



A project report on  
**SHOPFORHOME**

Submitted in partial fulfilment of the requirements for the internship of  
Wipro

**JAVA & AWS C4 GROUP-3**

By

**Bhargavi Kunta  
Manas Hassija  
Kaza Geetanjali  
Sake Bhavana  
M Annie Merlin**

under the guidance of

**MR. NEERAJ KUMAR KHERIA**

**Great Learning**

**June 2022**

## ABSTRACT

ShopForHome is a popular Store in the market for shopping the home décor stuff . Due to Covid 19 all the offline shopping stopped. So, the store wants to move to the cloud platforms and wants their own web application. There are 2 users on the application:- Users and Admin.

Users can register, login, buy products, add products to their cart or Wishlist, view and use coupons, and also search for the products which they want to buy.

Admin can register, login, perform CRUD (Create, Read, Update, Delete) on products, users, coupons. Admin can also view the list of products and also upload products in bulk as csv file. They can also view sales report data which shows the quantity of products sold on the website, and they can also view the stock report data which can show the quantity of products which are available.

## **ACKNOWLEDGEMENTS**

We are profoundly grateful to MR.NEERAJ KUMAR KHERIA expert guidance and encouragement throughout the project. We would like to extend our sincere thanks to Wipro and great learning for providing us with this opportunity. We are highly indebted to our guide for his guidance and constant supervision as well as for providing necessary information regarding the project and also for his support in completing the project. The work is a team effort minus which the completion of this project was not possible.

**Bhargavi Kunta  
Manas Hassija  
Kaza Geetanjali  
Sake Bhavana  
M Annie Merlin**

## TABLE OF CONTENTS

Abstract	:	2
Acknowledgment	:	3
Table of Contents	:	4
CHAPTER 1: INTRODUCTION	:	5
CHAPTER 2: PROJECT OBJECTIVE	:	5
CHAPTER3: STUDY OF THE SYSTEM	:	5
CHAPTER 4: PROJECT DETAIL	:	6
CHAPTER 5: RESULTS	:	7
CHAPTER 6: CONCLUSION	:	20

## **1.0 INTRODUCTION:**

ShopForHome is for shopping the home decor stuff . Due to Covid 19 all the offline shopping stopped. So, the store wants to move to the cloud platforms and wants their own web application. The overall idea behind this project is to allow users to buy home decor products online.

The application aims to provide a shopping system for users where the Users can register, login, buy products, add products to their cart or Wishlist, view and use coupons, and also search for the products which they want to buy.

Admin can register, login, perform CRUD (Create, Read, Update, Delete) on products, users, coupons. Admin can also view the list of products and also upload products in bulk as csv file. They can also view sales report data which shows the quantity of products sold on the website, and they can also view the stock report data which can show the quantity of products which are available.

## **2.0 PROJECT OBJECTIVE:**

ShopForHome is a popular Store in the market for shopping home decor stuff. Due to Covid 19, all offline shopping stopped. So, the store wants to move to the cloud platforms and wants its own web application.

## **3.0 STUDY OF THE SYSTEM:**

The major objective of this ShopForHome application was to provide a platform for users to buy home décor products from the website. The application consists of two types of users –

- ADMIN – An admin can login, logout and register into the application. On successful login, admin can see the dashboard. Admin is able to perform CRUD on Users. Admin is able to perform CRUD on the products. Admin is able to get bulk upload option to upload a csv for product details. Admin is able to get the stocks. Admin is able to mail if any stock is less than 10. Admin is able to get the sales report of a specific duration. Admin is able to set the discount coupons for a specific set of users.
- USER – Users can register themselves on the platform by signup from the registration page. For registration, the user will need to provide the username, email address, password, phone number, and address. The user is able to log in, Logout, and Register into the application. User is able to see the products in different categories. The user is able to sort the products. The user is able to add the products to the shopping cart. The user is able to increase or decrease the quantity added to the cart. The user is able to add “n” number of products in the cart. The user is able to get the Wishlist option where he can add those products which he wants but don’t want to order now. Users can get different discount coupons.

## **4.0 Project Detail:**

ShopForHome is a popular Store in the market for shopping the home décor stuff . Due to Covid 19 all the offline shopping stopped. So, the store wants to move to the cloud platforms and wants their own web application.

Our project uses the following technologies:

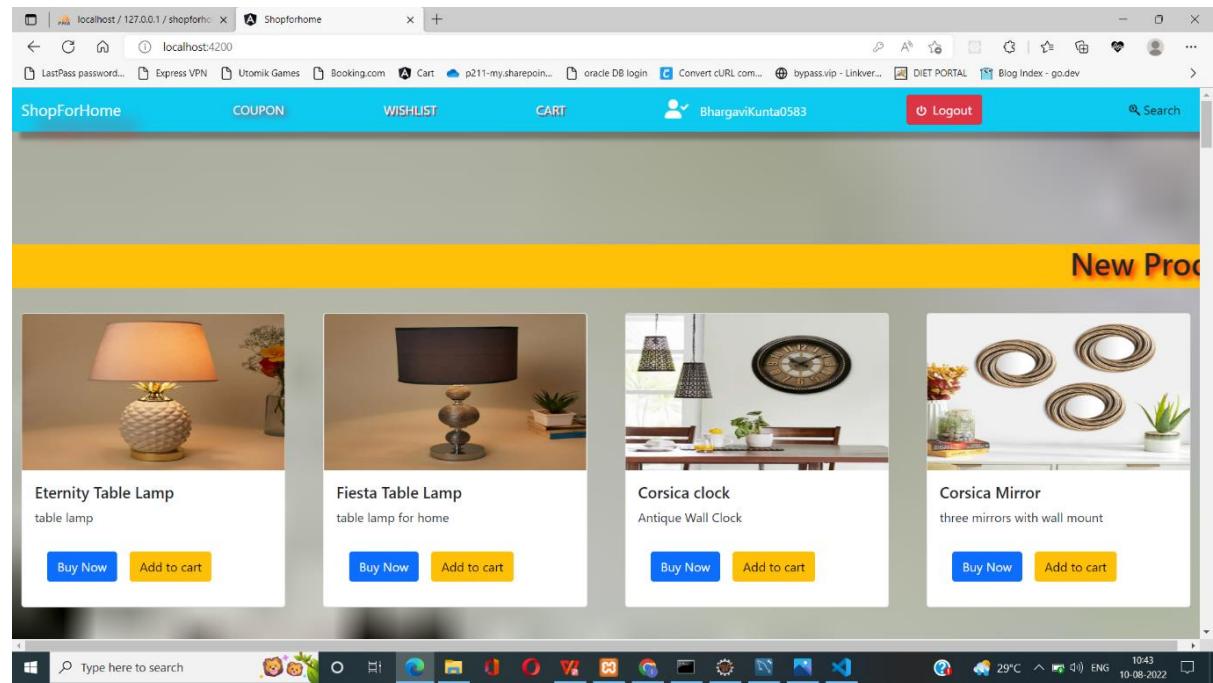
- Angular
- Spring Boot
- SQL
- AWS

## **Softwares Used:**

- Eclipse IDE
- Visual Studio code
- MySQL
- AWS

## 7.0 RESULTS

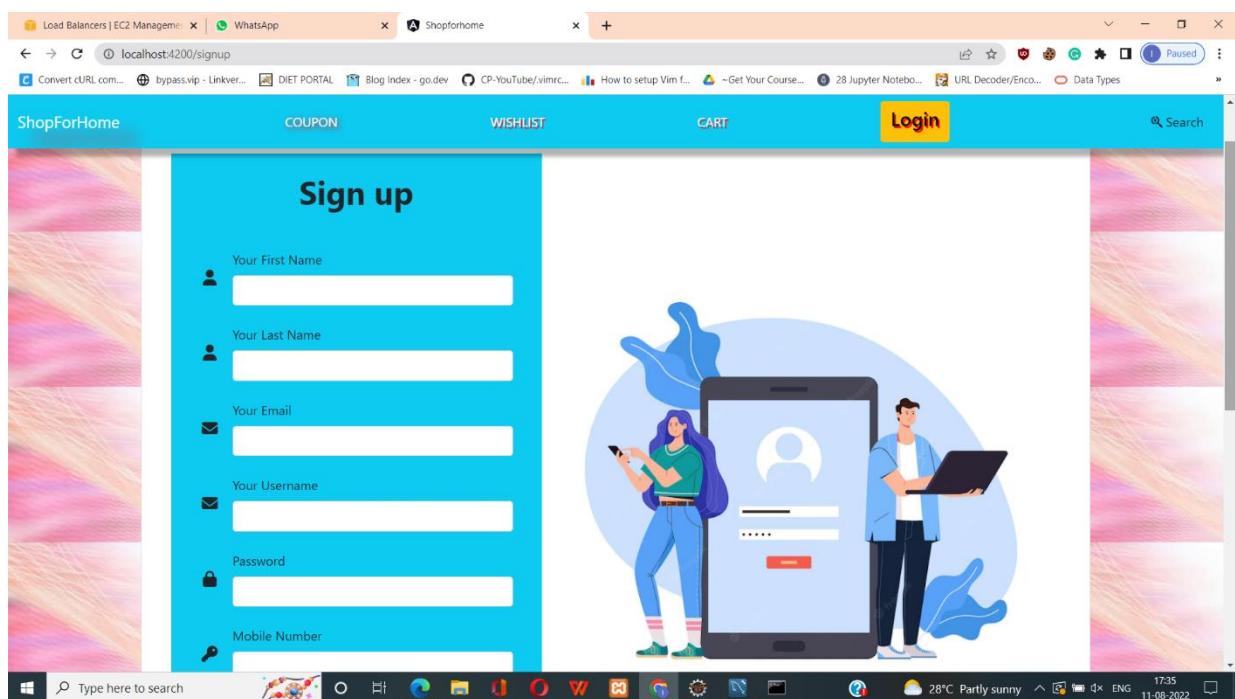
### HOME PAGE:



The Home Screen is the main screen where all products are shown

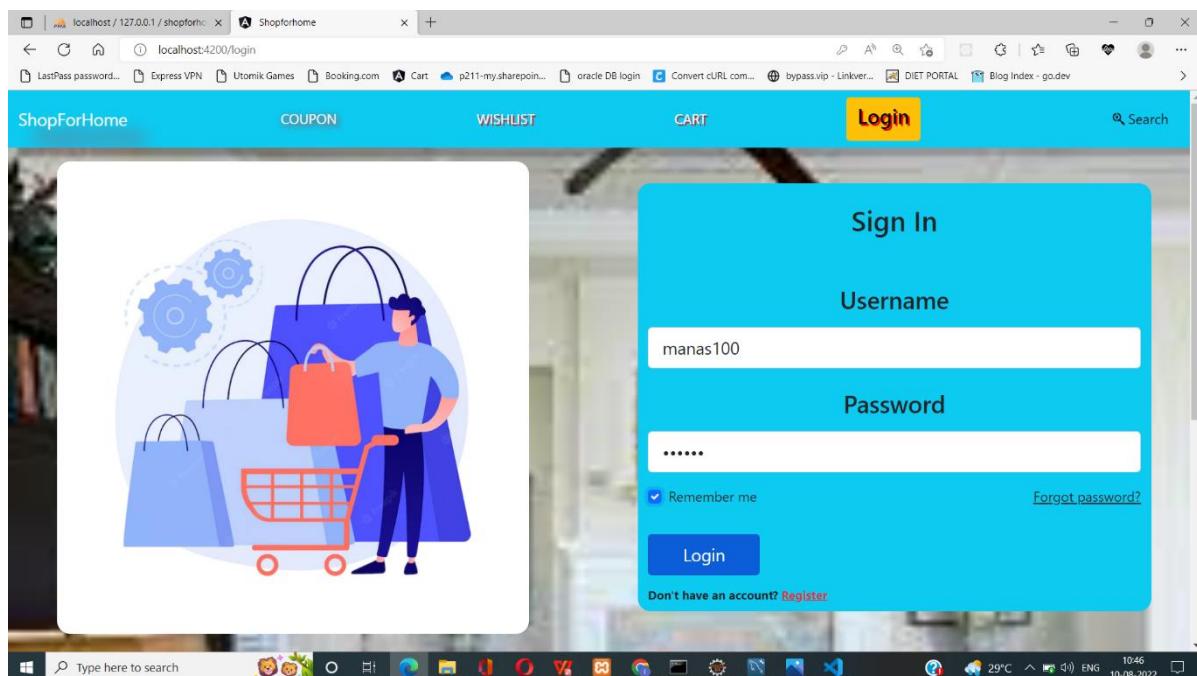
### SIGNUP PAGE:

User and admin are able to register into the Application



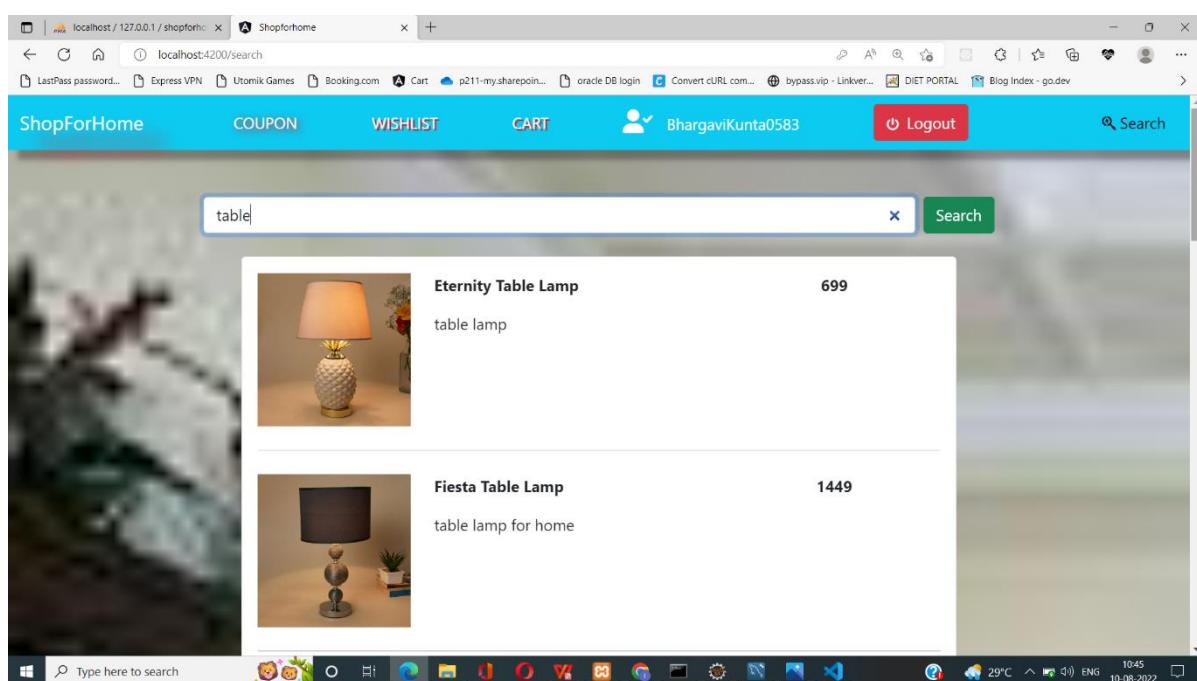
## LOGIN PAGE:

User and admin are able to login and logout from the website.



