

SHOP FOR HOME

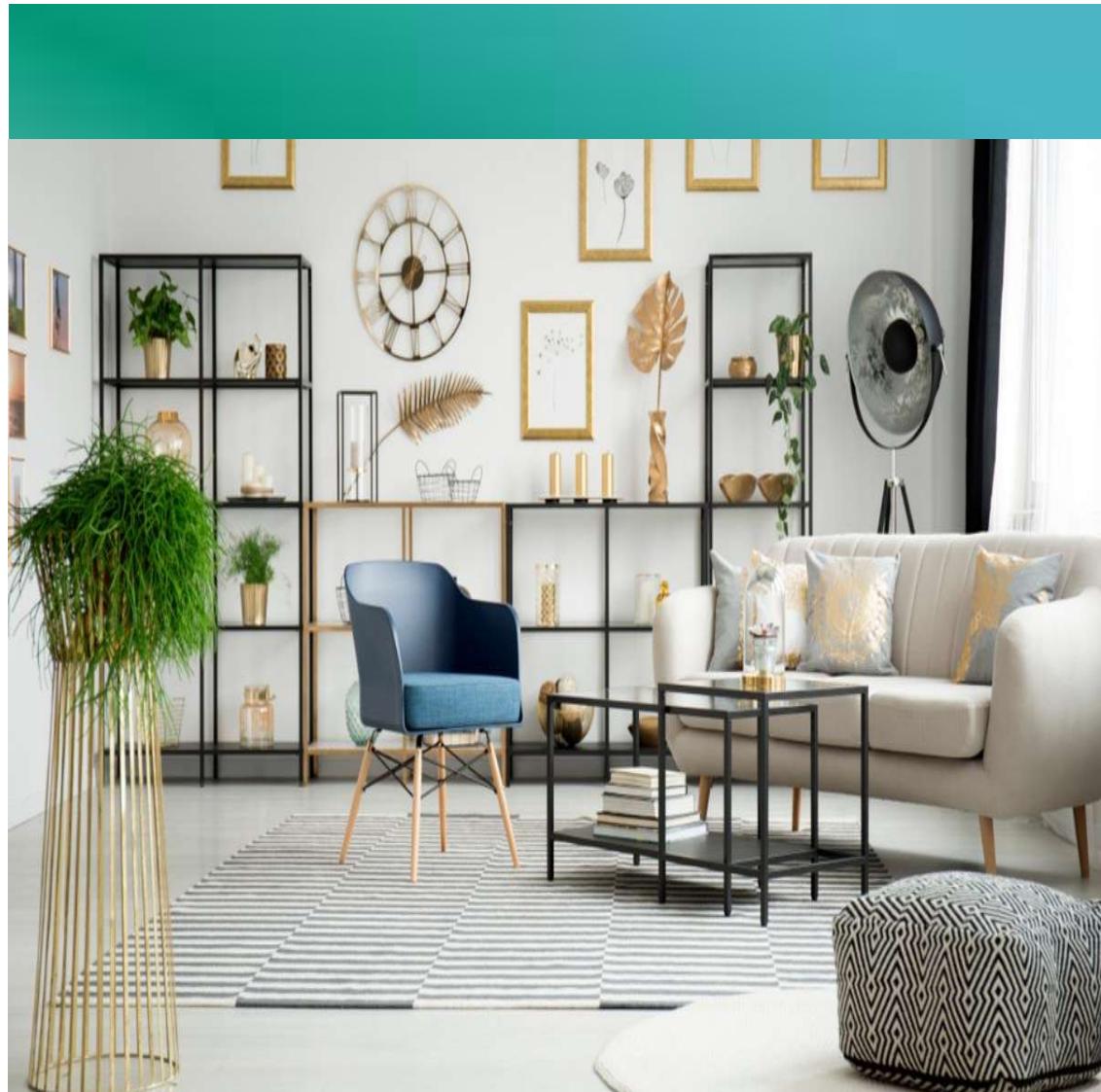
Capstone Project Group-2

MENTOR

Neeraj Kumar Kheria

SUBMITTED BY

A. Sravan
K. Chandana Sri
K. Kavya Sri
K. Kajal Singh
D. Hrishikesh Kiran



Contents

- Problem Statement
- Introduction
- Requirements
- User Stories
- Admin Stories
- Database Schema
- Backend
- AWS Deployment
- Git Repository
- Conclusion

Problem Statement

Shop For Home 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.



Introduction

- Online shopping named ShopForHome is a form of E-commerce website which allows customers to directly buy home decor stuff over the internet using a web browser.
- Online stores are available 24 hours a day, so people can buy at any time.
- The customers must have access to internet to access the ShopForHome application.
- Online Shopping can save time for both the buyer and retailer, reducing phone calls about availability, specifications, hours of operation or other information easily found on company and product pages.

