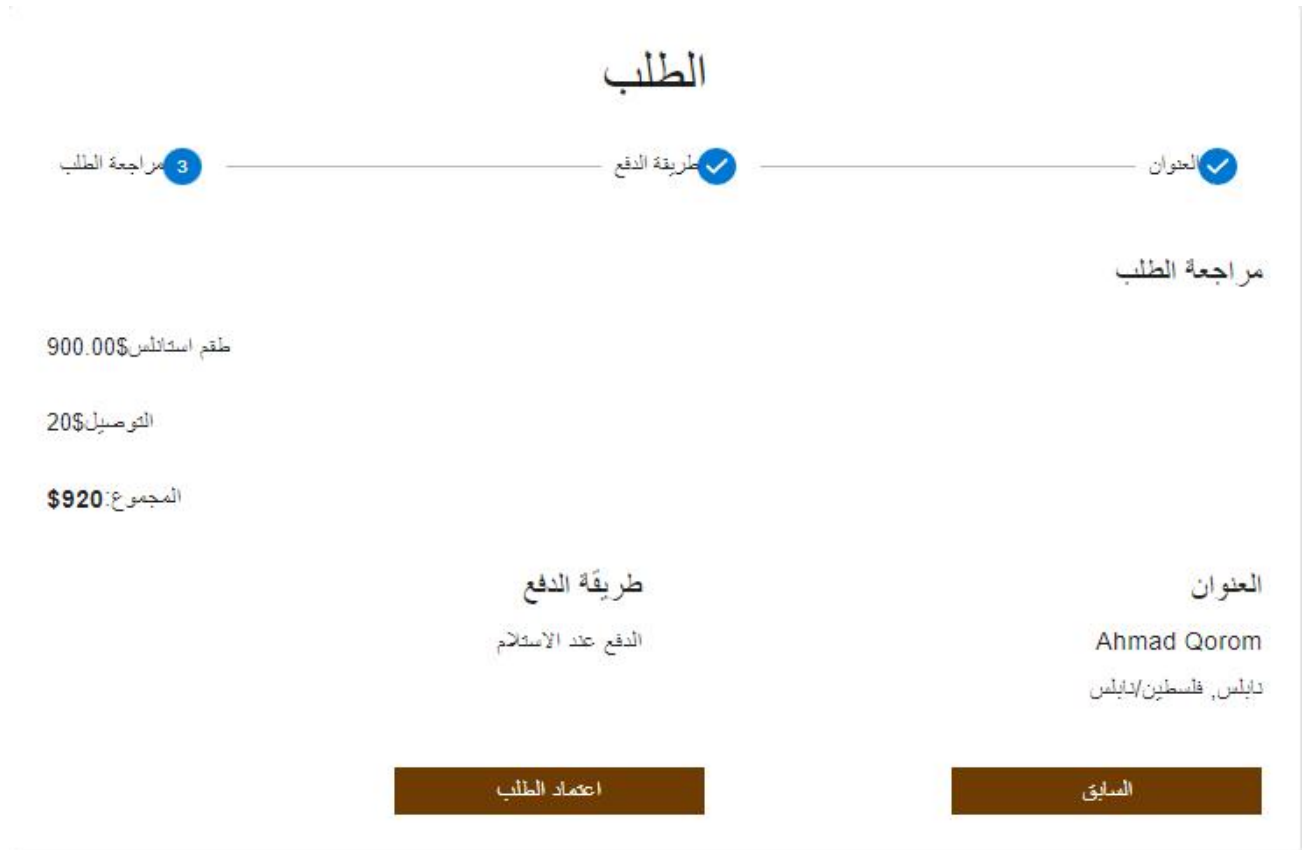


Figure 27: Review of Order Page

When the user presses OK (اعتماد الطلب), the payment will done successfully.



Figure 28: Order Done

Let's now talk about a feature that is necessary in discount seasons or in high order of one product situations, if we suppose the user want to buy 3 products of some product and this product have 5 total copies of this product in store. So, the user can buy it without any problem, but suppose if another user wants to buy the same product and the same quantity, so both users will go to payment page, so once one user of them complete all payment process, the second user cannot pay this product and the site will return him to cart page.