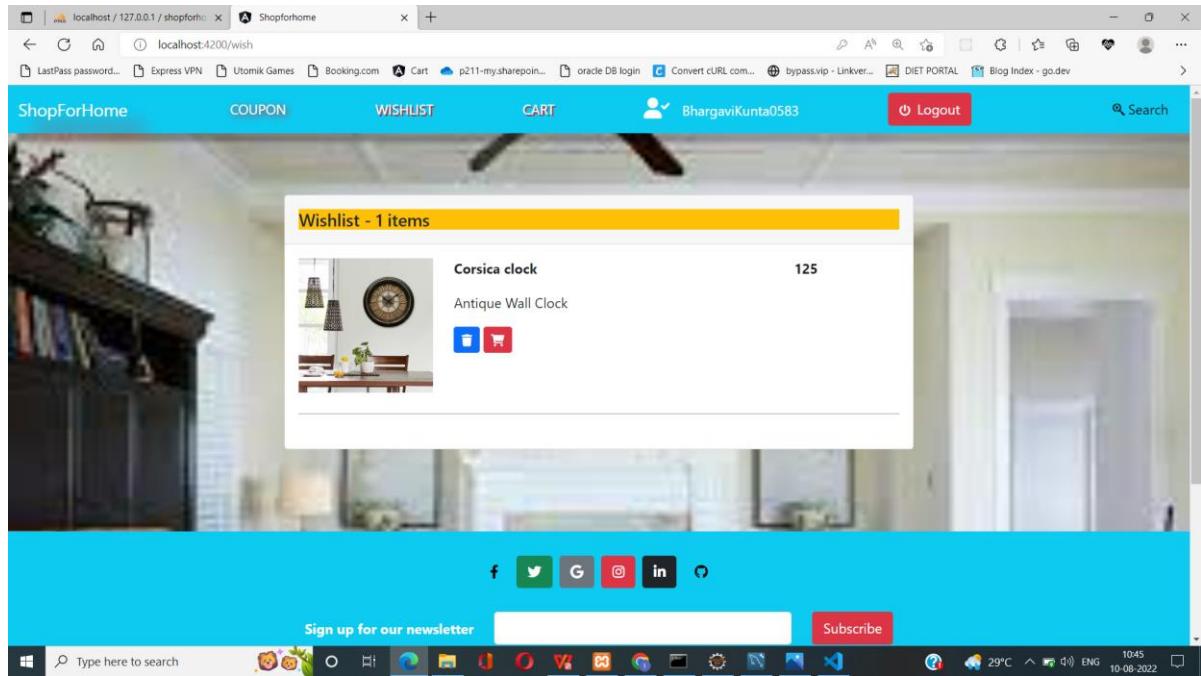
## SEARCH:

Users can search for different product items

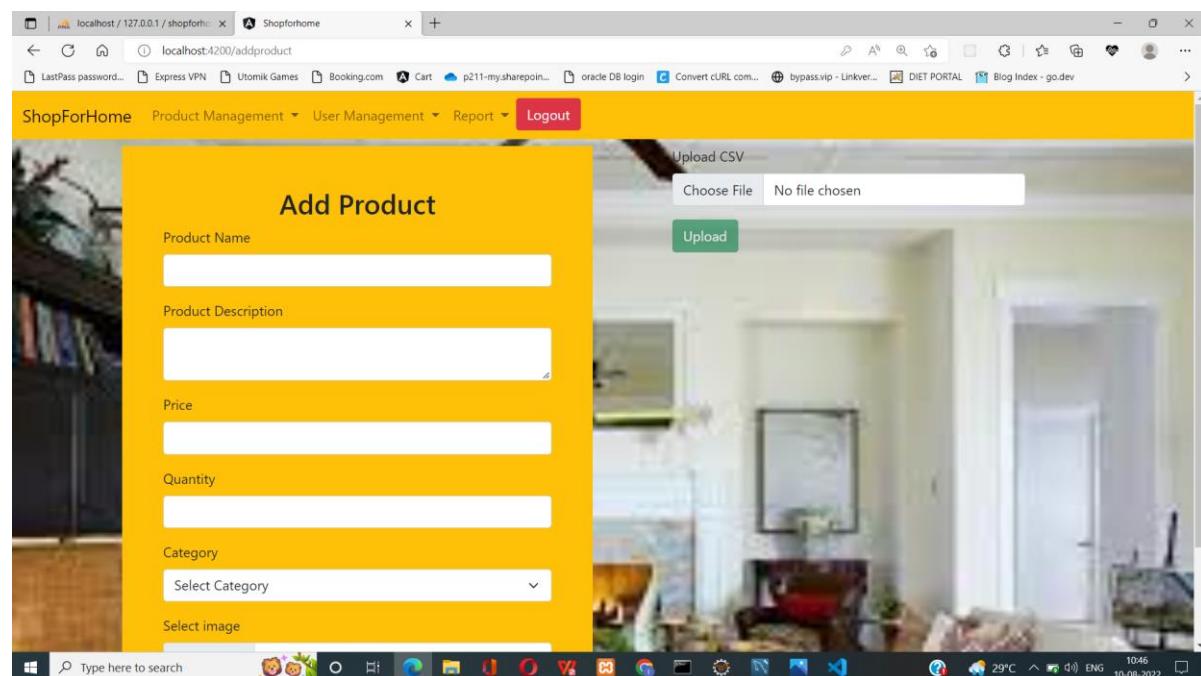


## WISHLIST:

Users are able to add, view and remove items in the Wishlist.



## BULK UPLOAD:



	A1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	name	description	quantity	price	categoryid	discountid	imgurl													
2	Fiesta Bell	Bell for horn	5	500	1	0	assets/img/bell1.jpg													
3	Budhha	Budhha figu	7	799	1	0	assets/img/budha.jpg													
4	Eternity Tat	table lamp	10	699	1	0	assets/img/lamp1.jpg													
5	Fiesta Table	table lamp f	2	1449	1	0	assets/img/lamp2.jpg													
6	Corsica clc	Antique Wa	6	125	1	0	assets/img/clock1.jpg													
7	Corsica Mirr	three miroir	3	399	2	0	assets/img/mirror1.jpg													
8	Valencia pla	Gold Solid N	20	1899	2	0	assets/img/plant1.jpg													
9	Galaxy Glas	Candle hold	5	399	2	0	assets/img/candle1.jpg													
10	Mayra Votiv	votive holde	17	799	2	0	assets/img/holder1.jpg													
11	Taper Candi	Candles for	23	250	2	0	assets/img/taper1.jpg													
12	Adrian Oil	B oil burner	11	899	3	0	assets/img/burner1.jpg													
13	Austin leaf	F leaf platter	10	999	5	0	assets/img/plate1.jpg													
14	Contour tra	Decorative'	8	799	5	0	assets/img/tray2.jpg													
15	Light Holder	Corsica Gol	9	199	5	0	assets/img/holder2.jpg													
16	Pooja bowl	Serenity Gol	3	2499	5	0	assets/img/bowl.jpg													
17	Monk Figuri	Corsica Han	7	399	5	0	assets/img/monk1.jpg													
18	Peacock Fig	Corsica Mys	12	1519	4	0	assets/img/peacock1.jpg													
19	Table Accer	Eternity Ete	15	1999	4	0	assets/img/accent.jpg													
20	Votive Hold	Peacock Shi	10	949	4	0	assets/img/votive2.jpg													
21																				
22																				
23																				
24																				
25																				
26																				
27																				
28																				
29																				
30																				

## DISCOUNT COUPONS:

User is able to get different discount coupons.