The Application consists of two modules:

1. User Module
2. Admin Module

Requirements

Frontend:



Backend:



Database:



Deployment:



Git Repository:



User Stories

- As a user I should be able to login, Logout and Register into the application.
- As a user I should be able to see the products in different categories.
- As a user I should be able to sort the products.
- As a user I should be able to add the products into the shopping cart.
- As a user I should be able to increase or decrease the quantity added in the cart.
- As a user I should be able to add “n” number of products in the cart.
- As a user I should be able to get the Wishlist option where I can add those products which I want but don’t want to order now.
- As a user I should get different discount coupons.

Home Page

The screenshot shows a web browser window for the 'ShopForHome' website, which is running locally at `localhost:4200/category/6`. The page has a green header bar with the title 'ShopForHome' and a search bar. Below the header is a navigation bar with links for 'Search', 'Wishlist', 'Register', 'Login', 'Logout', and a shopping cart icon showing 'cart value ₹3,499.00' and 'Quantity 2'. On the left side, there is a sidebar with a logo for 'OnlineShop' and a list of categories: 'Indoor plants', 'Photo Frames', 'Wall Prints', 'Timber Decors', 'Ceiling Decors', and a 'Sort' button. The main content area features a large image of a modern living room interior. The room has a dark teal wall, a bright orange sofa, a black coffee table, and various decorative items like deer heads and framed pictures.

Registration Form

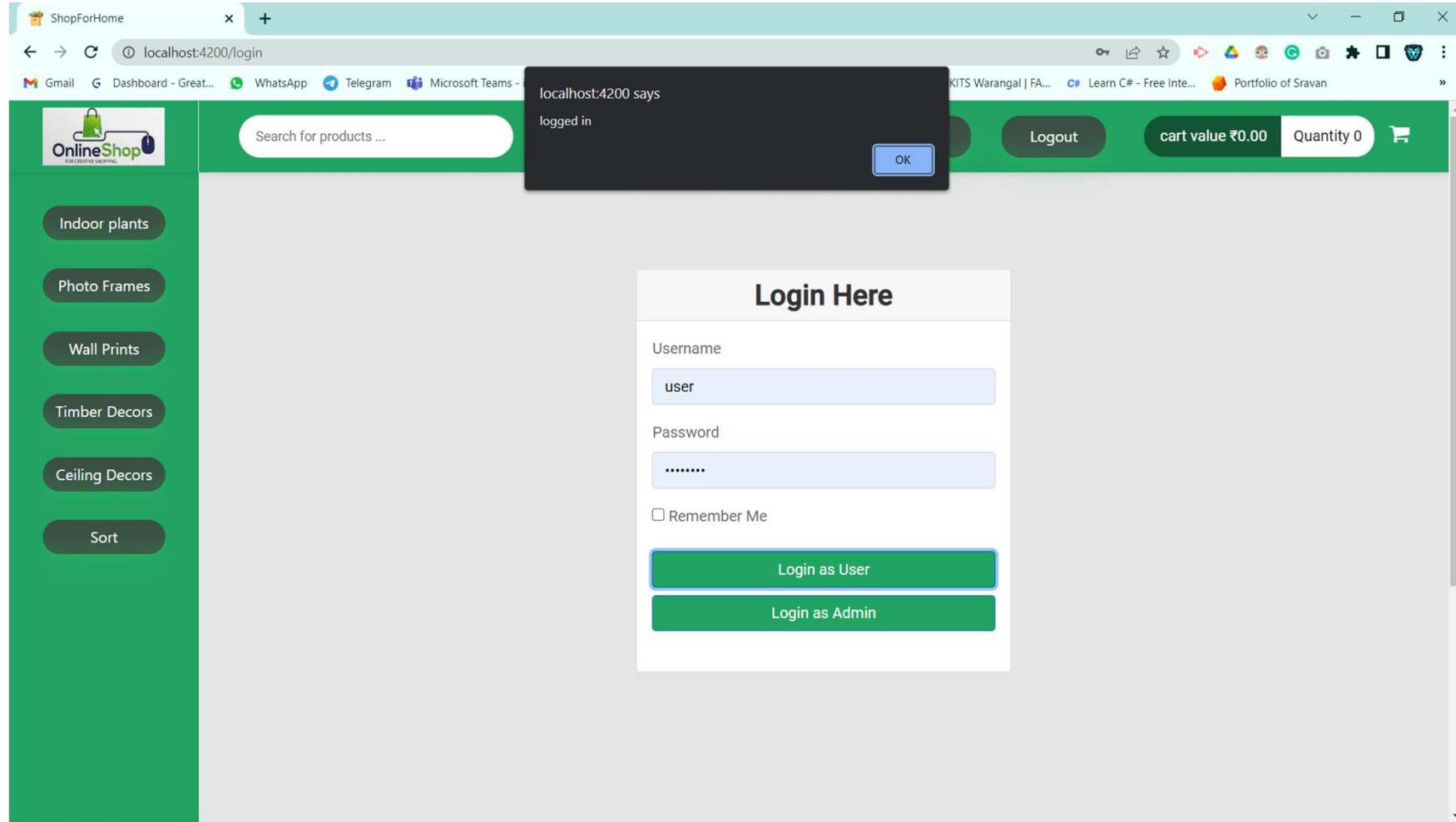
A screenshot of a web browser window titled "ShopForHome" showing a registration form. The URL in the address bar is "localhost:4200/register". The browser interface includes standard navigation buttons (back, forward, search) and a tab bar with various application icons.

