

USER CENTRED WEB DESIGN
FOR AN E-COMMERCE WEBSITE

TABLE OF CONTENT

LIST OF FIGURES	3
INTRODUCTION	4
ANALYSIS	5
REQUIREMENT ELICITATION TECHNIQUES.....	5
1. PERSONA	5
2. CARD-SORTING	7
3. QUESTIONNAIRE	9
4. REQUIREMENTS LIST	10
DESIGN.....	11
1. USER INTERFACE PROTOTYPING	11
2. USER INTERFACE DESIGN	12
IMPLEMENTATION.....	15
1. SYSTEM BACK-END IMPLEMENTATION	15
A. REGISTRATION	15
B. SIGN IN	18
C. PRODUCT ENTRY	19
D. BASKET	20
E. FILTERING	22
TESTING.....	23
1. BLACKBOX TESTING	23
2. USABILITY TESTING	24
CONCLUSION.....	26
REFERENCES	27

LIST OF FIGURES

Figure 1.Persona-1	6
Figure 2.Persona-2	7
Figure 3.Persona-3	7
Figure 4.Card Sorting UXtweak UI	9
Figure 5.Desktop UI Prototype	11
Figure 6.Mobile UI Prototype	11
Figure 7.Tablet PC UI Prototype	12
Figure 8.Desktop UI.....	13
Figure 9.Mobile UI	14
Figure 10.Tablet PC UI	14
Figure 11.Sign Up Button	15
Figure 12.Registration Form	15
Figure 13.Registration PHP codes	16
Figure 14.Registration DB	16
Figure 15.Validation of the required fields	16
Figure 16.Validation of email format	17
Figure 17.Validation of checkbox	17
Figure 18.Password check	17
Figure 19.Encrypted password DB	18
Figure 20.Encrypted password PHP	18
Figure 21.Sign In Form.....	18
Figure 22.Sign In PHP	18
Figure 23.Sign In Error	19
Figure 24.Session.....	19
Figure 25.Product Entry Form	20
Figure 26.Product Entry DB	20
Figure 27.Basket Form.....	20
Figure 28.Basket PHP	21
Figure 29.Adding Item to Basket PHP	21
Figure 30.Remove Items from Basket PHP	21
Figure 31.Payment PHP.....	22
Figure 32.Filtering by Category.....	22
Figure 33.Filtering by Category php codes.....	22
Figure 34.Interaction Design Foundation - IxDF, 2016.....	25

INTRODUCTION

The emergence of online platforms has revolutionized the way digital products are bought and sold. This study focuses on the development of a dedicated e-commerce website named "Greenwich Gallery" aiming to facilitate the sale of images featuring iconic landmarks situated in the Old Royal Naval College.

The "Greenwich Gallery" website not only provides users with an experience to register, log in, search, browse, and purchase images but also ensures a visually appealing and user-friendly interface. In addition to this, admin users are able to operate CRUD on items sold on the website.

The comprehensive requirement analysis involved understanding user needs and preferences through various techniques, with a specific focus on User Experience Design (UXD) principles. The implementation phases, both front-end and back-end, have been meticulously detailed in this report, providing evidence of the website's construction and functionality.

To guarantee a user-centred design, testing and heuristic evaluation studies were conducted. Black box testing assessed the functionality of the website without delving into its internal code, ensuring that users could interact with the platform effortlessly. Heuristic evaluation, on the other hand, involved assessing the interface against recognized usability principles, contributing to the overall user-centred design of the "Greenwich Gallery" website.

ANALYSIS

REQUIREMENT ELICITATION TECHNIQUES

Requirements engineering involves the activities of extracting, comprehending, defining, and confirming the requirements of customers and users. (Nora et al., 2006) In this stage, users and developers work together to plan how the final application will be. They talk about what the application should do, the reports it should create, what information to keep, and how to keep it secure. It's really important because even one change in what's needed might mean redoing the whole application. That means more work and more time to finish. (Hussain & Sumari, 2016)

There are various types of requirement analysis techniques; and among them, persona, card sorting, surveys, and scenario analysis have been utilized.

1. PERSONA

A persona is a user model that looks at what an individual wants to achieve with a tool. This model is a tool used in software and product design. It's similar to traditional user profiles, but it has some key differences. Essentially, it represents typical characteristics of actual or potential users. (Blomkvist, 2002)

The persona method involves the creation of fictional characters representing specific user types or groups. Designers employ personas to craft user-centred solutions that meet the needs of the target audience during the design and development process. By providing a shared language and understanding, personas assist teams in developing products and services that effectively address users' demands. (Cooper et al., 2014)

To develop a robust persona, adhere to the following steps:

Research users: Use interviews, surveys, and other methods to gather info on user behaviors, preferences, and challenges.

Analyse data: Look at the information and identify recurring themes and patterns.

Create personas: Develop personas that represent different user groups based on the collected data.

Add details: Bring the persona to life by creating a story that includes the user's goals, strategies, and challenges.

Verify: Test and confirm that the personas accurately represent the target audience (Nielsen, 2002).

In creating personas for this e-commerce site, decision-making has been carried out by engaging with prospective users through interviews, surveys, and analysing published data. This process has facilitated an understanding of how users behave and what they prefer. Through this comprehensive approach, the aim is to develop personas that accurately represent the diverse needs and preferences of our users.

Persona.1


 JANE THE ARTIST	<i>Tech-savvy</i> <i>Mobile phone user</i> <i>Depending on social media influencers' preference</i>
<i>Age:</i> 25 <i>Profession:</i> Student at Art <i>Education level:</i> <i>Bachelor's Degree</i> <i>Location:</i> London	<ul style="list-style-type: none">• A seamless and user-friendly shopping experience• Look for cheaper options• Reliable product• The ability to receive discounts and coupons• The ability to earn rewards for their purchases• Product sharing on social media channels.• Chatbot for assistance

Figure 1.Persona-1

Persona.2


 <p>AUNTY</p>	<p><i>Intermediate user</i> <i>Desktop pc, mobile phone</i> <i>People's recommendation</i></p>
<p>Age:65 Profession: Collector Education level: College Location: Brighton</p>	<ul style="list-style-type: none"> • Simple navigation • She wants detailed information • real time assistance • Large font size, High contrast colors • The ability to find products at the best possible price • shop for products online and pick them up in store

Figure 2.Persona-2

Persona.3


 <p>DOCTOR MIKE</p>	<p><i>Advanced user</i> <i>Mobile phone,</i> <i>Word of mouth</i></p>
<p>Age:40 Profession: Doctor Education level: Master's Degree Location: Glasgow</p>	<ul style="list-style-type: none"> • A quick and easy checkout process • The ability to purchase products using their preferred payment methods • The ability to return or exchange products easily • Fast delivery/ Easy Return • Mobile shopping

Figure 3.Persona-3

2. CARD-SORTING

Card Sorting is a recognized technique for gathering information (Barrett and Edwards, 1995) and has been extensively applied in Psychology, Knowledge Engineering, Software Engineering, and Web Site Design. In Requirements Engineering, it is considered the most efficient method for understanding problem domains. (Maiden and Rugg, 1996)