Your Promo Code 21 Type : Percentage Value : 2 Detail: table Validity: 0	Your Promo Code 23 Type : Flat Value : 5 Detail: chair Validity: 0
---	---

The screenshot shows a web browser window for the 'ShopForHome' website. The URL is 'localhost / 127.0.0.1 / shopforhome'. The page title is 'Shopforhome'. The top navigation bar includes links for 'COUPON', 'WISHLIST', 'CART', and a user profile with the name 'BhargaviKunta0583'. A red 'Logout' button is also present. The main content area displays a 'Cart - 6 items' section with three products:

- Corsica clock**: Antique Wall Clock. Quantity: 3. Price: 125.
- Fiesta Table Lamp**: table lamp for home. Quantity: 2. Price: 1449.
- Eternity Table Lamp**: table lamp. Quantity: 1. Price: 699.

Each product item has a small image, a product name, a description, a quantity input field with plus and minus buttons, and a price. Below the cart summary, there is a 'Summary' section showing:

- Products: 3892.56
- Total amount (including Tax): 3892.56

A coupon input field contains '21', and a blue 'Go to checkout' button is visible. The bottom of the screen shows a Windows taskbar with various pinned icons and the system tray indicating the date and time as 10:44 on 10-08-2022.

## CART OPTION:

User is able to add the products into the shopping , to increase or decrease the quantity added in the cart and to add “n” number of products into the cart.

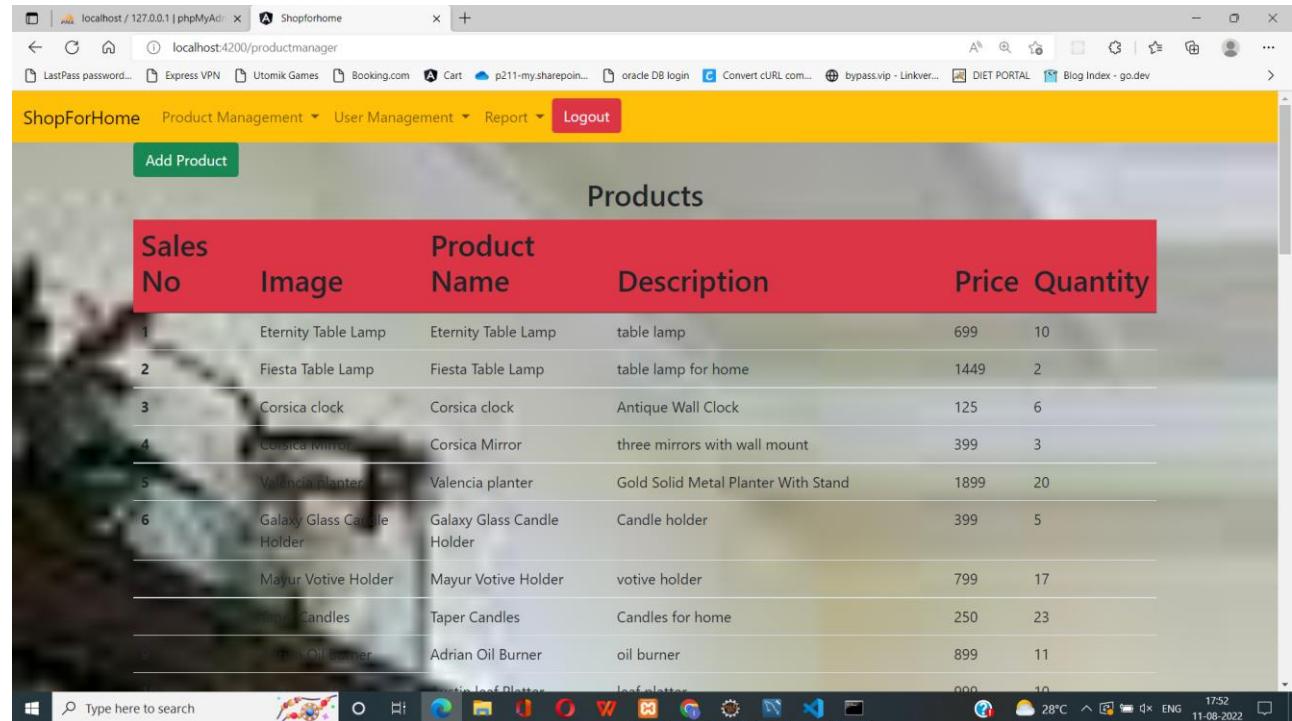
This screenshot shows the same website interface as the previous one, but with updated cart contents. The total number of items in the cart is now 6, and the total amount is 3972. The 'Corsica clock' quantity is now 3, and the 'Fiesta Table Lamp' quantity is now 2. The 'Eternity Table Lamp' quantity remains at 1. The 'Summary' section shows:

- Products: 3972
- Total amount (including Tax): 3972

A coupon input field is empty, and a blue 'Apply' button is visible next to it. The rest of the page, including the product list and the 'Go to checkout' button, is identical to the first screenshot.

## STOCK AND SALES REPORT (ADMIN):

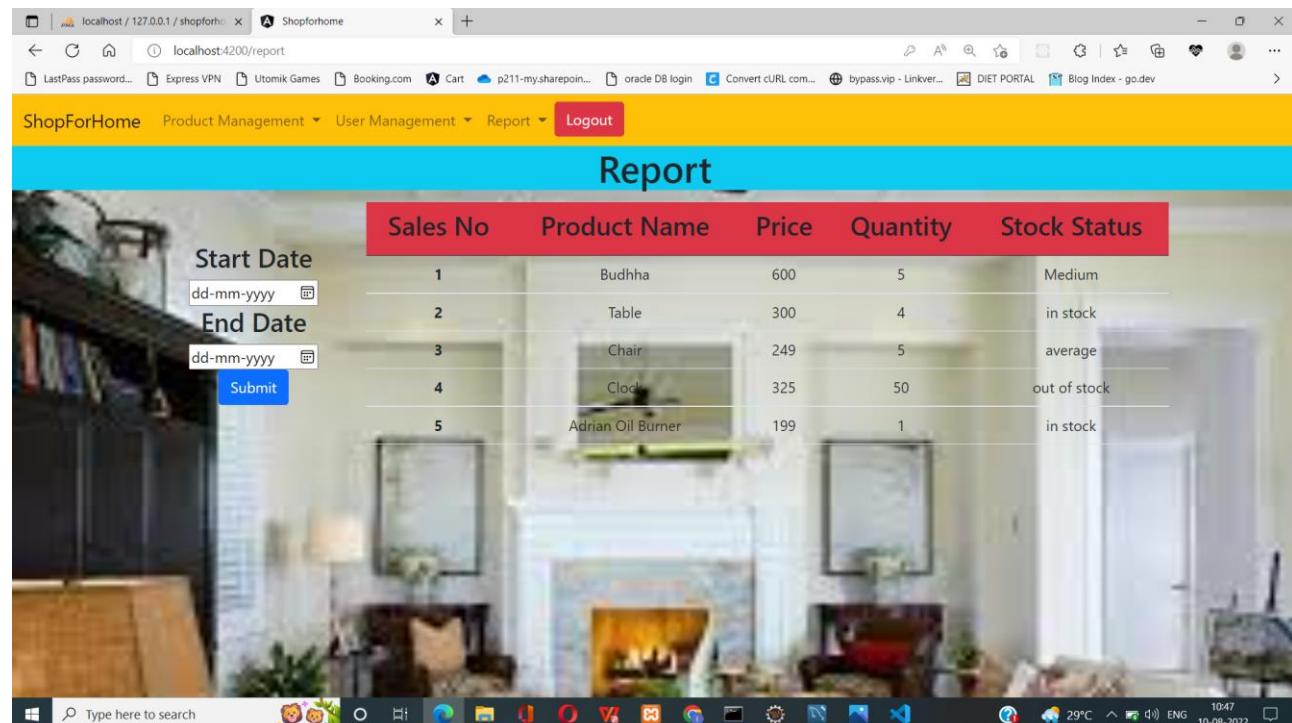
Admin is able to get the sales report of a specific duration.