The main content area features a dark overlay with a central modal window. The modal has a title "Register Here" and contains fields for Name, Email, Username, Password, and Confirm Password. The "Name" field contains "user", the "Email" field contains "user@gmail.com", the "Username" field contains "user", and both the "Password" and "Confirm Password" fields contain "*****". A large green "Register" button is at the bottom of the modal.

A black notification bar at the top of the page displays the message "localhost:4200 says registered successfully!" with an "OK" button.

The left sidebar of the website lists categories: Indoor plants, Photo Frames, Wall Prints, Timber Decors, Ceiling Decors, and Sort. The header includes a logo for "OnlineShop", a search bar, and a cart summary showing "cart value ₹0.00" and "Quantity 0".

User Login Form



Product Categories

S ShopForHome X + localhost:4200/category/3

Gmail Dashboard - Great... WhatsApp Telegram Microsoft Teams - i... Home | OTS typing practice ZenQ Naukri.com KITS Warangal | FA... C# Learn C# - Free Inte... Portfolio of Sravan

OnlineShop

Search for products ... Search Wishlist Register Login Logout cart value ₹0.00 Quantity 0

Indoor plants
Photo Frames
Wall Prints
Timber Decors
Ceiling Decors
Sort

 Wall Print-1 ₹599.00 Add to Cart Add to Wishlist	 Wall Print-2 ₹899.00 Add to Cart Add to Wishlist	 Wall Print-3 ₹1,500.00 Add to Cart Add to Wishlist	 Wall Print-4 ₹1,600.00 Add to Cart Add to Wishlist
 Wall Print-5 ₹2,000.00 Add to Cart Add to Wishlist	 Wall Print-6 ₹999.00 Add to Cart Add to Wishlist	 Wall Print-7 ₹1,600.00 Add to Cart Add to Wishlist	 Wall Print-8 ₹1,789.00 Add to Cart Add to Wishlist

Get Product

The screenshot shows a web browser window for 'ShopForHome' at localhost:4200/products/20. The page has a green header bar with a navigation bar containing links like Gmail, Dashboard - Great..., WhatsApp, Telegram, Microsoft Teams - i..., Home | OTS, typing practice, ZenQ, Naukri.com, KITS Warangal | FA..., Learn C# - Free Inte..., and Portfolio of Sravan. The main content area features a sidebar on the left with categories: Indoor plants, Photo Frames, Wall Prints, Timber Decors, Ceiling Decors, and Sort. The main content area displays a product image of three potted plants and a white decorative object on a wooden surface. The product title is 'Plant-20' with a price of ₹499.00. Below the title are two buttons: 'Add to cart' (blue) and 'Add to wishlist' (green). A 'Description' section follows, stating: 'Blooming blossoming; flowering; flourishing; thriving in vigor, health and beauty. Blossoming the process and time of budding and unfolding of blossoms. Blushing having a delicate and warm color like some flowers; roseate; blooming'. At the bottom of the page is a link 'Back to Product List'.

ShopForHome

localhost:4200/products/20

Gmail Dashboard - Great... WhatsApp Telegram Microsoft Teams - i... Home | OTS typing practice ZenQ Naukri.com KITS Warangal | FA... Learn C# - Free Inte... Portfolio of Sravan

OnlineShop FOR CREATIVE SHOPPING

Search for products ...

Search Wishlist Register Login Logout cart value ₹0.00 Quantity 0

Indoor plants

Photo Frames

Wall Prints

Timber Decors

Ceiling Decors

Sort

Plant-20

₹499.00

Add to cart Add to wishlist

Description

Blooming blossoming; flowering; flourishing; thriving in vigor, health and beauty. Blossoming the process and time of budding and unfolding of blossoms. Blushing having a delicate and warm color like some flowers; roseate; blooming

[Back to Product List](#)

Sort

Screenshot of an e-commerce website for "ShopForHome" showing a product category page for wall prints.

The page features a green header bar with the shop logo, search bar, and navigation links (Search, Wishlist, Register, Login, Logout). It also displays a cart summary with a value of ₹3,499.00 and a quantity of 2.

The sidebar on the left lists categories: Indoor plants, Photo Frames, Wall Prints, Timber Decors, Ceiling Decors, and a prominent "Sort" button.

The main content area displays eight wall print products arranged in two rows of four:

Wall Print	Price
Wall Print-1	₹500.00
Wall Print-2	₹700.00
Wall Print-3	₹800.00
Wall Print-4	₹1,000.00
Wall Print-5	₹1,200.00
Wall Print-6	₹1,300.00
Wall Print-7	₹1,450.00
Wall Print-8	₹1,550.00

Each product card includes a thumbnail image, the product name, the price (highlighted with a red box), and two buttons: "Add to Cart" and "Add to Wishlist". Blue arrows point from the price boxes of the first seven products towards the price box of the eighth product, illustrating a sorting operation where the eighth item is being compared or positioned relative to the others.