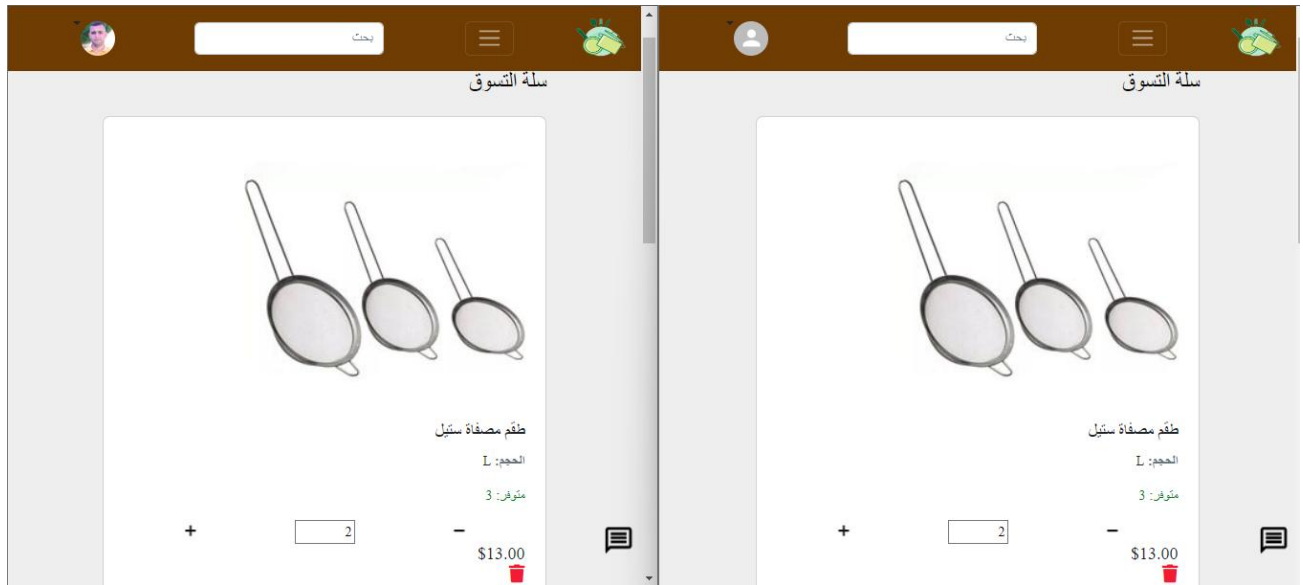


Figure 29: Two Purchases Example (1)

Both two users want to buy the same product and they added to the cart page the same product with same quantity, so now both users can go to payments process.

Now if one user of them go to payment page, and then the second user also go to payment page, when one user finishes the payment process and get the product he wants, the total quantity of this product will be 1 in the store, but the second user need 2 copies of this product, so the site will return him to cart page.