A screenshot of a web browser window titled "localhost / 127.0.0.1 | phpMyAdmin" showing a sales report. The page has a yellow header with "ShopForHome", "Product Management", "User Management", "Report", and "Logout". Below the header is a green button labeled "Add Product". The main content is a table titled "Products" with columns: Sales No, Image, Product Name, Description, Price, and Quantity. The table contains 10 rows of product information. The browser's taskbar at the bottom shows various open tabs and system icons.

Sales No	Image	Product Name	Description	Price	Quantity
1	Eternity Table Lamp	Eternity Table Lamp	table lamp	699	10
2	Fiesta Table Lamp	Fiesta Table Lamp	table lamp for home	1449	2
3	Corsica clock	Corsica clock	Antique Wall Clock	125	6
4	Corsica Mirror	Corsica Mirror	three mirrors with wall mount	399	3
5	Valencia planter	Valencia planter	Gold Solid Metal Planter With Stand	1899	20
6	Galaxy Glass Candle Holder	Galaxy Glass Candle Holder	Candle holder	399	5
7	Mayur Votive Holder	Mayur Votive Holder	votive holder	799	17
8	Taper Candles	Taper Candles	Candles for home	250	23
9	Adrian Oil Burner	Adrian Oil Burner	oil burner	899	11
10	Leaf Platter	Leaf Platter	leaf platter	000	10

Admin is able to get the stock report.

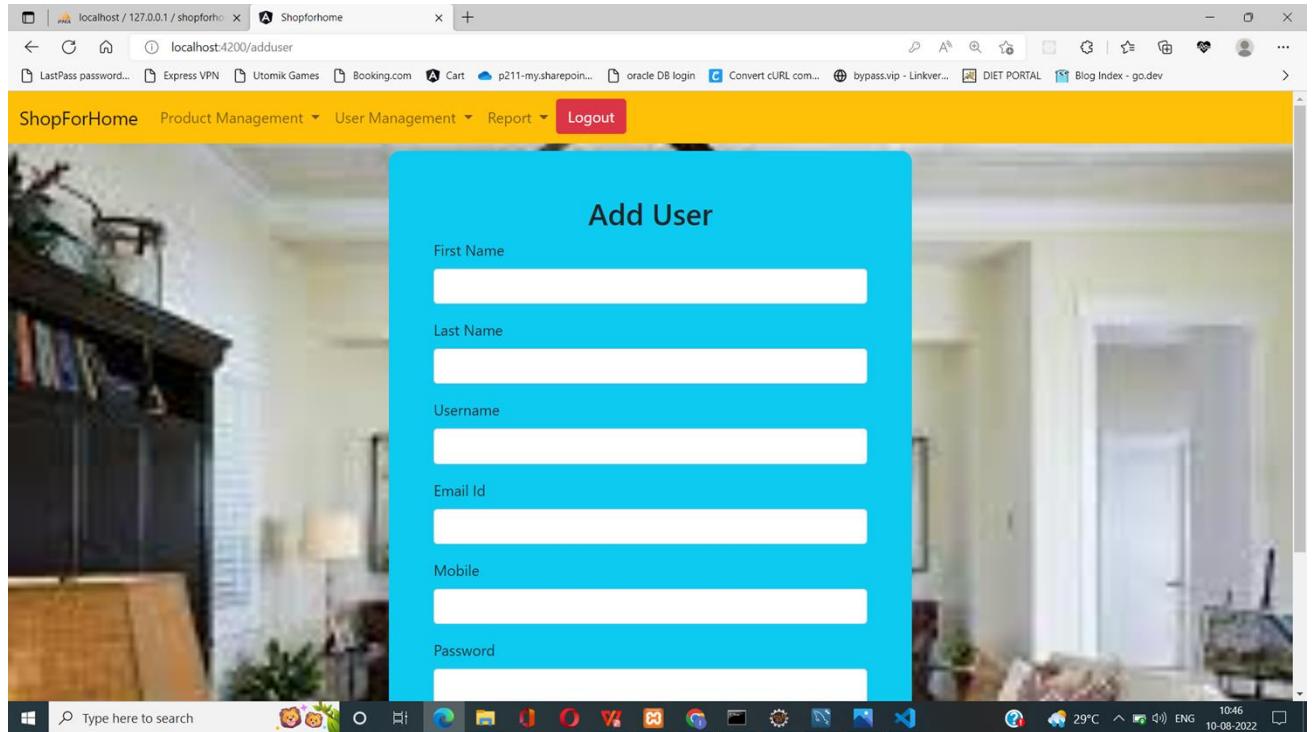


A screenshot of a web browser window titled "localhost / 127.0.0.1 | shopforhome" showing a stock report. The page has a yellow header with "ShopForHome", "Product Management", "User Management", "Report", and "Logout". Below the header is a blue bar with the word "Report". On the left side, there are two input fields: "Start Date" and "End Date", both with "dd-mm-yyyy" placeholder text and calendar icons. A blue "Submit" button is positioned below the "End Date" field. The main content is a table with columns: Sales No, Product Name, Price, Quantity, and Stock Status. The table contains 5 rows of product information. The browser's taskbar at the bottom shows various open tabs and system icons.

Sales No	Product Name	Price	Quantity	Stock Status
1	Budhha	600	5	Medium
2	Table	300	4	in stock
3	Chair	249	5	average
4	Clock	325	50	out of stock
5	Adrian Oil Burner	199	1	in stock

## CRUD ON USERS (ADMIN):

Admin can perform crud on users.