Wishlist

ShopForHome localhost:4200/wishlist

Gmail Dashboard - Great... WhatsApp Telegram Microsoft Teams - i... Home | OTS typing practice ZenQ. Naukri.com KITS Warangal | FA... Learn C# - Free Inte... Portfolio of Sravan

Search for products ... Search Wishlist Register Login cart value ₹0.00 Quantity 0

OnlineShop FOR CREATIVES SHOPPING

Indoor plants
Photo Frames
Wall Prints
Timber Decors
Ceiling Decors
Sort

9. Plant-9

1. Plant-1

2. Plant-2

6. Plant-6

27. Frame-2

localhost:4200/wishlist

The screenshot shows a web browser window for 'ShopForHome' at the URL 'localhost:4200/wishlist'. The page has a green header bar with various links and a search bar. A red box highlights the 'Wishlist' button in the header. The main content area displays a list of items, each with a small image and a label. The items are: 9. Plant-9, 1. Plant-1, 2. Plant-2, 6. Plant-6, and 27. Frame-2. On the left side, there is a sidebar with categories like Indoor plants, Photo Frames, Wall Prints, etc. At the bottom left, there is a footer bar with the same URL.

Cart

ShopForHome

localhost:4200/cart-details

Gmail Dashboard - Great... WhatsApp Telegram Microsoft Teams - i... Home | OTS typing practice ZenQ Naukri.com KITS Warangal | FA... Learn C# - Free Inte... Portfolio of Sravan

OnlineShop

Search for products ...

Search Wishlist Register Logout cart value ₹3,499.00 Quantity 2

Indoor plants Photo Frames Wall Prints Timber Decors Ceiling Decors Sort

Product Image	Product Detail	Order Details
	Plant-4 ₹499.00	Product Quantity: 1 Subtotal: ₹499.00
	Timber Decor-7 ₹3,000.00	Product Quantity: 1 Subtotal: ₹3,000.00
	discount-coupon	Apply
		Total Items Quantity: 2 Shipping: Free Total Price: ₹3,499.00

Check Out

Discount Coupon

Screenshot of the ShopForHome website showing the cart details page with a discount coupon applied.

The browser title is "ShopForHome" and the URL is "localhost:4200/cart-details".

The cart contains two items:

- Plant-4**: ₹499.00
- Timber Decor-7**: ₹3,000.00

A coupon code **C3BATCH2** is entered in the input field and applied.

The applied discount message is: **Hurrey..! 50% Discount Applied**.

The final order summary is:

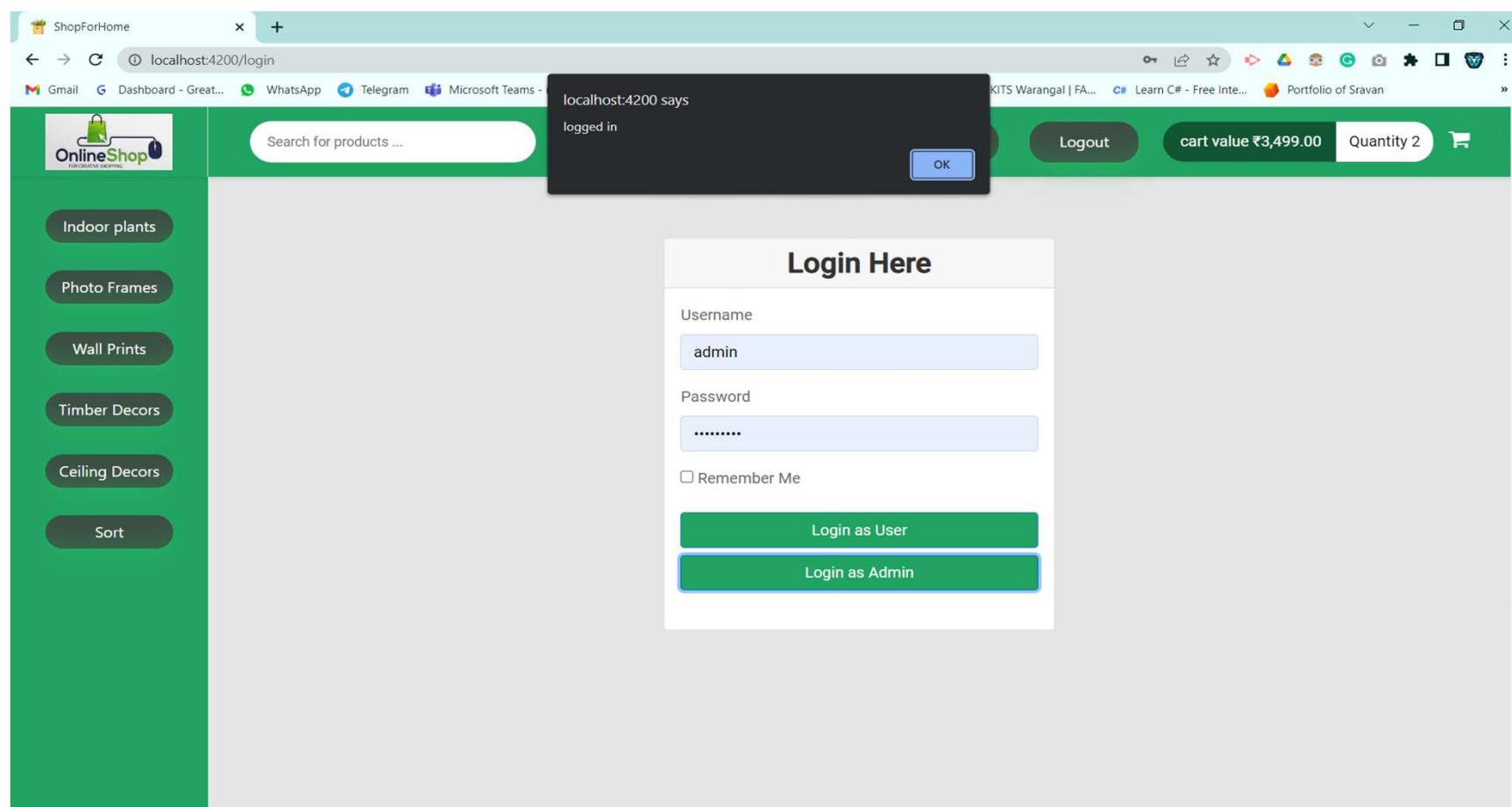
- Total Items Quantity: 2**
- Shipping: Free**
- Total Price: ₹1,749.50**