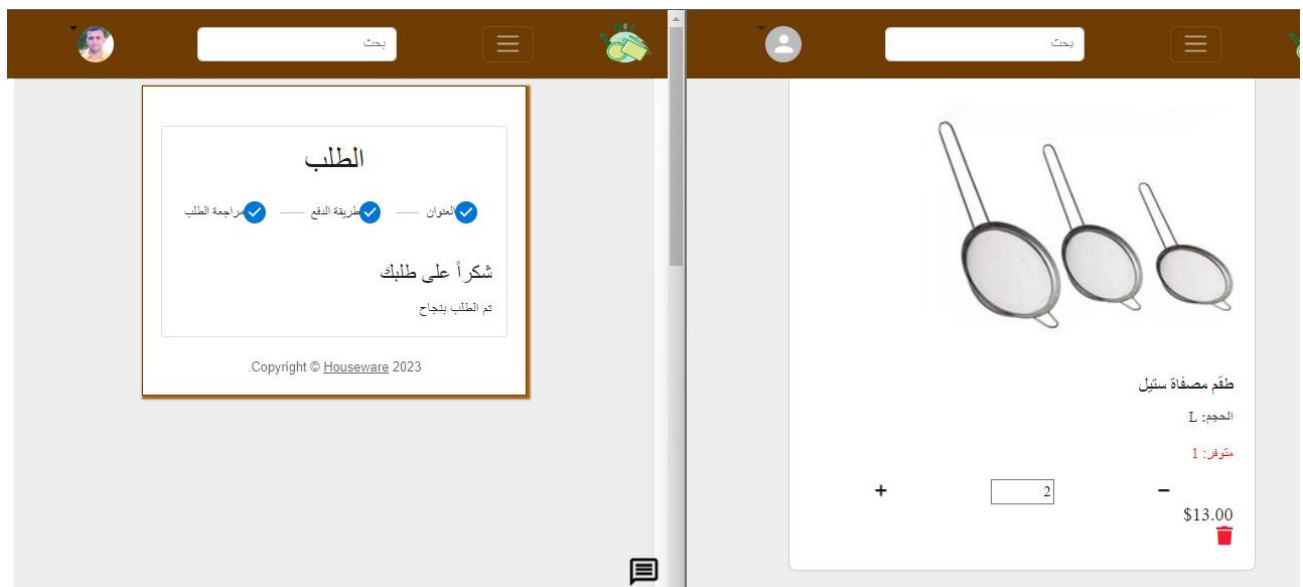


Figure 30: Two Purchases Example (2)

As we can see, the first user completes the payment information, but the second user return to cart page, and the quantity of this product will be 1 with color red, that's mean he cannot go to payment process, unless the second user decrease quantity of his product to 1.



Figure 31: Two Purchases Example (3)

So it checks two times if this product is available in the quantity ordered in. First in Cart page then second when the user clicks "اعتماد الطلب" button.

## 5.4.1 Admin Page

### 5.4.1.1 Main section(الرئيسية)

In admin page, we design an account for the store as admin. Admin accounts are the only accounts that can access admin page.

We designed the admin page different sections like main section(الرئيسية). In this section the store can see the rating of this store, and see a chart showing the sales of the store in last week. Also he can see a list of his products where he can filter and edit all of his products with ease.

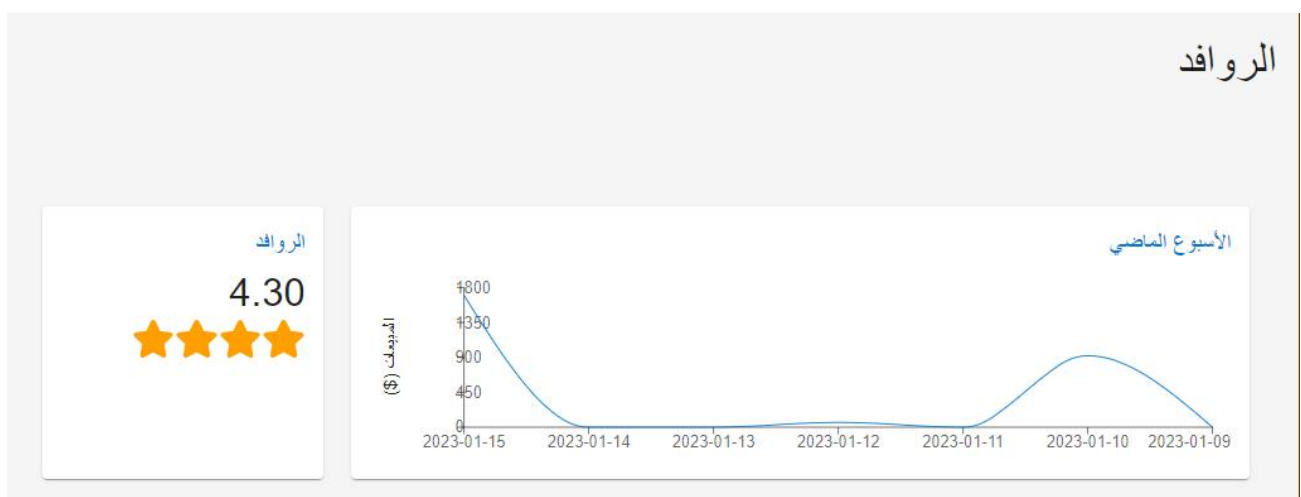


Figure 32: Admin Main Section

We provide in main section(الرئيسية), a list of all products in the store, we also put the percentage of discount, by selecting the products and specifying the percentage of discount, we can make update

to our products, on addition we provide a link add products to the Store.