Card sorting has been proven to be a valuable elicitation tool with several advantages as follows:

- It enables the examination of respondents' recall knowledge regarding the domain entity. (Rugg and McGeorge, 1997)
- It is effective in distinguishing between high and low-level problems. (Barrett and Edwards, 1995)
- Card sorts provide deeper insights into the target population's perspectives on the topic (Zimmerman and Akerelrea, 2002)
- The results of card sorting can be utilized as input for other techniques and further analysis (Upchurch et al., 2001)
- They can also contribute to the development of hierarchies or classifications, such as information architectures for websites.(Zimmerman and Akerelrea, 2002)
- The card sorting process is efficient, cost-effective, and manageable for researchers. (Upchurch et al., 2001) (Rugg et al., 1992)

The *uxtweak* (UXTweak,2023) page was used for the card sorting study, where users were asked, according to the *open card sort* method, to determine subcategories within pre-defined areas for main categories. Ten users participated in the open card sort exercise through an online platform, sorting the subcategories. Based on the sorting results, the website's category design was adjusted.

The relevant study can be accessed through this link:
<https://study.uxtweak.com/cardsort/VAN8Fe6ofleNJ0v7n0IFy>

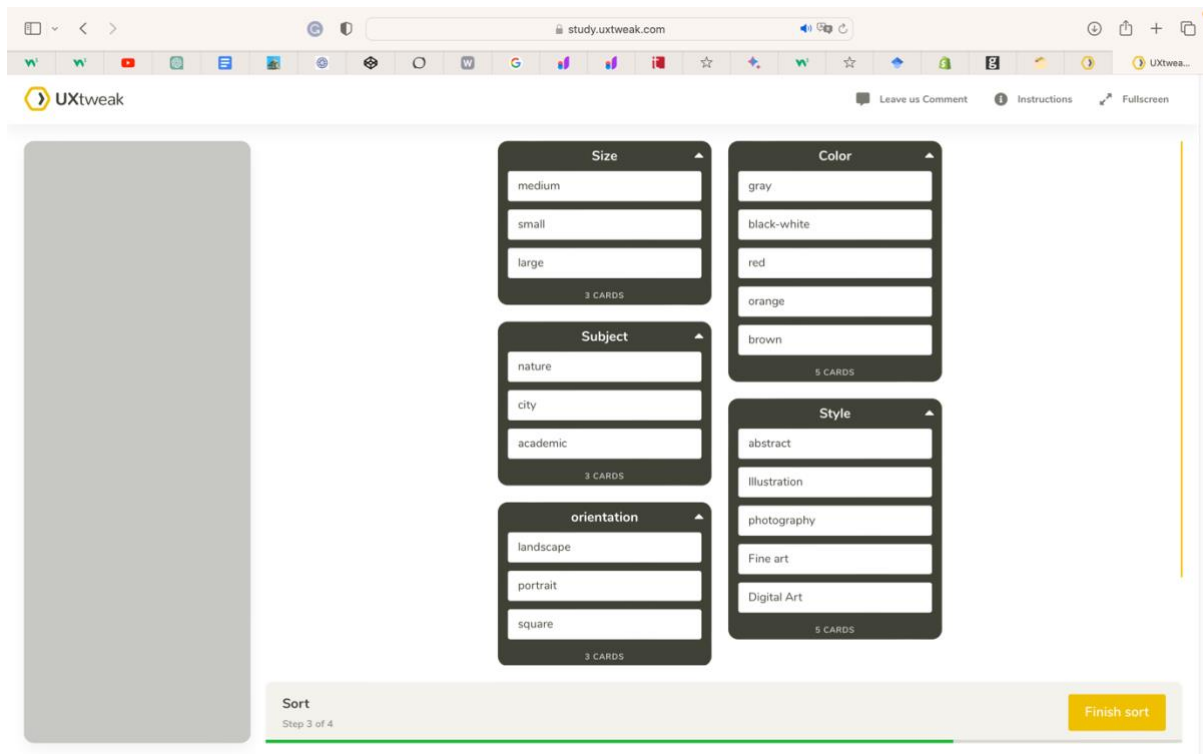


Figure 4. Card Sorting UXtweak UI

3. QUESTIONNAIRE

The questionnaire is a cost-effective and time-efficient method for gathering requirements from a large number of individuals. A carefully crafted questionnaire proves beneficial in eliciting clear and consistent requirements from stakeholders. The effectiveness of the questionnaire is contingent on the quality of its design and the facilitator's skills in conducting the session. (Tiwari, Rathore, & Gupta, 2012)

To better understand potential users and to be able to create a user-centred design, a survey has been conducted that includes the following questions:

Question List

1. Age
2. Gender
3. Location
4. Technology competency
5. How frequently do you shop online?

6. What devices do you use while shopping online?
7. Do you need any physical, health or social consideration for shopping online?
8. How do you prefer to receive assistance or support from the website?
9. What payment methods do you use for online shopping?
10. What kind of products do you mostly buy online?

4. REQUIREMENTS LIST

The needs identified are determined as a result of the applied requirement elicitation techniques, as seen in the table below:

REQUIREMENTS	
FUNCTIONAL REQUIREMENT	NONFUNCTIONAL REQUIREMENTS
<i>Registration</i>	<i>Encrypted password format</i>
<i>Login</i>	<i>Normalisation</i>
<i>Search</i>	<i>User-Centred Design</i>
<i>Browse</i>	<i>Responsive design</i>
<i>Pagination</i>	<i>Valid e-mail format</i>
<i>Adding to Basket</i>	<i>Validation of user input correctness</i>
<i>View basket</i>	
<i>Calculate Total</i>	
<i>Update Basket</i>	
<i>Purchase</i>	
<i>Add product</i>	
<i>Delete product</i>	
<i>Update product</i>	
<i>Logout</i>	

DESIGN

1. USER INTERFACE PROTOTYPING

To ensure accessibility and usability, responsive design principles have been adopted. Accordingly, below are designs tailored for three different screen sizes: one for desktop, the second for mobile phones, and the last one designed for tablet computers. All illustrations regarding wireframes have been designed on draw.io. website.