A red box highlights the coupon input field and the applied discount message.

Admin Stories

- As an Admin I should be able to login, Logout and Register into the application.
- As an Admin I should be able to perform CRUD on Users.
- As an Admin I should be able to Perform CRUD on the products.
- As an Admin I should be able to get bulk upload option to upload a csv for products details
- As an Admin I should be able to get the stocks.
- As an Admin I should be able to mail if any stock is less than 10.
- As an Admin I should be able to get the sales report of a specific duration.
- As an Admin I should be able to set the discount coupons for the specific set of users

Admin Login



Admin Dashboard

The screenshot shows a web browser window titled "ShopForHome" with the URL "localhost:4200/admin". The page has a green header bar with various links and a search bar. On the left, there's a sidebar with categories like "Indoor plants", "Photo Frames", "Wall Prints", "Timber Decors", "Ceiling Decors", and "Sort". The main content area is titled "Admin Dashboard" and contains a section for "CRUD ON USER" with the following options:

- 1. Show All user
- 2. Create A user
- 3. Read A user
- 4. Delete A user
- 5. Update A user

Below this, there's a form field labeled "Enter a choice :" containing the number "1". A table below shows two user entries:

ID	Username	Name	email
1	admin	admin	admin@gmail.com
3	user	user	user@gmail.com

Admin Product Edit

The screenshot shows a web application interface for managing products. The top navigation bar includes links for Gmail, Dashboard - Great..., WhatsApp, Telegram, Microsoft Teams - i..., Home | OTS, typing practice, ZenQ, Naukri.com, KITS Warangal | FA..., Learn C# - Free Inte..., and Portfolio of Sravan. The main header features a logo for "OnlineShop FOR CREATIVE SHOPPING", a search bar, and buttons for Search, Wishlist, Register, Login, Logout, cart value ₹3,499.00, Quantity 2, and a shopping cart icon.

The left sidebar contains buttons for Indoor plants, Photo Frames, Wall Prints, Timber Decors, Ceiling Decors, and Sort. The main content area is titled "CRUD ON PRODUCTS" and lists the following options:

- 1. Show All Products
- 2. Create A Prouct
- 3. Read A Product
- 4. Delete A Product
- 5. get all stocks
- 6. get sales report

An input field labeled "Enter a choice :" contains the value "1". Below this, a table displays a list of products:

Name	sku	unit price
Plant-1	SHOP-FOR-HOME00	399
Plant-2	SHOP-FOR-HOME01	599
Plant-3	SHOP-FOR-HOME02	799
Plant-4	SHOP-FOR-HOME03	499
Plant-5	SHOP-FOR-HOME04	799
Plant-6	SHOP-FOR-HOME05	699

Bulk Upload

The screenshot shows a web browser window for 'ShopForHome' at 'localhost:4200/admin'. The page has a green header bar with a logo, a search bar, and navigation links for 'Search', 'Wishlist', 'Register', 'Login', 'Logout', 'cart value ₹3,499.00', and 'Quantity 2'. On the left, there's a sidebar with categories: 'Indoor plants', 'Photo Frames', 'Wall Prints', 'Timber Decor', 'Ceiling Decor', and 'Sort'. The main content area is titled 'Bulk Upload' and features a file input field with 'Choose File BulkUpload1.csv'. Below it is a table with 10 rows of data:

ID	Product Name	Product Description	Stock Available	Price
1	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	699
2	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	599
3	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	1999
4	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	2099
5	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	699
6	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	599
7	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	1899
8	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	2099
9	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	1999
10	Photo Frame	The photo frame is a home decor item which highlights our memories on a wall or table	100	1699