الوصف	نوع 3	نوع 2	نوع 1	تعديل	السعر	التقييم	الصف	الفئة	الصورة	الاسم
K-809 ماسك دهن حمام	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	30.00\$	0	رفاق الحمام	المترول		دهن حمام
جلبى بالصينية رخامى 2406 635	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	208.00\$	0	حذف وبورسلين	السفرة		جلبى بالصينية
شاى بالصينية مدور بورسلين 007	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	15.00\$	0	حذف وبورسلين	السفرة		شاى بالصينية
12 قطعة بورسلين بمحمل ستيل	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	800.00\$	0	حذف وبورسلين	السفرة		12 قطعة بورسلين
بوب ارت اورانج 69 قطعة مربع	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	800.00\$	0	أركوبيركس	السفرة		بوب ارت اورانج

Figure 33: List of Store Products

In addition, you can delete selected products by selecting some products and press the delete button.

الوصف	نوع 3	نوع 2	نوع 1	تعديل	السعر	التقييم	الصف	الفئة	الصورة	الاسم
طقم جرابيت مدور 18 قطعة تركى	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	775.00\$	0	لواني المطبخ	المطبخ		طقم جرابيت مدور <input type="checkbox"/>

Figure 34: Delete Button

### 5.4.1.2 Statistics about the store

We add this section to show statistics about the Sales and ratings of this store in different periods as following:

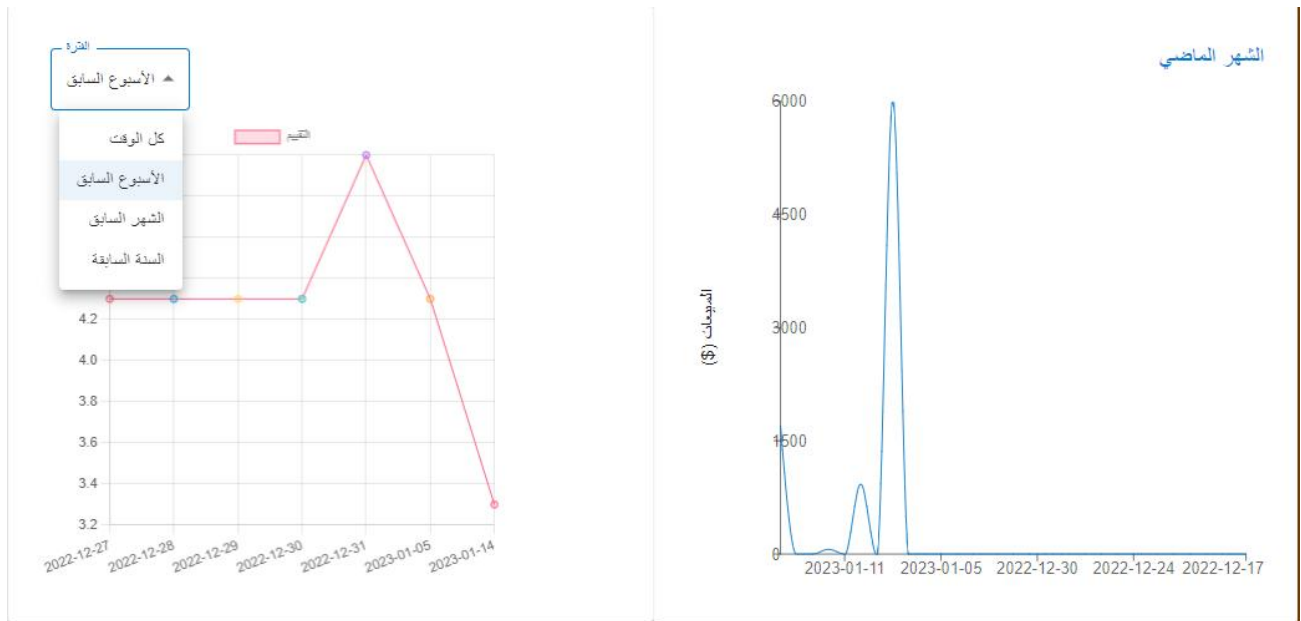


Figure 35: Statistics section

### 5.4.1.3 Add product

The Store can add product to his products by using form to fill the information about the product as following:

Figure 36 shows a form titled 'اضف منتج' (Add Product). The form includes the following fields and controls:

- الإسم (Name): Text input field.
- السعر (Price): Text input field.
- نسبة الخصم (Discount Percentage): Text input field.
- الكمية (Quantity): Text input field.
- اللون (Color): Dropdown menu.
- الفئة (Category): Dropdown menu.
- قابل للإرجاع (Returnable): Checkbox, currently checked.
- الصفحة (Page): Dropdown menu.
- الحجم (Size): Dropdown menu.
- الوصف (Description): Text area.
- إضافة صورة (Add Image): Button.
- إضافة (Add): Button.