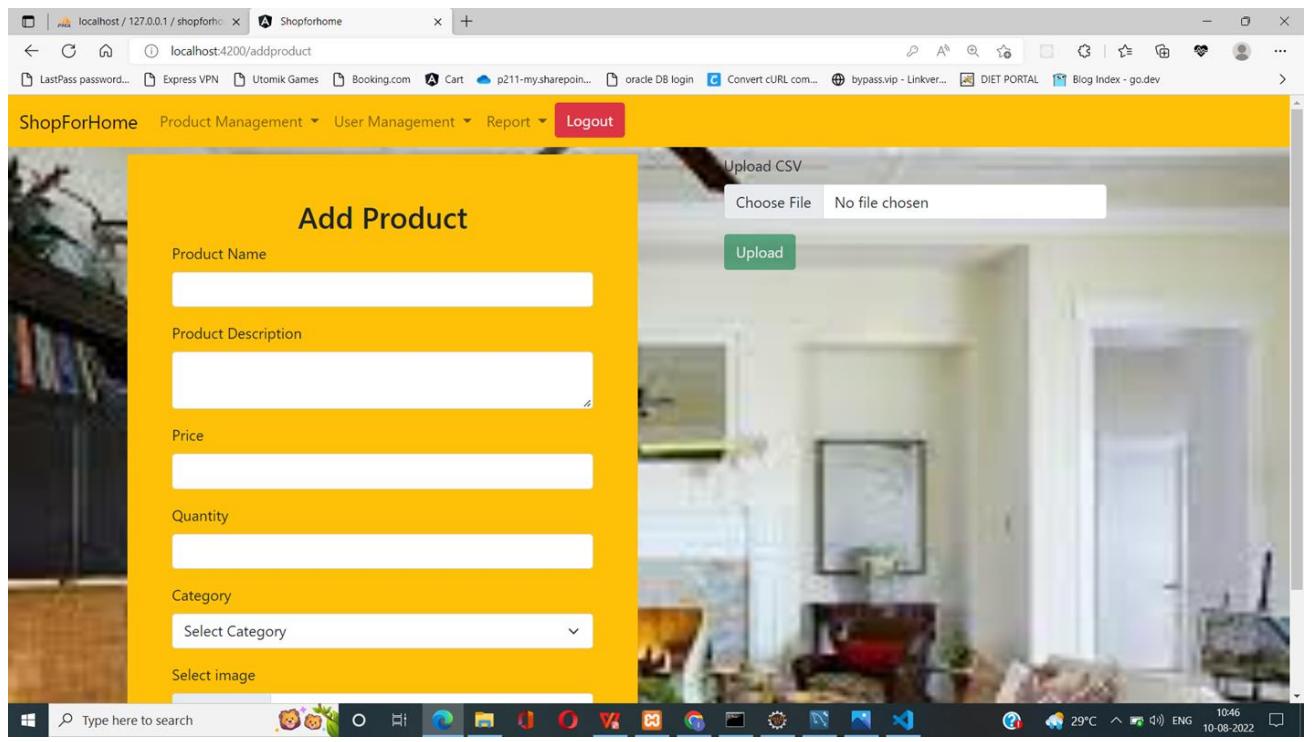


A screenshot of a web browser window showing a list of users. The title of the page is 'Users'. The table has columns: #, Fullname, Username, Email, Mobile, Role, and Action. Each row contains a 'Delete' button. The data is as follows:

#	Fullname	Username	Email	Mobile	Role	Action
1	manas hassija	manas100	manas2000@gmail.com	9234857532	admin	<button>Delete</button>
2	Bhargavi Kunta	BhargaviKunta0583	bhargavikunta99@gmail.com	9347950236	customer	<button>Delete</button>
3	komali radha	komali123	komaliradha2000@gmail.com	7578900059	professor	<button>Delete</button>
4	wipro capstone	capstone	wiprocapstone@gmail.com	1234567890	consumer	<button>Delete</button>
5	Geetha k	Geetha123	geetha@gmail.com	1124558686	Consumer	<button>Delete</button>

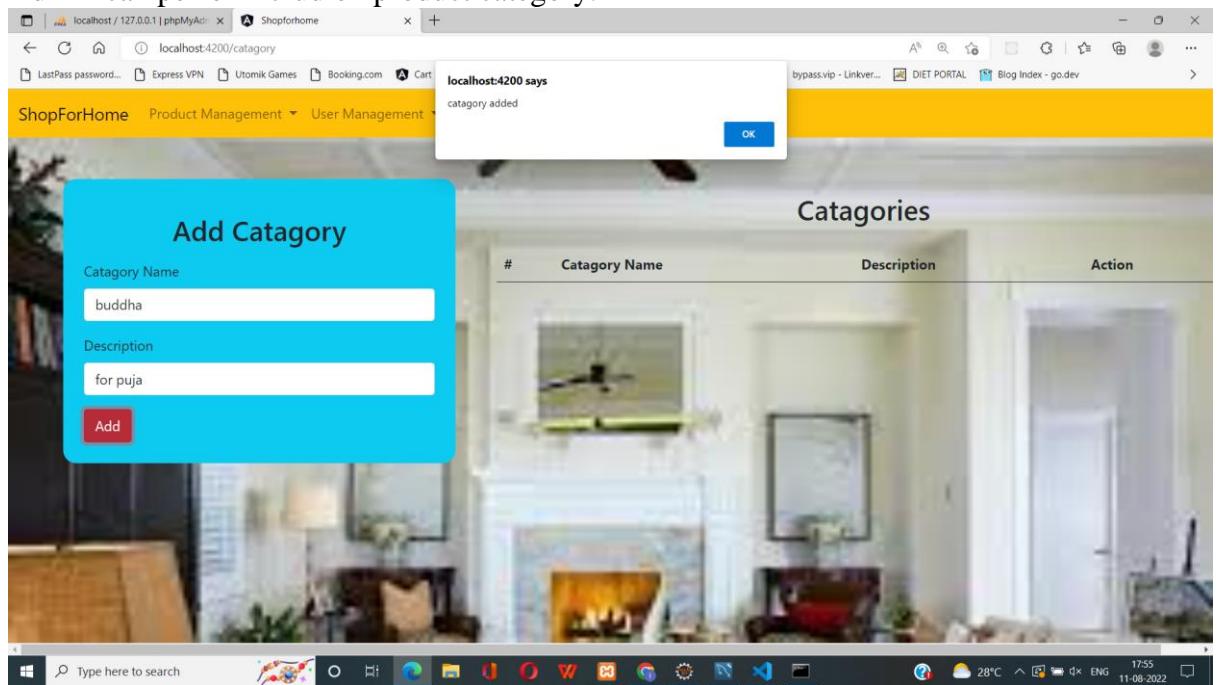
## CRUD ON PRODUCTS (ADMIN):

Admin can perform crud on products.



## CRUD ON PRODUCT CATEGORY(ADMIN):

Admin can perform crud on product category.



The screenshot shows a web browser window for 'Shopforhome' at localhost:4200/catagory. On the left, a modal dialog titled 'Add Catagory' contains fields for 'Catagory Name' (with a placeholder) and 'Description' (with a placeholder), and a red 'Add' button. On the right, a table titled 'Catagories' lists one item: '# 1 buddha' with 'Description: for puja'. The background shows a blurred interior room with furniture and decorations.

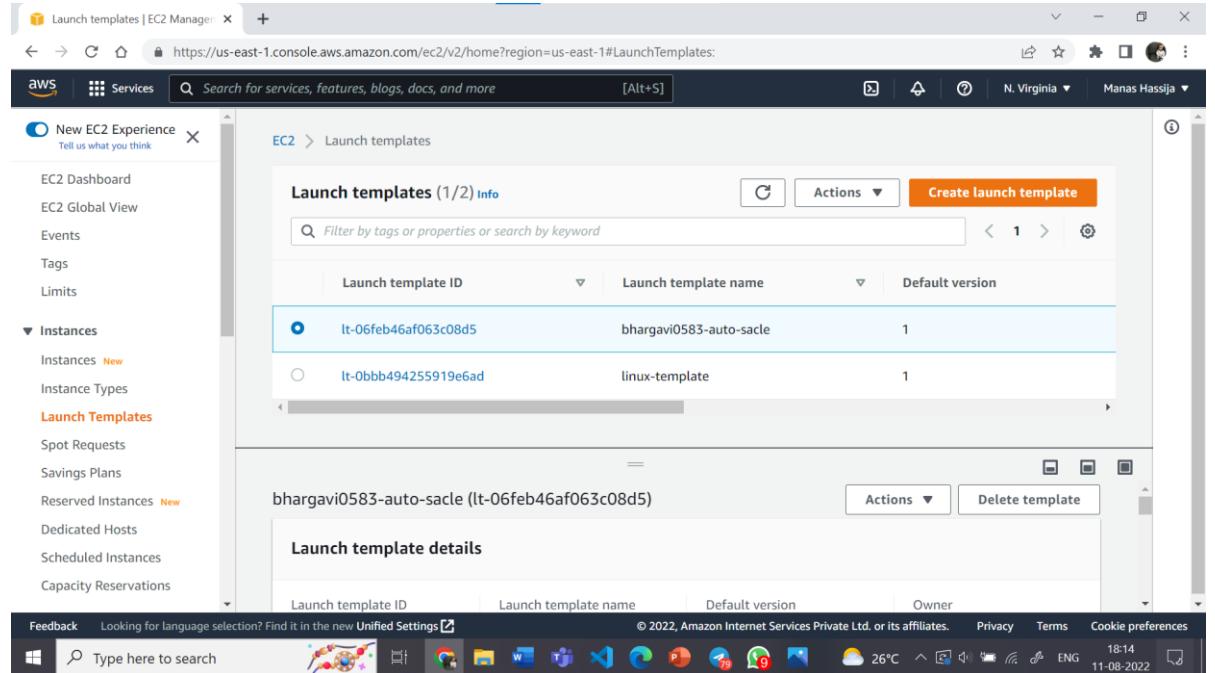
## CRUD ON COUPONS (ADMIN):

Admin can perform crud on coupons.