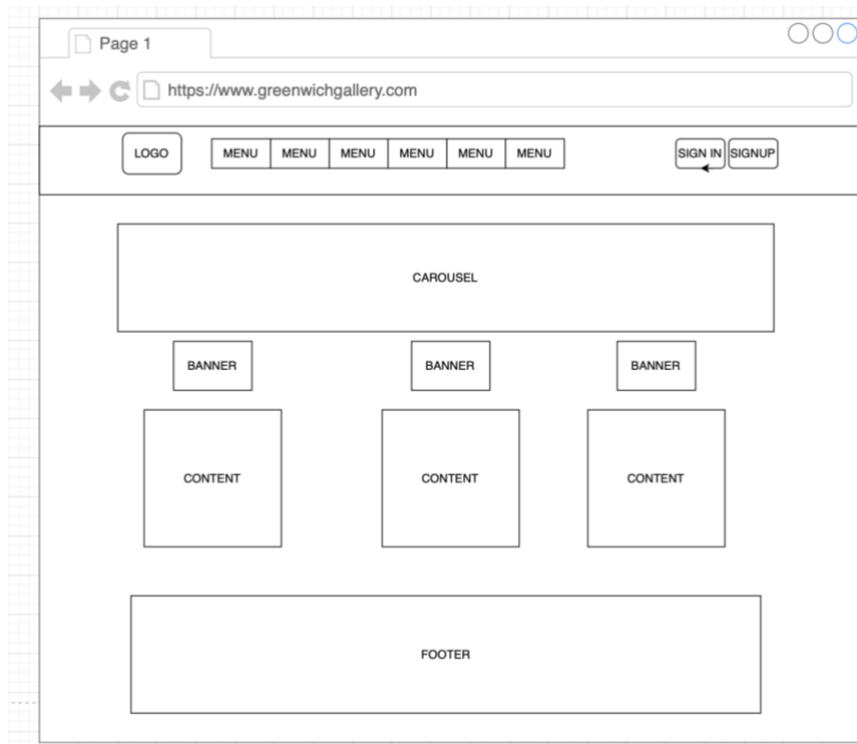


Figure 5.Desktop UI Prototype

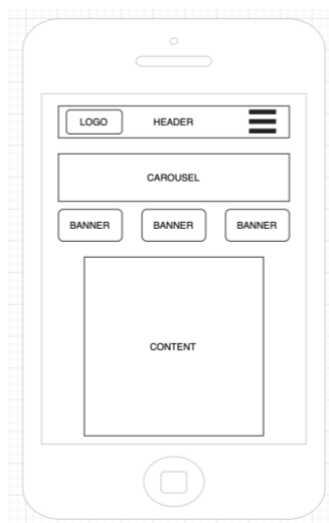


Figure 6.Mobile UI Prototype

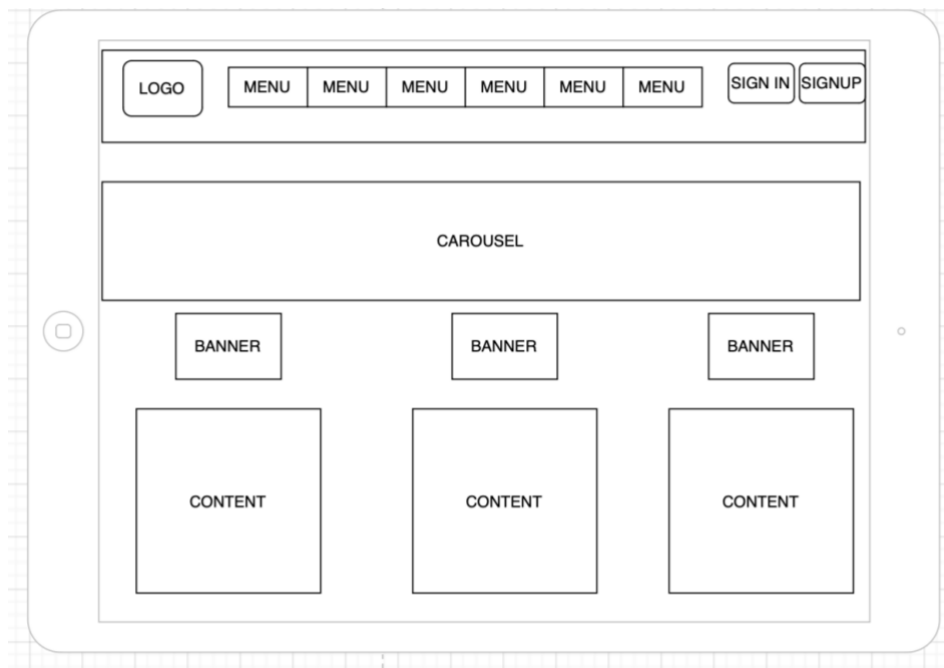


Figure 7. Tablet PC UI Prototype

2. USER INTERFACE DESIGN

For the front-end form design, HTML and certain Bootstrap elements have been employed, with Visual Studio Code chosen as the preferred Integrated Development Environment (IDE).

The desktop version of the index.php is depicted below. The navigation section features a logo, image categories for user filtering, and two buttons for login and signup. Subsequent elements consist of a carousel, card view, and a footer.