Figure 36: Form to add Product

Using this form you can add a new Product by filling the text fields or select form the checkbox with

data (name,price,discount percentage, quantity, returnable, color,size,category,sub-category,description and picture).

You can't add two products with the same name because that will make a confusion on the customer side.

#### 5.4.1.4 List of Products

This section we mentioned some of its functionality earlier because it is also in the main section but here, we will go deeper.

Here you can order the products by name category etc.... ascending or descending and filter the products on these fields also with ease.

منتجاتك

الوصف	نوع 3	نوع 2	نوع 1	تعديل	السعر	التقييم	الصف	الفئة	الصورة	الاسم
K-809 ماسك دش حمام	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	30.00\$	0	رفاع الحمام	المنزل		ماسك دش حمام
3... قطعة تركي مقاسات 26-28-30	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	800.00\$	0	أواني المطبخ	المطبخ		طقم صينية فرن
12... قطعة بورسلين بحامل ستيل	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	800.00\$	0	خفف وبورسلين	السفرة		طقم شاي بورسلين
007 شاي بالصينية منور بورسلين	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	15.00\$	0	خفف وبورسلين	السفرة		طقم شاي بالصينية
2406 635 جيلي بالصينية رخامي	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	208.00\$	0	خفف وبورسلين	السفرة		طقم جيلي بالصينية

1-5 of 11

[إضافة منتج](#)

Figure 37: Order Products Descending by name

منتجاتك

الوصف	نوع 3	نوع 2	نوع 1	تعديل	السعر	التقييم	الصف	الفئة	الصورة	الاسم	
القيمة	عامل	الأعمدة	السفر	يحتوي	الفئة	<a href="#">تعديل</a>	800.00\$	0	أر كوبركس	السفرة	طقم ار كوبركس
2406 635 جيلي بالصينية رخامي	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	208.00\$	0	خفف وبورسلين	السفرة		طقم جيلي بالصينية	
007 شاي بالصينية منور بورسلين	<a href="#">إضافة نوع</a>	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	15.00\$	0	خفف وبورسلين	السفرة		طقم شاي بالصينية	
12... قطعة بورسلين بحامل ستيل	<a href="#">إضافة نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل نوع</a>	<a href="#">تعديل</a>	800.00\$	0	خفف وبورسلين	السفرة		طقم شاي بورسلين	

4 rows selected

1-4 of 4

[إضافة منتج](#)

Figure 38: filter Products by category and select them

After that you can select one or all of them to make a discount on them or delete them as mentioned earlier.

In addition, you can edit the main information about the product by pressing "تعديل" like

(name,category,sub-category,returnable,description), Or adding a variant to the same product by pressing "تعديل نوع" that has (price,quantity,discount,size,color,image) different or same as the main variant.

Also, you can edit each one of the variants by pressing "إضافة نوع" to edit these values (price, quantity, discount, size, color, image).

## تعديل منتج

الاسمدبل جريل جرانييت

63

---

20

---

14

اللونبنفسجي

الغنةالمطبخ

قابل للإرجاع

المنطقأواني المطبخ

الحجمL

الوصف2207 دبل جريل جرانييت مقاس 36

×أضيفت الصورة

تعديل

Figure 39: edit Product Form

We can also add variants of same product from product list (قائمة منتجاتك), by typing in add type to navigate to variation form page as Figure 40.



Figure 40: add Variation Form

And also, we can edit in some product to make discount for example and the quantity of this product, as in Figure 41.



Figure 41: Edit Variation Form

## 5.5 Chat and Notification

This is an important feature in our application where the customers can chat with the Stores and vice versa. Also, when the Customer or the Store Receive a Message, it will show on the notification section ”التنبيهات” until he read it.

This is the only Feature that is implemented using Firebase. Because Firebase provide us with the tools to build chat Messaging and notifications easily and efficiently.

Here is an example of messages between Customer and Store الروافد.

First the Customer send ”السلام عليكم” to the الروافد Store as shown:



Figure 42: Customer Send a Message to Store

Then when the Store open his account, he will see a notification that “Issa Abbadi” send you a message and the chat component open.