Stock Report

The screenshot shows a web browser window for 'ShopForHome' at localhost:4200/admin. The interface includes a navigation bar with links like 'Dashboard - Great...', 'WhatsApp', 'Telegram', 'Microsoft Teams - i...', 'Home | OTS', 'typing practice', 'ZenQ.', 'Naukri.com', 'KITS Warangal | FA...', 'Learn C# - Free Inte...', and 'Portfolio of Sravan'. The main content area features a sidebar with categories: 'Indoor plants', 'Photo Frames', 'Wall Prints', 'Timber Decors', 'Ceiling Decors', and 'Sort'. A search bar at the top has placeholder text 'Search for products ...'. Below it are buttons for 'Search', 'Wishlist', 'Register', 'Login', 'Logout', and a shopping cart icon showing 'cart value ₹3,499.00' and 'Quantity 2'. A central panel titled 'CRUD ON PRODUCTS' contains a numbered list of actions: 1. Show All Products, 2. Create A Prouct, 3. Read A Product, 4. Delete A Product, 5. get all stocks, and 6. get sales report. An input field below the list contains the number '5'. The main table displays product data:

Name	sku	unit price	units in stock
Plant-1	SHOP-FOR-HOME00	399	100
Plant-2	SHOP-FOR-HOME01	599	100
Plant-3	SHOP-FOR-HOME02	799	100
Plant-4	SHOP-FOR-HOME03	499	100
Plant-5	SHOP-FOR-HOME04	799	100
Plant-6	SHOP-FOR-HOME05	699	100
Plant-7	SHOP-FOR-HOME06	999	100

Sales Report

The screenshot shows a web application interface for managing products in an online shop. The left sidebar has a green background with various categories: Indoor plants, Photo Frames, Wall Prints, Timber Decors, Ceiling Decors, and Sort. The main content area has a white background with a green header bar containing a logo, a search bar, and navigation links for Search, Wishlist, Register, Login, Logout, cart value (₹3,499.00), Quantity 2, and a shopping cart icon.

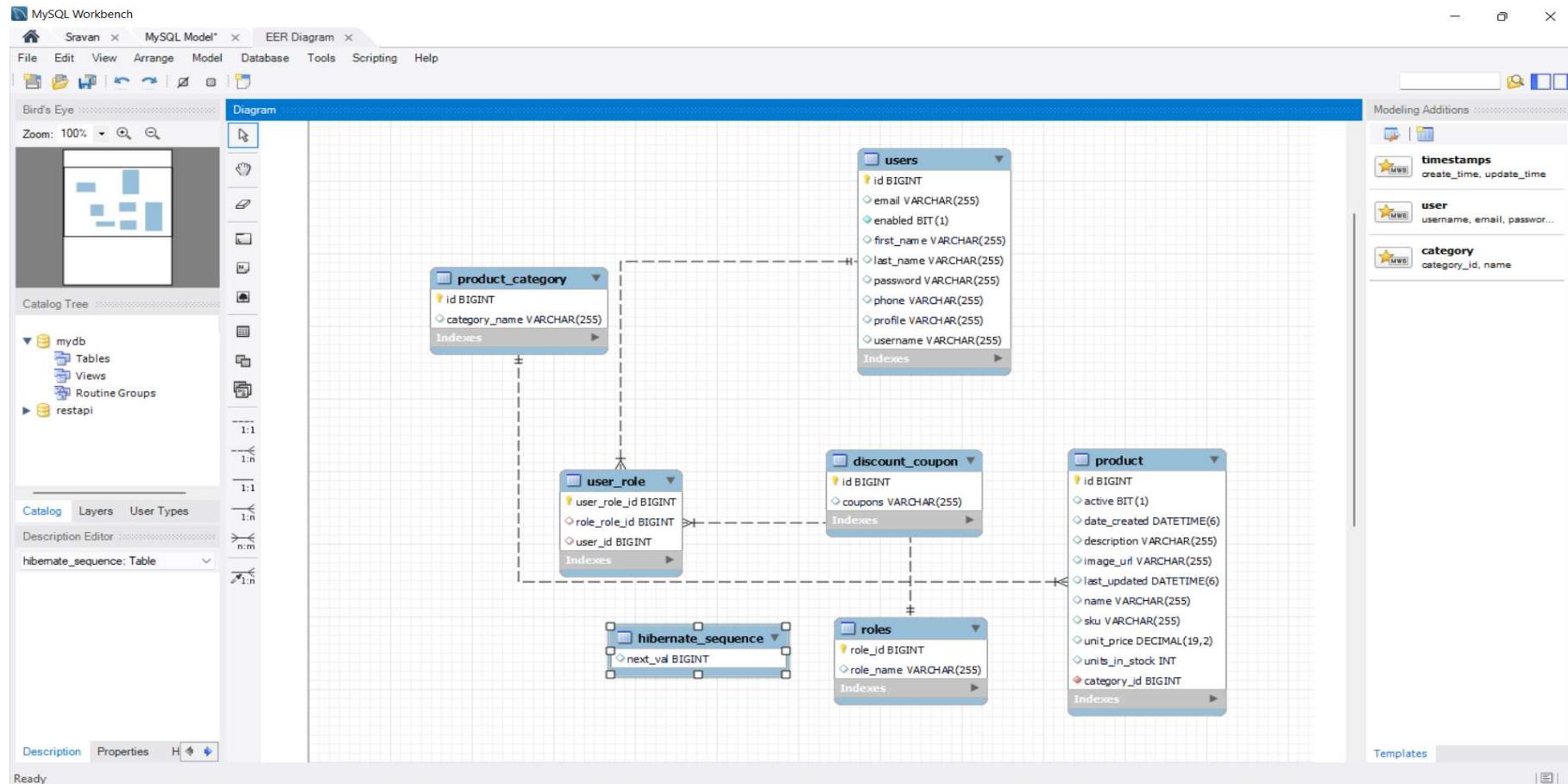
The central part of the page is titled "CRUD ON PRODUCTS" and contains a numbered list of actions:

1. Show All Products
2. Create A Prouct
3. Read A Product
4. Delete A Product
5. get all stocks
6. get sales report

Below this list, there is an input field labeled "Enter a choice :" with the number "6" typed into it. Below the input field is a table showing product data:

Name	sku	unit price	units in stock	Date Created
Plant-1	SHOP-FOR-HOME00	399	100	2022-08-04T11:52:47.000+00:00
Plant-2	SHOP-FOR-HOME01	599	100	2022-08-04T11:52:47.000+00:00
Plant-3	SHOP-FOR-HOME02	799	100	2022-08-04T11:52:47.000+00:00
Plant-4	SHOP-FOR-HOME03	499	100	2022-08-04T11:52:47.000+00:00
Plnat-5	SHOP-FOR-HOME04	799	100	2022-08-04T11:52:47.000+00:00
Plant-6	SHOP-FOR-HOME05	699	100	2022-08-04T11:52:47.000+00:00
Plant-7	SHOP-FOR-HOME06	999	100	2022-08-04T11:52:47.000+00:00

Database Schema



MySql Product Categories

The screenshot shows the MySQL Workbench interface with the following details:

- Top Bar:** MySQL Workbench, unconnected, Sravan.
- Menu Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL Workbench toolbar icons.
- Navigator:** Schemas, Filter objects, sys.
- Query Editor:** Query 1, SQL File 1*, new sql*. The code is as follows:

```
1 •  create database if not exists restapi;
2 •  use restapi;
3 •  INSERT INTO product_category(category_name) VALUES ('Indoor plants');
4 •  INSERT INTO product_category(category_name) VALUES ('Photo Frames');
5 •  INSERT INTO product_category(category_name) VALUES ('Wall Prints');
6 •  INSERT INTO product_category(category_name) VALUES ('Timber Decors');
7 •  INSERT INTO product_category(category_name) VALUES ('Ceiling Decors');
8 •  INSERT INTO product_category(category_name) VALUES ('Sort');
```

- SQLAdditions:** Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
- Output Tab:** Action Output, showing the execution log:

#	Time	Action	Message	Duration / Fetch
1	11:52:15	DROP DATABASE 'restapi'	7 row(s) affected	0.172 sec
2	11:52:21	create database if not exists restapi	1 row(s) affected	0.000 sec
3	11:52:21	use restapi	0 row(s) affected	0.000 sec
4	11:52:32	INSERT INTO product_category(category_name) VALUES ('Indoor plants')	1 row(s) affected	0.000 sec
5	11:52:32	INSERT INTO product_category(category_name) VALUES ('Photo Frames')	1 row(s) affected	0.000 sec
6	11:52:32	INSERT INTO product_category(category_name) VALUES ('Wall Prints')	1 row(s) affected	0.000 sec
7	11:52:32	INSERT INTO product_category(category_name) VALUES ('Timber Decors')	1 row(s) affected	0.000 sec
8	11:52:32	INSERT INTO product_category(category_name) VALUES ('Ceiling Decors')	1 row(s) affected	0.000 sec
9	11:52:32	INSERT INTO product_category(category_name) VALUES ('Sort')	1 row(s) affected	0.000 sec
10	11:52:47	INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id, date_created, date_updated) VALUES ('P1', 'Indoor Plants', 'A variety of indoor plants.', 'https://...', 1, 10, 10.0, 1, '2023-10-10 11:52:47', '2023-10-10 11:52:47')	1 row(s) affected	0.015 sec
11	11:52:47	INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id, date_created, date_updated) VALUES ('P2', 'Photo Frames', 'A collection of various photo frames.', 'https://...', 1, 10, 10.0, 2, '2023-10-10 11:52:47', '2023-10-10 11:52:47')	1 row(s) affected	0.000 sec
12	11:52:47	INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id, date_created, date_updated) VALUES ('P3', 'Wall Prints', 'A set of wall prints featuring various designs.', 'https://...', 1, 10, 10.0, 3, '2023-10-10 11:52:47', '2023-10-10 11:52:47')	1 row(s) affected	0.000 sec
13	11:52:47	INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id, date_created, date_updated) VALUES ('P4', 'Timber Decors', 'A range of timber decors including shelves and boxes.', 'https://...', 1, 10, 10.0, 4, '2023-10-10 11:52:47', '2023-10-10 11:52:47')	1 row(s) affected	0.000 sec
14	11:52:47	INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id, date_created, date_updated) VALUES ('P5', 'Ceiling Decors', 'A collection of ceiling decors like fans and lights.', 'https://...', 1, 10, 10.0, 5, '2023-10-10 11:52:47', '2023-10-10 11:52:47')	1 row(s) affected	0.000 sec
15	11:52:47	INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id, date_created, date_updated) VALUES ('P6', 'Sort', 'A general category for sorting products.', 'https://...', 1, 10, 10.0, 6, '2023-10-10 11:52:47', '2023-10-10 11:52:47')	1 row(s) affected	0.000 sec