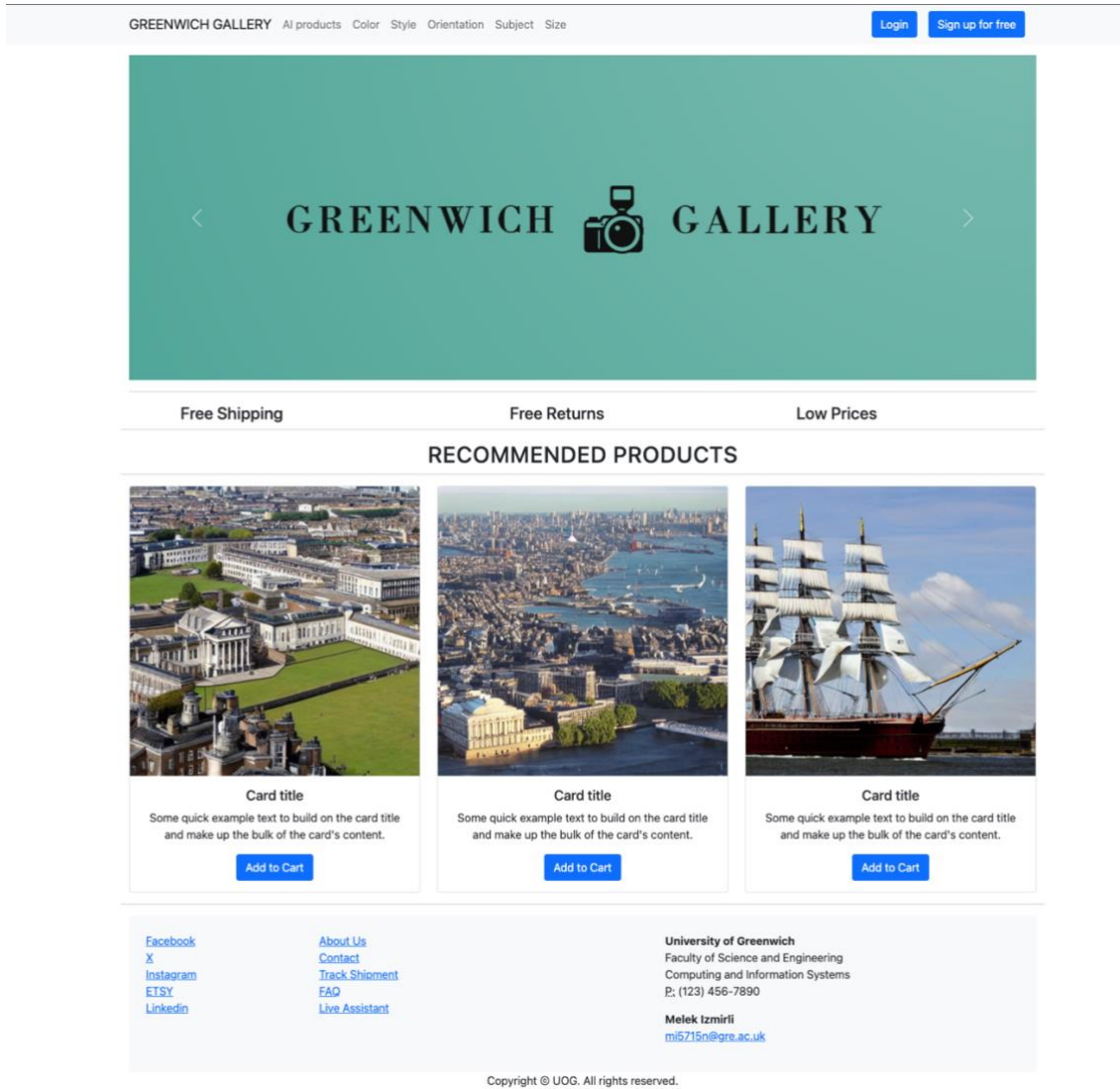


Figure 8.Desktop UI

To demonstrate the appearance on different devices, relevant UI views were obtained using the “Inspect” feature in the “Google Chrome” browser. The first image below is designed to accommodate iPhone 14 Pro Max users, while the second image is tailored for iPad Pro users, providing UI views suitable for each device.

All images serve as indicators of how responsive the website is, showcasing its adaptability to different devices.

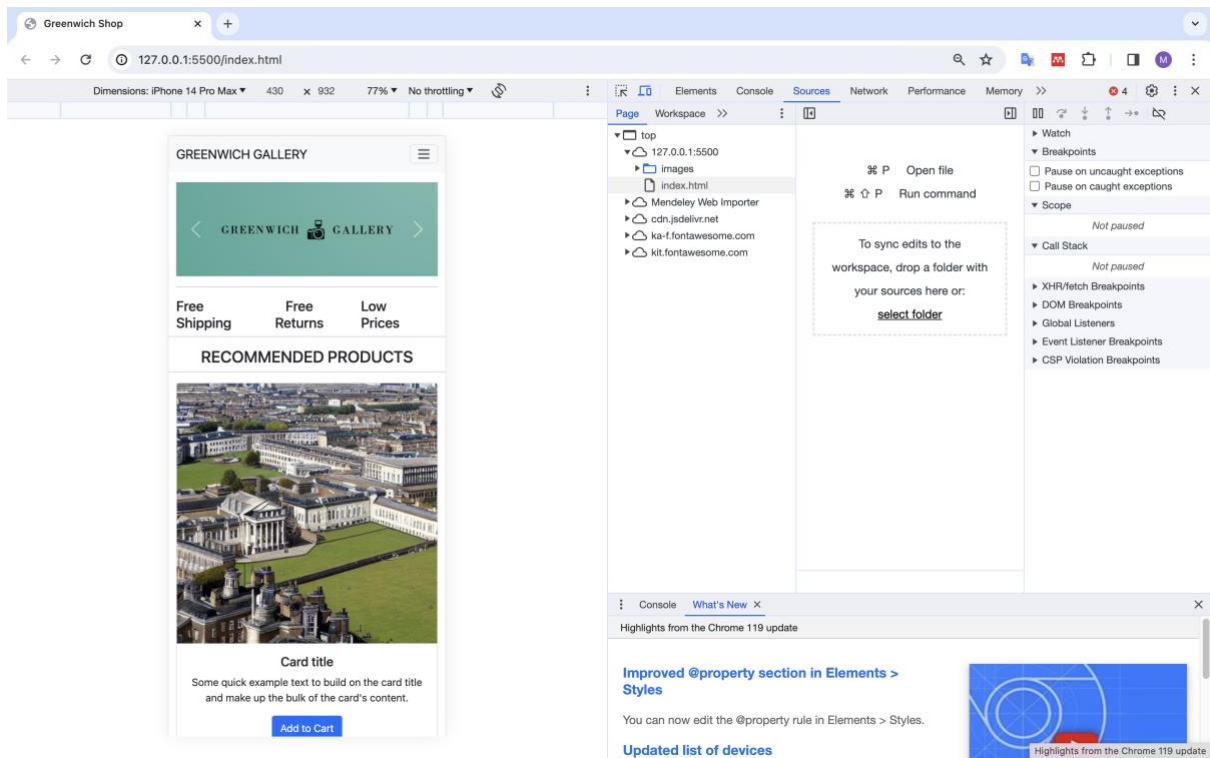


Figure 9.Mobile UI

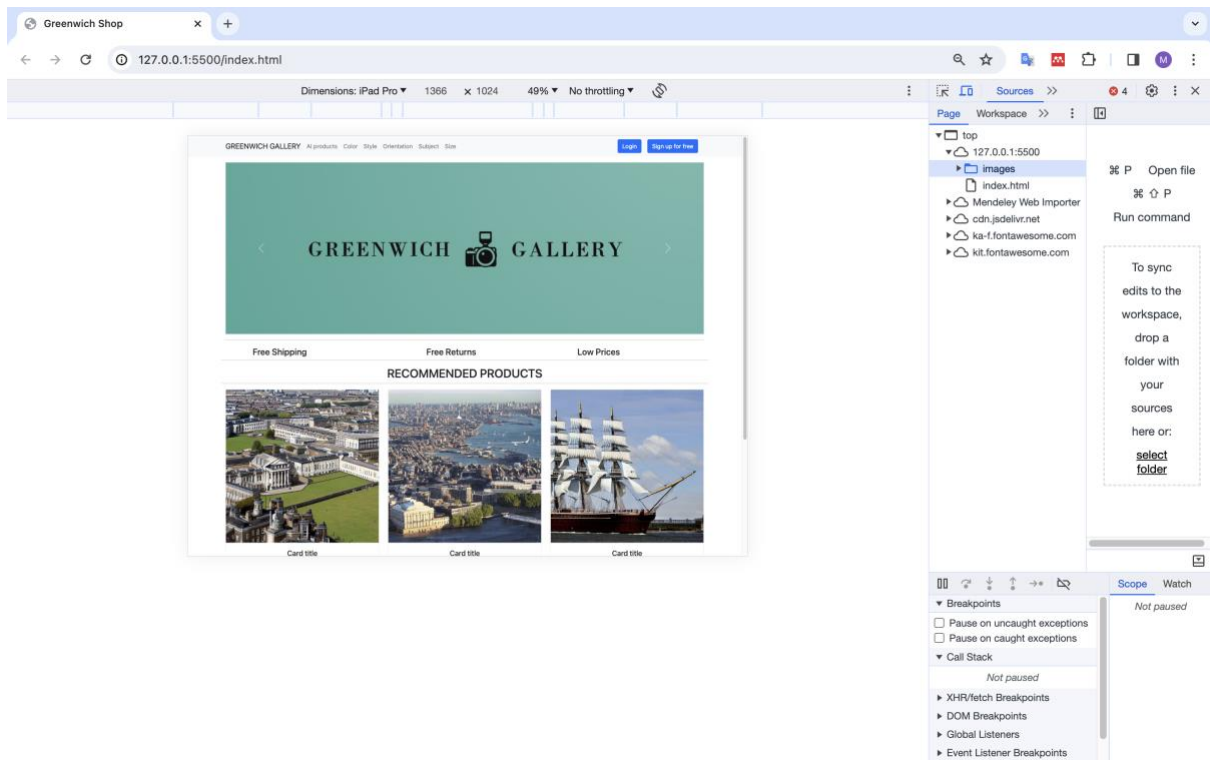


Figure 10.Tablet PC UI

IMPLEMENTATION

In this research, the development interface employed included Visual Studio Code and MySQL Workbench. To address compatibility challenges with the MacBook Air, a database dump file has been included in the project's zip file. The query content is accessible within the dump file.