The screenshot shows a web browser window for 'Shopforhome' at localhost:4200/discount. On the left, a table titled 'Coupons' lists two entries: '1 table Percentage 2' and '2 chair Flat 5'. On the right, a modal dialog titled 'Add Coupon' contains fields for 'Coupon Code' (placeholder), 'Description' (placeholder), 'Type' (dropdown menu), and 'Value' (placeholder), with a red 'Add' button. The background shows a blurred interior room with furniture and decorations.

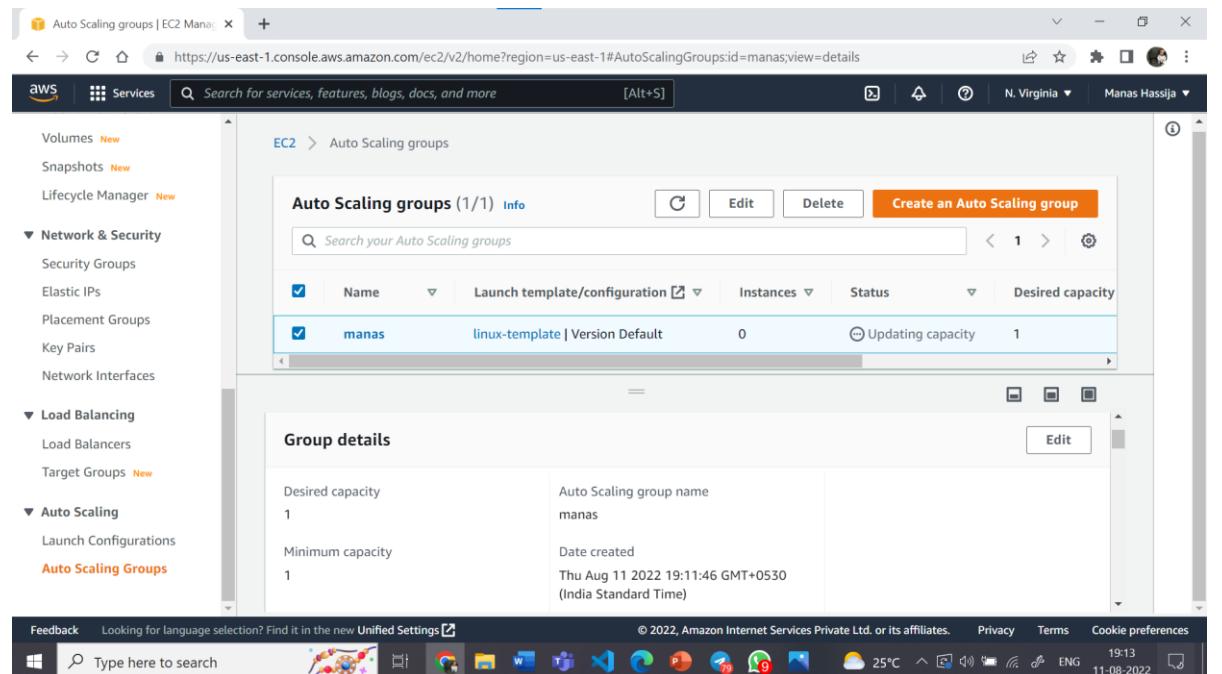
# DEPLOYING EC2 INSTANCES:

## Launch Templates



The screenshot shows the AWS EC2 Launch Templates page. On the left, there's a navigation sidebar with options like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, Instances (selected), Instances, Instance Types, Launch Templates (selected), Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, and Capacity Reservations. The main content area displays a table titled "Launch templates (1/2) Info". It lists two entries: "lt-06feb46af063c08d5" with name "bhargavi0583-auto-sacle" and version 1, and "lt-0bbb494255919e6ad" with name "linux-template" and version 1. Below the table, a modal window titled "bhargavi0583-auto-sacle (lt-06feb46af063c08d5)" shows "Launch template details". The bottom of the screen includes a taskbar with various icons and system status information.

## Auto Scaling Group



The screenshot shows the AWS Auto Scaling Groups page. On the left, there's a navigation sidebar with options like Volumes, Snapshots, Lifecycle Manager, Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), Load Balancing (Load Balancers, Target Groups), and Auto Scaling (Launch Configurations, Auto Scaling Groups selected). The main content area displays a table titled "Auto Scaling groups (1/1) Info". It lists one entry: "manas" with a launch template "linux-template | Version Default", 0 instances, and a status of "Updating capacity" with a value of 1. Below the table, a modal window titled "Group details" shows the configuration for the "manas" group. The bottom of the screen includes a taskbar with various icons and system status information.

## EC2 Instances

The screenshot shows the AWS EC2 Management Console. The left sidebar has 'Instances' selected under 'Instances'. The main area shows a table of instances with one entry:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
linux-server	i-01147900bbc18b73e	Running	t2.micro	2/2 checks passed	No alarms

The details pane for the instance 'linux-server' is open, showing the following information:

Details	Security	Networking	Storage	Status checks	Monitoring	Tags
Instance ID: i-01147900bbc18b73e (linux-server)	Public IPv4 address: 54.144.190.226   open address	Private IPv4 addresses: 172.31.80.193				
IPv6 address: -	Instance state: Running	Public IPv4 DNS: ec2-54-144-190-226.compute-1.amazonaws.com   open address				

At the bottom, the Windows taskbar shows the search bar and various pinned icons.

## Target Group:

The screenshot shows the AWS EC2 Management Console. The left sidebar has 'Target Groups' selected under 'Load Balancing'. The main area shows a table of target groups with one entry:

Name	ARN	Port	Protocol	Target type	Load balancer
manas2-target-group	arn:aws:elasticloadbalancing:us-east-1:179851911950:targetgroup/manas2-target-group/a00264c7dcf2537b	443	HTTPS	Instance	bhargavi0583-load-balancer

The details pane for the target group 'manas2-target-group' is open, showing the following information:

Details
arn:aws:elasticloadbalancing:us-east-1:179851911950:targetgroup/manas2-target-group/a00264c7dcf2537b

At the bottom, the Windows taskbar shows the search bar and various pinned icons.

## Load Balancer:

The screenshot shows the AWS EC2 Management Console with the 'Load Balancers' section selected. On the left sidebar, under 'Load Balancing', 'Load Balancers' is highlighted. The main pane displays a table of load balancers, with one entry selected: 'bhargavi0583-load-balancer'. The table columns include Name, DNS name, State, VPC ID, and Availability Zones. The selected load balancer's details are shown in a modal window, including its basic configuration: Name (bhargavi0583-load-balancer), ARN (arn:aws:elasticloadbalancing:us-east-1:179851911950:loadbalancer/app/bhargavi0583-load-balancer/0a42783df7cdb753), DNS name (bhargavi0583-load-balancer-462992910.us-east-1.elb.amazonaws.com), and State (Active). The modal also has tabs for Description, Listeners, Monitoring, Integrated services, and Tags.