Figure 43: notification on messages

After he press on the user “Issa Abbadi” and see his message the notification will be removed.

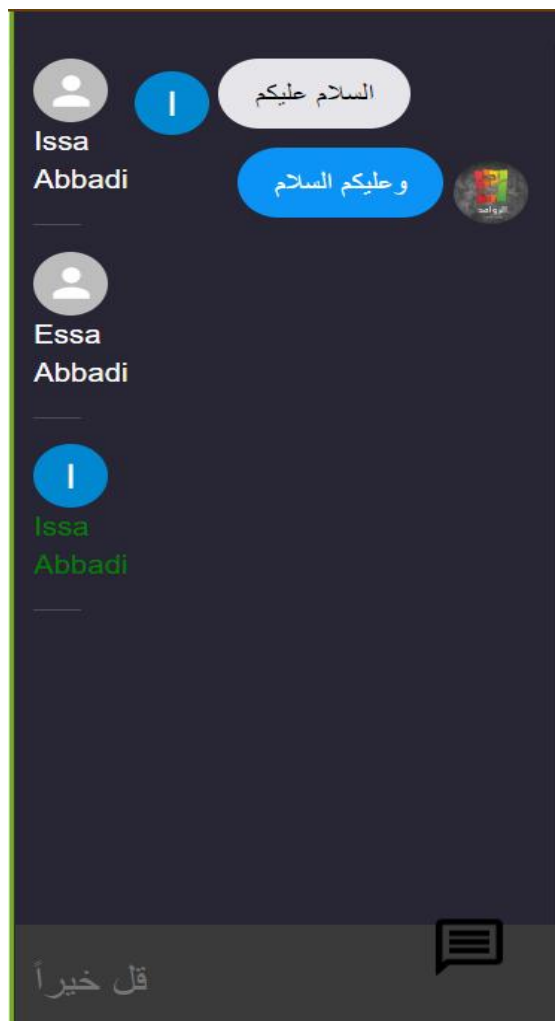


Figure 44: Store replied to the Customer

Here is the notification section after the Store sees the message.



Figure 45: Notification removed

There is also another type of notification used in this application which is notifications from admin to Stores. Usually for discount seasons or holidays. Also, if there is an error. Here is an example of it.



Figure 46: Notifications from the Admin

## 5.6 Wish List Page

One of features we designed is wish list, when the customers shopping between products and see some product and he needs him in specified period of time like Islamic holidays, discount seasons...

The customer can add this product to wish list to buy it later or he can drop it later.



Figure 47: Wish List Page

the user can add these products to cart page or he can drop them.