1. SYSTEM BACK-END IMPLEMENTATION

A. REGISTRATION

On the homepage, the login and registration buttons are located on the right side of the navigation. Users wishing to register can access registration for clicking on “Sign up for free” button.

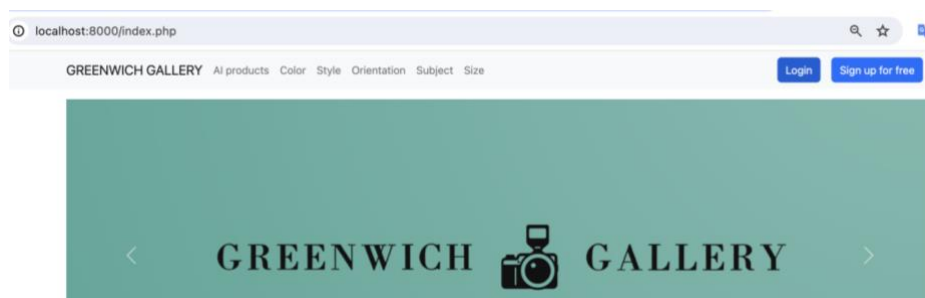


Figure 11. Sign Up Button

Users who want to register on the Greenwich Gallery e-commerce site need to enter their name, surname, email, and password information in the registration form below and check the "terms and conditions" option.

A screenshot of a web browser displaying the registration form on the Greenwich Gallery website. The browser's address bar shows '127.0.0.1:5500/signup.html'. The website's navigation bar includes the text 'GREENWICH GALLERY'. Below the navigation bar is a large white box containing the registration form. The form is titled 'Sign Up' and includes the following fields: 'Name' (with a sub-label 'Enter your first name.'), 'Surname' (with a sub-label 'Enter your surname.'), 'Email' (with a sub-label 'Enter your email'), 'Password' (with a sub-label 'Enter your password.'), and 'Confirm Password' (with a sub-label 'Re-enter your password.'). Below these fields is a checkbox labeled 'I accept all terms and conditions.' and a blue 'Sign Up' button.

Figure 12. Registration Form

The relevant PHP codes for the mentioned functionality can be seen in the "signup.php" file below.

```

1 <?php
2 include 'connection.php';
3 $email=$_POST['email'];
4 $name=$_POST['name'];
5 $surname=$_POST['surname'];
6 $password=$_POST['password'];
7 $repassword=$_POST['repassword'];
8 if($password!=$repassword){
9     $failure = "password does not match ";
10    exit($failure);
11    //header("Location: signup.html");
12 }else{
13     $pepper = "clisvFdxHDdeJ0lvxpecFv";
14     $pwd = $_POST['password'];
15     $pwd_peppered = hash_hmac("sha256", $pwd, $pepper);
16     $pwd_hashed = password_hash($pwd_peppered, PASSWORD_ARGON2ID);
17
18     $sql="insert into users (create_time,email,name,surname,password,isActive,Roleid) values (now(),'$email','$name','$surname','$pwd_hashed','true','2')";
19     $result = mysqli_query($con,$sql);
20     header("Location: login.html");
21 }
22
23

```

Figure 13.Registration PHP codes

After successfully signing up, a row is created on the users table in the database.

	id	create_time	email	name	surname	password	isActive	Roleid
1	3	0000-00-00 00:00:00	test1@test.com	test	'test'	'\$argon2id\$v=19\$m=655	true	2
2	4	0000-00-00 00:00:00	test@test.com	test	'test'	'\$argon2id\$v=19\$m=655	true	3
3	5	2023-12-06 00:14:24	melek_izmirli@hotmail.com	Melek	İzmirli	'\$argon2id\$v=19\$m=655	true	3

Figure 14.Registration DB

Validations have been carried out by some bootstrap features. This includes controlling the required areas on the registration form and checking the input format.

Sign Up

Email
Enter your Email.

Name
Enter your name. Please fill in this field.

Surname
Enter your surname

Password
Enter your password.

Confirm Password
Re-enter your password.

☐ I accept all terms and conditions.

Sign Up

Figure 15.Validation of the required fields

Sign Up

Email

melek

Name

Enter your name

Surname

Enter your surname

Password

...

Confirm Password

...

☐ I accept all terms and conditions.

Sign Up

Figure 16.Validation of email format

Sign Up

Email

melek@melek.com

Name

melek

Surname

izmirli

Password

...

Confirm Password

...

☐ I accept all terms and conditions.

Please tick this box if you want to proceed.

Figure 17.Validation of checkbox

Additional validations include the verification of password entries through back-end controls, which are executed as outlined below:



Figure 18.Password check

```

products  users  signup.html  signup.php
signup.php
1  <?php
2  //password validation
3  $password=$POST['password'];
4  if($password!=$password){
5      $failure = "password does not match ";
6      exit($failure);
7  }
8  //header("Location: signup.html");
9  }else{
10     $pepper = "clisvFdMddj0lvxpecFw";
11     $pwd = $_POST['password'];
12     $pwd_peppered = hash_hmac("sha256", $pwd, $pepper);
13     $pwd_hashed = password_hash($pwd_peppered, PASSWORD_ARGON2ID);
14
15     $sql="insert into users (create_time,email,name,surname,password,isActive,RoleId) values (now(),'$email','$name','$sur";
16     $result = mysql_query($sql,$con);
17     header("Location: login.html");
18 }
19
20
21
22
23
    
```

Figure 16.PHP Backend validation and signup

User passwords are stored in encrypted format in the database. This encryption is facilitated through the following PHP codes.