## S3 BUCKET:

The screenshot shows the AWS S3 Management Console with the 'Buckets' section selected. On the left sidebar, under 'Storage Lens', 'Dashboards' is highlighted. The main pane displays a table of buckets, with one entry selected: 'backendmanas'. The table columns include Name, AWS Region, Access, and Creation date. The selected bucket's details are shown in a modal window, including its basic configuration: Name (backendmanas), AWS Region (US East (N. Virginia) us-east-1), Access (Objects can be public), and Creation date (August 10, 2022, 16:19:31 (UTC+05:30)). The modal also has tabs for Account snapshot, Info, and a 'View Storage Lens dashboard' button.

## Building JAR file:

```

[INFO] mvnw
[INFO] Finished at: 2022-08-11T14:27:14+05:30
[INFO] -----
E:\c4\project\Backend\discount>cd ..
E:\c4\project\Backend>cd order
E:\c4\project\Backend\order>mvnw clean install -DskipTests=true
[INFO] Scanning for projects...
[INFO] -----
[INFO] < com.shopforhome:order >-----
[INFO] Building order 0.0.1 SNAPSHOT
[INFO] -----[ jar ]-----
[INFO] ...
[INFO] --- maven-clean-plugin:3.2.0:clean (default-clean) @ order ---
[INFO] Deleting E:\c4\project\Backend\order\target
[INFO] ...
[INFO] --- maven-resources-plugin:3.2.0:resources (default-resources) @ order ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Using 'UTF-8' encoding to copy filtered properties files.
[INFO] Copying 1 resource
[INFO] Copying 0 resource
[INFO] ...
[INFO] --- maven-compiler-plugin:3.10.1:compile (default-compile) @ order ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 8 source files to E:\c4\project\Backend\order\target\classes
[INFO] ...
[INFO] --- maven-resources-plugin:3.2.0:testResources (default-testResources) @ order ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Using 'UTF-8' encoding to copy filtered properties files.
[INFO] skip non existing resourceDirectory E:\c4\project\Backend\order\src\test\resources
[INFO] ...
[INFO] --- maven-compiler-plugin:3.10.1:testCompile (default-testCompile) @ order ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to E:\c4\project\Backend\order\target\test-classes
[INFO] ...
[INFO] --- maven-surefire-plugin:2.22.2:test (default-test) @ order ---
[INFO] Tests are skipped.
[INFO] ...
[INFO] --- maven-jar-plugin:3.2.2:jar (default-jar) @ order ---
[INFO] Building jar: E:\c4\project\Backend\order\target\order-0.0.1-SNAPSHOT.jar
[INFO] ...
[INFO] --- spring-boot-maven-plugin:2.7.1:repackage (repackage) @ order ---
[INFO] Replacing main artifact with repackaged archive
[INFO] ...
[INFO] --- maven-install-plugin:3.2.2:install (default-install) @ order ---
[INFO] Installing E:\c4\project\Backend\order\target\order-0.0.1-SNAPSHOT.jar to C:\Users\91934\.m2\repository\com\shopforhome\order\0.0.1-SNAPSHOT\order-0.0.1-SNAPSHOT.jar
[INFO] Installing E:\c4\project\Backend\order\pom.xml to C:\Users\91934\.m2\repository\com\shopforhome\order\0.0.1-SNAPSHOT\order-0.0.1-SNAPSHOT.pom
[INFO] ...
[INFO] BUILD SUCCESS

```

## Starting the Application shop-for-home.jar:

```

[ec2-user@ip-172-31-80-193:~]
Last login: Thu Aug 11 04:29:10 2022 from 157.48.200.10
[ec2-user@ip-172-31-80-193:~]$ wget https://backendmanas.s3.amazonaws.com/user-0.0.1-SNAPSHOT.jar
--2022-08-11 08:33:14-- https://backendmanas.s3.amazonaws.com/user-0.0.1-SNAPSHOT.jar
Resolving backendmanas.s3.amazonaws.com (backendmanas.s3.amazonaws.com)... 52.216.44.169
Connecting to backendmanas.s3.amazonaws.com (backendmanas.s3.amazonaws.com)|52.216.44.169|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 38883283 (37M) [application/x-www-form-urlencoded]
Saving to: 'user-0.0.1-SNAPSHOT.jar' (4)

100%[=====] 38,883,283 76.5MB/s in 0.5s

2022-08-11 08:33:15 (76.5 MB/s) - 'user-0.0.1-SNAPSHOT.jar' 4 saved [38883283/38883283]

[ec2-user@ip-172-31-80-193:~]$ ls
product-0.0.1-SNAPSHOT.jar  user-0.0.1-SNAPSHOT.jar.1  user-0.0.1-SNAPSHOT.jar.3
user-0.0.1-SNAPSHOT.jar  user-0.0.1-SNAPSHOT.jar.2  user-0.0.1-SNAPSHOT.jar.4
[ec2-user@ip-172-31-80-193:~]$ java -jar user-0.0.1-SNAPSHOT.jar.4

                               .\n  .\n  /\\ \n  ( ) \n  \\\n  :\n  Spring\n  Boot :\n  (v2.7.1)\n\n
2022-08-11 08:33:53.144  INFO 31480 --- [           main] com.shopforhome.user.UserApplication      : Starting UserApplication v0.0.1-SNAPSHOT using Java 17.0.4 on ip-172-31-80-193.ec2.internal with PID 31480 (/home/ec2-user/user-0.0.1-SNAPSHOT.jar.4 started by ec2-user in /home/ec2-user)
2022-08-11 08:33:53.144  INFO 31480 --- [           main] com.shopforhome.user.UserApplication      : No active profile set, falling back to 1 default profile: "default"
2022-08-11 08:33:53.144  INFO 31480 --- [           main] o.s.b.a.AnnotationConfigEmbeddedWebApplicationContext : Bootstraping Spring Data JPA repositories in DEFAULT mode.
2022-08-11 08:33:53.507  INFO 31480 --- [           main] o.s.b.a.AnnotationConfigEmbeddedWebApplicationContext : Finished Spring Data repository scanning in 193 ms. Found 1 JPA repository interfaces.
2022-08-11 08:33:57.067  INFO 31480 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2022-08-11 08:33:57.095  INFO 31480 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-08-11 08:33:57.264  INFO 31480 --- [           main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.64]
2022-08-11 08:33:57.264  INFO 31480 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[] : Initializing Spring embedded WebApplicationContext
2022-08-11 08:33:58.208  INFO 31480 --- [           main] w.s.WebApplicationInitializer : 
2022-08-11 08:33:58.319  INFO 31480 --- [           main] o.hibernate.Version                : 
2022-08-11 08:33:58.677  INFO 31480 --- [           main] o.hibernate.annotations.common.Version : 
2022-08-11 08:33:58.891  INFO 31480 --- [           main] com.zaxxer.hikari.HikariDataSource : 
2022-08-11 08:33:59.916  INFO 31480 --- [           main] com.zaxxer.hikari.HikariDataSource : 
2022-08-11 08:33:59.969  INFO 31480 --- [           main] org.hibernate.dialect.Dialect : 
2022-08-11 08:34:01.595  INFO 31480 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator : 

```

## **8.0 CONCLUSION:**

Hence we created an e-commerce application successfully with all the requirements mentioned in the objective.