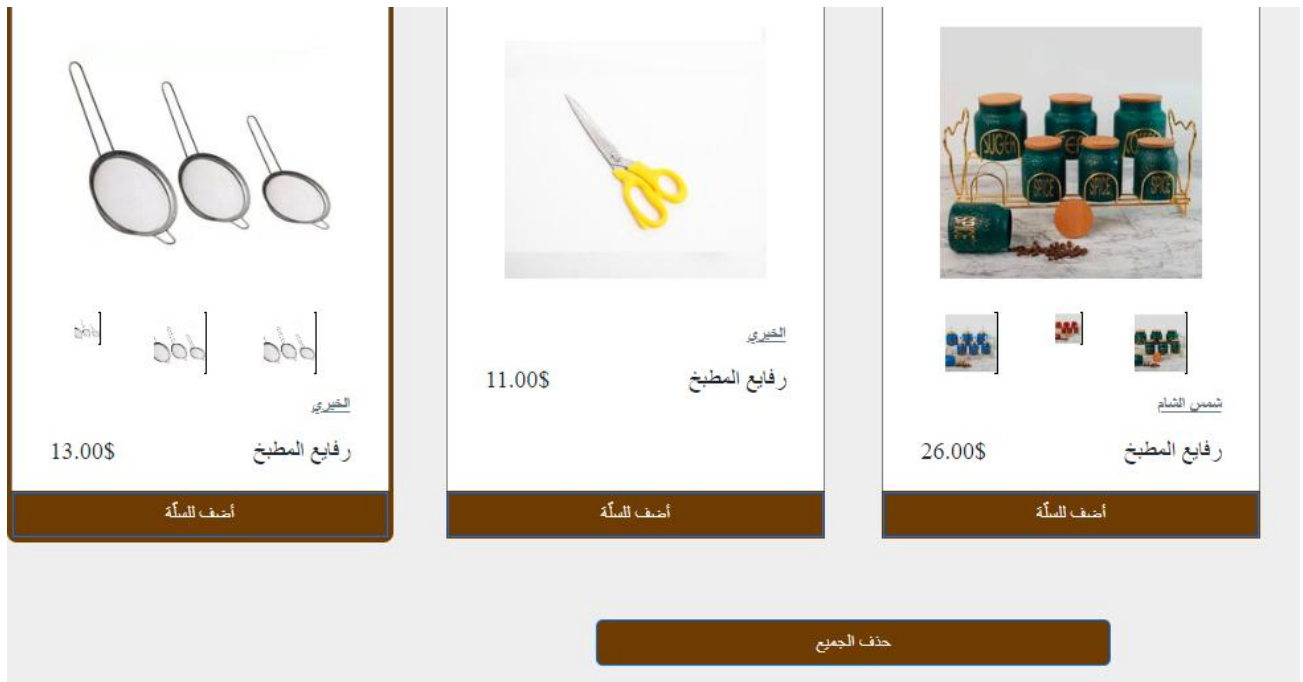


Figure 48: Remove Products from Wish List

# Chapter 6

## Results & Discussion

### 6.1 Final Application

The final product is a web application that allows Customers to:

1. Buy from different Store and Choose the best Products in the best size and color.
2. Using different Sections, Filters and Search they can go to specific product quickly.
3. See a comparison between Stores and know details about each one of them.
4. Chat with the store and send and receive messages with notifications.

Also allows Stores to:

1. Add products in different Sizes and Colors and edit them with ease.
2. Show them different Statistics about their Store and other Stores to compete with them.
3. Make discount on multiple Products with ease (in holidays for example).

### 6.2 Project Outcomes

Hopefully, this application will:

1. Management the different Housewares Stores in it.
2. Make a competitive environment between different Housewares Stores.
3. Allow Customers to get the best value for price products.

# Chapter 7

## Conclusion & Future Work

### 7.1 Summary

Housewares is a Management Housewares Stores application aimed to create a competitive environment. While also lead Customers to the best products. There are a lot of other alternatives, but most of them are in other fields, limited to one Store or contain a lot of unnecessary features which makes the application overwhelming and not easy to get into.

Additionally, Housewares aimed to enhance the communication experience between the Customers and Stores either by chatting and notification system or by rating products and stores.

We believe that Housewares fulfilled the goal for which it was intended. Due to the short timeline and emphasis on more crucial elements, a small group of minor features that we had intended to include in the project were regrettably deleted.

### 7.2 Future Work

There are some aspects that Housewares could be improved in. that includes:

1. The mobile application doesn't contain all the features that the website provides. These features could be implemented.
2. Adding more Charts with filters in Home page to make the competition more active.
3. Edit the User interface to be more user friendly and Consistent.
4. Allow Stores to make discounts on products at certain times automatically.
5. Adding a reporting system to report extra information to the Housewares Stores.

# Bibliography

- [1] React. url: <https://reactjs.org/>. (accessed: 21.12.2021) (page 10).
- [2] MUI. url: <https://mui.com/>. (accessed: 21.12.2021) (page 10).
- [3] MUI Lab. url: <https://mui.com/components/about-the-lab/>. (accessed: 21.12.2021) (page 11).
- [4] ExpressJS. url: <https://expressjs.com/>. (accessed: 21.12.2021) (page 11).
- [5] Mongoose. url: <https://mongoosejs.com/>. (accessed: 21.12.2021) (page 11).
- [6] MongoDB Atlas. url: <https://www.mongodb.com/atlas>. (accessed: 21.12.2021) (page 11).
- [7] Firebase. url: <https://firebase.google.com/>. (accessed: 21.12.2021) (page 12).
- [8] React Router. url: <https://reactrouter.com/>. (accessed: 21.12.2021) (page 12).
- [9] bcrypt. url: <https://www.npmjs.com/package/bcrypt>. (accessed: 21.12.2021) (page 12).
- [10] axios. url: <https://github.com/axios/axios>. (accessed: 21.12.2021) (page 12).