The screenshot shows a database query result for the 'users' table. The query is 'SELECT * FROM users LIMIT 100'. The results are displayed in a table with columns: create_time, email, name, surname, password, isActive, and Roleid. The passwords are encrypted using a hashing function.

create_time	email	name	surname	password	isActive	Roleid
0000-00-00 00:00:00	test1@test.com	test	tesQ	'\$argon2id\$v=19\$m=65536,t=4,p=1\$aXJYUHZDa112	true	2
0000-00-00 00:00:00	test@test.com	test	'test'	'\$argon2id\$v=19\$m=65536,t=4,p=1\$c25LWnpWUHQ	true	3
2023-12-06 00:14:24	melek_izmirli@hotmail.com	Melek	izmirli	'\$argon2id\$v=19\$m=65536,t=4,p=1\$R1hxNTYzSEFW	true	3

Figure 19.Encrypted password DB

The screenshot shows a snippet of PHP code for password encryption. It uses a pepper and a hashing function to generate a hashed password.

```

12 }else{
13     $pepper = "c1isvfdxM0dmj0lvxpecFv";
14     $pwd = $_POST['password'];
15     $pwd_peppered = hash_hmac("sha256", $pwd, $pepper);
16     $pwd_hashed = password_hash($pwd_peppered, PASSWORD_ARGON2ID);
17 }

```

Figure 20.Encrypted password PHP

B. SIGN IN

Users who want to log into the system initiate their session by using their username and password.

The screenshot shows the 'Sign In' form in the 'GREENWICH GALLERY' application. The form has two input fields: 'Email' and 'Password'. Below the fields is a 'Sign In' button.

Figure 21.Sign In Form

The screenshot shows the 'login.php' file in a code editor. The code handles the login process, including session management, password verification, and redirection to the index page.

```

1 <?php
2 include 'connection.php';
3 session_start();
4 $email = $_POST['email'];
5 $password = $_POST['password'];
6 $sql = "select * from users where email='$email' ";
7 $result = mysqli_query($con, $sql);
8 if ($result) {
9     $row = mysqli_fetch_array($result, MYSQLI_ASSOC);
10     $count = mysqli_num_rows($result);
11     if ($count == 0) {
12         echo "Error: user not found";
13         exit();
14     }
15     $pepper = "c1isvfdxM0dmj0lvxpecFv";
16     $pwd_peppered = hash_hmac("sha256", $password, $pepper);
17     $pwd_hashed = $row['password'];
18     if (!password_verify($pwd_peppered, $pwd_hashed)) {
19         echo "Password incorrect.";
20         exit();
21     }
22     $active = $row['isActive'];
23     $roleId = $row['Roleid'];
24     $id = $row['id'];
25     if ($count == 1 && $active == 1) {
26         $_SESSION['email'] = $email;
27         $_SESSION['roleId'] = $roleId;
28         $_SESSION['id'] = $id;
29         header("Location: index.php");
30     } else {
31         $error = "Your Login Name or Password is invalid";
32     }
33 } else {
34     echo "Error: " . $sql . "<br>" . mysqli_error($con);
35     exit();
36 }
37 }

```

Figure 22.Sign In PHP

Necessary confirmations and error controls have also been conducted in this functionality of the system.

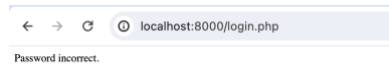


Figure 23. Sign In Error

The crucial point to note here is the authorization of users. If a user is assigned a role ID of 3 in the database, they are authorized as an admin. If, during user registration, the default role assigned is 2 and it hasn't been changed by the admin, the user continues to operate in the system as a basic user. Users with admin authority have managerial roles, such as product entry, visible in the "Products" tab during their sessions.

Additionally, the session is started by the system upon successfully login. Basket icon and username appears following the start of session.

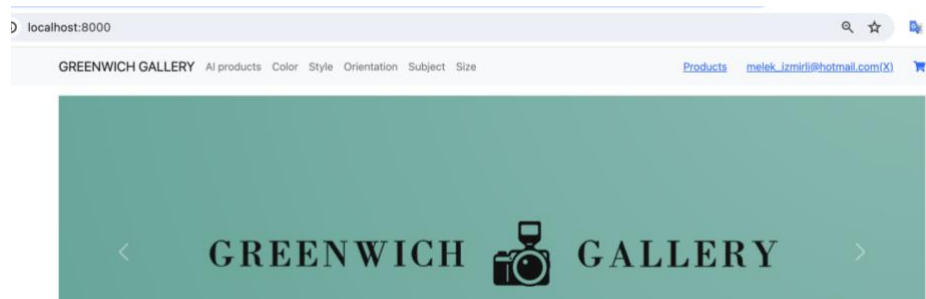


Figure 24. Session

C. PRODUCT ENTRY

Admin users are able to operate some crud roles including adding items into database on "productEntry.html".

Greenwich Shop

127.0.0.1:5500/productEntry.html

GREENWICH GALLERY

Product Entry

Name

Description

Category

Stock

Price

Upload photo

Choose file No file chosen No file chosen

Save

Figure 25.Product Entry Form

Products are saved into products table of the database through the product entry form. The main screen displays and allows filtering of the products.

productEntry.php products

SELECT * FROM products LIMIT 100

Search results

Cost: 12ms < 1 > Total 11

	id	name	description	categoryid	stock	price	photo
1	31	'test'	'test'	1	3	21321.0000	X'ffd8ffe000104a464946
2	32	'test'	'test'	1	3	21321.0000	X'ffd8ffe000104a464946
3	33	'test'	'test'	1	3	21321.0000	X'ffd8ffe000104a464946
4	34	'test'	'test'	2	3	21321.0000	X'ffd8ffe000104a464946
5	35	'test'	'test'	2	3	21321.0000	X'ffd8ffe000104a464946
6	36	'test'	'test'	3	3	21321.0000	X'ffd8ffe000104a464946

Figure 26.Product Entry DB

D. BASKET

Users can add items into basket, delete items from the basket and the system calculate total amount according to the change made. Additionally, a mock-up “pay” button has been added at the bottom of the page that can clear basket and total when clicked.

localhost:8000/Basket.php

GREENWICH GALLERY

#	Name	Price	Actions
1	'test'	21321.0000	Remove(x)
2	'test'	21321.0000	Remove(x)
		Total : 42642	Pay

Figure 27.Basket Form

```

<th scope="col">Name</th>
<th scope="col">Price</th>
<th scope="col">Actions</th>
</tr>
</thead>
<tbody>
<?php
    $say=0;
    $total=0;
    $result = $con->query("select b.id,p.name,p.price from Basket b
    INNER JOIN users u on u.id=b.userId
    INNER JOIN products p on p.id=b.productId
    where u.id=$id");
    while ($row = $result->fetch_assoc()) {
        $say++;
        echo "<tr>";
        echo "<th scope='row'>". $say."</th>";
        echo "<td>". $row['name']. "</td>";
        echo "<td>". $row['price']. "</td>";
        echo "<td>";
        echo "<a href='basket_item_remove.php?id=".$row['id']."'>Remove(x)</a> ";
        echo "</td>";
        echo "</tr>";
        $total+=$row['price'];
    }
?>
<tr>
<th scope='row'></th>
<td></td>
<td>Total : <?php echo $total ?></td>
<td>
<a href='Pay.php' class='btn btn-sm btn-primary' >Pay</a>
</td>
</tr>
</tbody>

```

Figure 28.Basket PHP

```

1 <?php
2 include 'connection.php';
3 session_start();
4 $id = 0;
5 if( isset( $_GET['id'] ) )
6 {
7     $id= $_GET['id'];
8 }
9
10 $sql = "SELECT * FROM products where id=$id";
11 print $sql;
12 $result = mysqli_query($con, $sql);
13 $row = mysqli_fetch_array($result, MYSQLI_ASSOC);
14 $userId=$SESSION['id'];
15 $sql="insert into Basket (productId,userId,create_time) values ($id,$userId,now())";
16 if (mysqli_query($con, $sql)) {
17
18 } else {
19     echo "Error: " . $sql . "<br>" . mysqli_error($con);
20     exit();
21 }
22 header("Location: productdetail.php?id=$id");
23 ?>

```

Figure 29.Adding Item to Basket PHP

```

1 <?php
2 include 'connection.php';
3 session_start();
4 $id = 0;
5 if( isset( $_GET['id'] ) )
6 {
7     $id= $_GET['id'];
8 }
9
10 $sql = "delete from Basket where id=$id";
11 $result = mysqli_query($con, $sql);
12 header("Location: basket.php");
13 ?>

```

Figure 30.Remove Items from Basket PHP

```

products Pay.php x users signup.htm
Pay.php
1 <?php
2 session_start();
3 include 'session.php';
4 include 'connection.php';
5 $id=$_SESSION['id'];
6
7 $sql = "delete from Basket where userId=$id";
8 $result = mysqli_query($con, $sql);
9 header("Location: basket.php");
10 ?>

```

Figure 31.Payment PHP

E. FILTERING

Users who wish to filter products from the website can utilise the categories in the header menu. Upon clicking on any category, related products are listed.

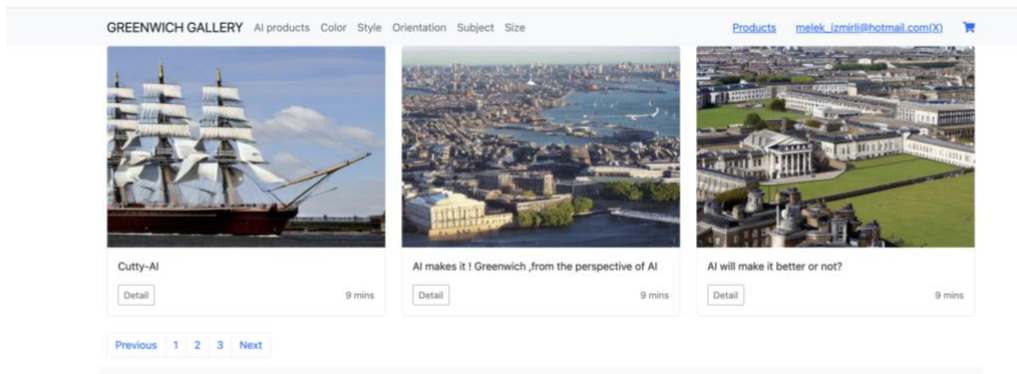


Figure 32.Filtering by Category

The back-end operations related to filtering are executed by the code in “productList.php”.

```

categories products productList.php x
productList.php
1 <?php
2 session_start();
3 include 'connection.php';
4 include 'header.php';
5 $catid=0;
6 if(isset($_GET['catid']))
7 {
8     $catid= $_GET['catid'];
9 }
10 ?>
11 <div class="container">
12
13     <div class="row">
14         <?php
15             $result = $con->query("SELECT * FROM products where categoryId=$catid LIMIT 9");
16             while ($row = $result->fetch_assoc()) {
17                 $id = $row['id'];
18                 $name = $row['name'];
19                 $image=$row['photo'];
20                 echo "<div class='col-md-4'>";
21                 echo "<div class='card mb-4 box-shadow'>";
22                 echo "<img class='card-img-top' src='data:image;base64, ".base64_encode($image)." ' style='height: 300px;";
23                 echo "<div class='card-body'>";
24                 echo "<p class='card-text'>". $row['description']. "</p>";
25                 echo "<div class='d-flex justify-content-between align-items-center'>";
26                 echo "<div class='btn-group'>";
27                 echo "<a href='productdetail.php?id=$id' class='btn btn-sm btn-outline-secondary'>Detail</a>";
28                 echo "</div>";
29                 echo "<small class='text-muted'>9 mins</small>";
30                 echo "</div>";

```

Figure 33.Filtering by Category php codes

TESTING

1. BLACKBOX TESTING

The results obtained from the test conducted according to the test plan are visible in the table below:

Test Plan					
Case	Item	Expected Result	Condition	Actual Result	Pass/Fail
1	Navigation	Check all navigations and menus are presented and accessible		Navigations work properly.	PASS
2	Navigation	Links lead to correct destination.		Links and navigations lead correct targets.	PASS
3	Navigation	Responsiveness of navigation on different devices		Navigations work properly on different interfaces, different size of devices.	PASS
	Registration	System allows visitors to sign up.	Open registration form	Sign-up is accessible for new users.	PASS
4	Registration	Check registration with valid inputs	Validation pass	System registers user.	PASS
5	Registration	Check registration with incorrect attempts		System gives error messages	PASS
6	Registration	Check password encryption		System stores passwords encrypted into database.	PASS
	Registration	User information should be stored in database		Users information stored in DB	PASS
7	Login	Verify successful login with valid credentials		System initiates session and leads to homepage.	PASS
	Login	Check error handling for incorrect entries.		System gives error messages regarding faulty entries.	PASS
	Search	Search should be available.	Upon clicking on	Search results appear.	PASS

			any category		
	Search	Verify search results for relevant items.	Upon clicking on any category	Items visible on the page	PASS
	Pagination	Search results should be paginated.	Upon clicking on any category	Works properly	PASS
	Browse	User should see the details of products.	Click on details button	Details page opens	PASS
	Basket	Adding items into basket		Items appear on basket page	PASS
		Updating basket	Adding item /removing item	Items are added, items are removed on the basket page	PASS
		Calculate total	Adding item(s)	Total appears on the basket page	PASS
		purchase	Clicking on Buy button	Clears basket	PASS
		Clear basket	Delete all items, buy items	All items are removed on the basket page	PASS
	Log out		Log in	Terminates session.	PASS

2. USABILITY TESTING

Usability testing is an approach employed to assess the effectiveness of a product or service design for its target users, whether it be a portal, software application, or mobile app (Nielsen, 1993). This involves observing a representative group of individuals as they interact with the product or service and analyzing their actions to identify any issues or possible enhancements (Rubin & Chisnell, 2008).

Among several testing methods for interface design, Nielsen heuristics is an effective approach for measuring design flaws that impact user experience (YáñezGómez Rosa et al., 2017). Following Nielsen's heuristics, the aesthetic and design aspects of the interface are examined to enhance flexibility and recall features (Szabo, 2017).

Nielsen's 10 heuristics (Nielsen, 1995) constitute a set of design principles formulated by Jacob Nielsen to assess the usability of interactive systems. The 10 heuristics, as outlined by Nielsen and NNGROUP, are as follows:

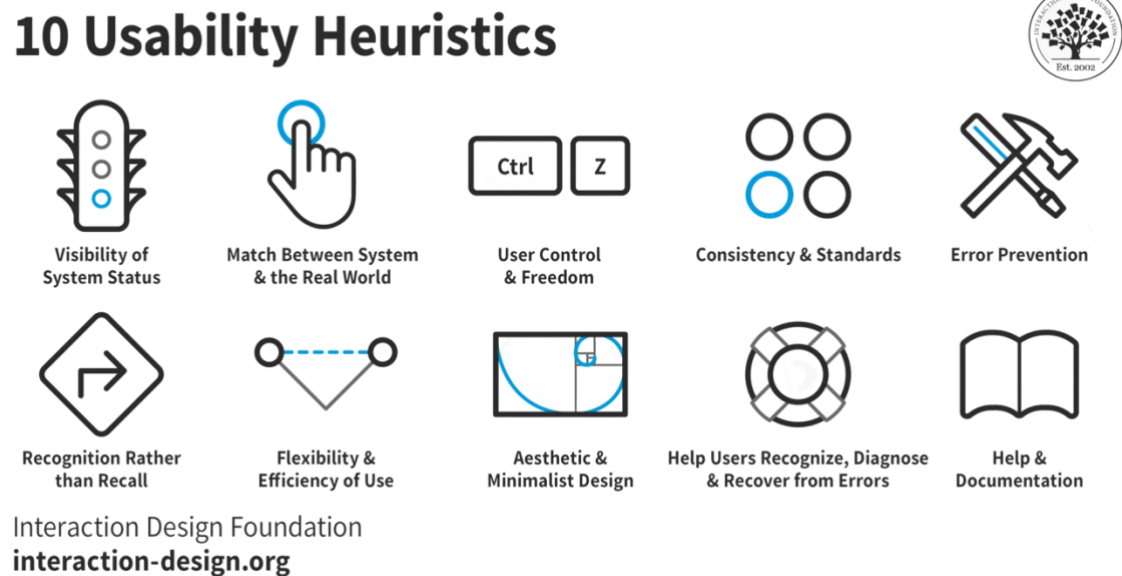


Figure 34. Interaction Design Foundation - IxDF, 2016

The Greenwich Gallery e-commerce website has been crafted in accordance with Nielsen's heuristics to ensure optimal usability. It prioritizes high visibility providing clear feedback on the system status. The website facilitates efficient navigation through well-defined categories, a basket, and a menu bar. Additionally, the system employs language and terminology familiar to e-commerce users for better understanding. Consistency in design and functionality is maintained across all pages and items. Error prevention is a priority, demanding informative error messages and user-friendly validation mechanisms for forms and inputs. Users should have the freedom to explore and modify their choices with easy navigation options. The aesthetics, layout, and visual hierarchy are designed to align with user expectations, contributing to an aesthetically pleasing and user-friendly interface. Lastly, the site ensures the provision of relevant information.

CONCLUSION

This project includes a user-centred e-commerce website. Making user-focused designs involves putting the user's needs at the centre and even requiring the designer to put themselves in the user's shoes. It means designing products in a way that prioritizes the user's experience, with the designer considering how the user will interact with the designed product.

This project has taught many lessons. Firstly, designing a product that caters to the needs of multiple users when creating a single item is not as easy as it may seem at first glance. Secondly, it can be said that there is a broad toolkit in this field considering the diversity and number of test and requirement elicitation techniques. Lastly, embracing SDLC principles serves as a guiding force for a well-organized and effective development process.

A prospective study could focus on the integration of AI tools for image generation within the Greenwich Gallery website. A majority of online content providers have proactively adopted AI tools to assist users in crafting personalized images. This involves considering how AI-driven image generation can be implemented to enhance user experiences and creative processes on the platform.

The addition of functionalities such as a chatbot for assistance, revealed during the persona study, could be also considered for future work.

REFERENCES

- Barrett, A.R. and Edwards, J.S., 1995. Knowledge elicitation and knowledge representation in a large domain with multiple experts. *Expert Systems with Applications*, 8(1), pp.169-176.
- Blomkvist, S., 2002. The user as a personality-using personas as a tool for design. KTH-Royal Institute of Technology, Stockholm Www. Nada. Kth. Se/tessy/Blomkvist. Pdf, 980.
- Cooper, A., Reimann, R., Cronin, D. and Noessel, C., 2014. About face: the essentials of interaction design. John Wiley & Sons. Available from: ProQuest Ebook Central. [01 December 2023].
- Hussain, Z. M., & Sumari, P. (2016). WERT technique in requirements elicitation for web applications. In 2016 International Conference on Electronics, Information, and Communications (ICEIC), Danang, Vietnam (pp. 1-4). doi:10.1109/ELINFOCOM.2016.7562976.
- Maiden, N.A. and Rugg, G., 1996. ACRE: selecting methods for requirements acquisition. *Software Engineering Journal*, 11(3), pp.183-192.
- Nielsen, J. (1993). *Usability engineering*. Academic Press.
- Nielsen, Jakob. "How to conduct a heuristic evaluation." Nielsen Norman Group 1.1 (1995): 8.
- Nielsen's 10 heuristics Interaction Design Foundation - IxDF. (2016, May 25). What is Heuristic Evaluation?. Interaction Design Foundation - IxDF. <https://www.interaction-design.org/literature/topics/heuristic-evaluation>.
- Nora, K., Gefei, Z., & Mar, A. J. (2006). "ESCALONA Model transformations from requirements to web system design." In Proceedings of the 6th international conference on Web engineering.

Rubin, R. S., & Chisnell, D. (2008). Handbook of usability testing: how to plan, design, and conduct effective tests. John Wiley & Sons.

Rosa Yáñez Gómez,¹Daniel Cascado Caballero,¹and José-Luis Sevillano Heuristic Evaluation on Mobile Interfaces: A New Checklist <https://doi.org/10.1155/2014/434326>.

Rugg, G. and McGeorge, P., 1997. The sorting techniques: a tutorial paper on card sorts, picture sorts and item sorts. Expert Systems, 14(2), pp.80-93.

Rugg, G., Corbridge, C., Major, N.P., Burton, A.M. and Shadbolt, N.R., 1992. A comparison of sorting techniques in knowledge acquisition. Knowledge Acquisition, 4(3), pp.279-291.

Szabo, P. W., 2017. User experience mapping: get closer to your users and create better products for them. Packt Publishing.

Tiwari, S., Rathore, S. S., & Gupta, A. (2012). Selecting requirement elicitation techniques for software projects. In 2012 CSI Sixth International Conference on Software Engineering (CONSEG) (pp. 1-10). Indore, India. doi: 10.1109/CONSEG.2012.6349486.

UXtweak. (2023). UXtweak. Available at: <https://study.uxtweak.com/cardsort/VAN8Fe6ofleNJ0v7n0lFy> (Accessed: 30/11/2023).

Zimmerman, D.E. and Akerelrea, C., 2002, September. A group card sorting methodology for developing informational web sites. In Proceedings. IEEE International Professional Communication Conference (pp. 437-445). IEEE.