Database Product Details

The screenshot shows the MySQL Workbench interface with a query editor containing SQL code for inserting data into a 'product' table. The code is as follows:

```
-- Indoor plants
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME00', 'Plant-1'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME01', 'Plant-2'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME02', 'Plant-3'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME03', 'Plant-4'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME04', 'Plant-5'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME05', 'Plant-6'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME06', 'Plant-7'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME07', 'Plant-8'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME08', 'Plant-9'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME09', 'Plant-10'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME10', 'Plant-11'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME11', 'Plant-12'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME12', 'Plant-13'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME13', 'Plant-14'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME14', 'Plant-15'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME15', 'Plant-16'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME16', 'Plant-17'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME17', 'Plant-18'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME18', 'Plant-19'
INSERT INTO product (sku, name, description, image_url, active, units_in_stock, unit_price, category_id,date_created) VALUES ('SHOP-FOR-HOME19', 'Plant-20')
```

The MySQL Workbench interface includes a toolbar, a query editor with syntax highlighting, a results pane showing the execution log, and various navigation and administration tools.

Backend API Application

The screenshot shows the Spring Tool Suite 4 interface with the following details:

- Project Explorer:** Displays the project structure for "demo [boot]". It includes packages like com.example.demo, com.example.demo.config, com.example.demo.entity, com.example.demo.repo, com.example.demo.service, and com.example.demo.service.impl. It also shows src/main/java, src/main/resources, src/test/java, and Maven Dependencies.
- Code Editor:** Shows the file `DemoApplication.java` with the following code:

```
package com.example.demo;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
public class DemoApplication {
    public static void main(String[] args) {
        SpringApplication.run(DemoApplication.class, args);
    }
}
```
- Console:** Displays the Spring Boot startup logs:

```
2022-08-04 11:52:56.555 INFO 1228 --- [           main] com.example.demo.DemoApplication      : Starting DemoApplication using Java 17.0.3 on Sravan with PID 1228
2022-08-04 11:52:56.560 INFO 1228 --- [           main] com.example.demo.DemoApplication      : No active profile set, falling back to 1 default profile: "default"
2022-08-04 11:52:57.541 INFO 1228 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2022-08-04 11:52:57.621 INFO 1228 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 66 ms. Found 5 JPA r
2022-08-04 11:52:58.294 INFO 1228 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8081 (http)
2022-08-04 11:52:58.311 INFO 1228 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-08-04 11:52:58.311 INFO 1228 --- [           main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.63]
2022-08-04 11:52:58.550 INFO 1228 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2022-08-04 11:52:58.551 INFO 1228 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1936 ms
2022-08-04 11:52:59.074 INFO 1228 --- [           main] o.h.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2022-08-04 11:52:59.154 INFO 1228 --- [           main] org.hibernate.Version                    : HHH000412: Hibernate ORM core version 5.6.9.Final
2022-08-04 11:52:59.368 INFO 1228 --- [           main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
2022-08-04 11:52:59.525 INFO 1228 --- [           main] com.zaxxer.hikari.HikariDataSource   : HikariPool-1 - Starting...
2022-08-04 11:52:59.832 INFO 1228 --- [           main] com.zaxxer.hikari.HikariDataSource   : HikariPool-1 - Start completed.
2022-08-04 11:52:59.857 INFO 1228 --- [           main] org.hibernate.dialect.Dialect       : HHH000400: Using dialect: org.hibernate.dialect.MySQL8Dialect
2022-08-04 11:53:00.892 INFO 1228 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator  : HHH000490: Using JtaPlatform implementation: [org.hibernate.eng]
```
- Boot Dashboard:** Shows a summary of the application's status, including memory usage and metrics.

Application Properties

The screenshot shows the Spring Tool Suite 4 interface with the following details:

- Package Explorer:** Displays the project structure for "demo [boot]".
- application.properties:** The contents of the application properties file are shown, including database configuration and other application settings.
- Console:** Shows the log output from the application startup. It includes Spring Boot initialization messages, standard Java logs, and specific Hibernate and Tomcat logs.

```
workspace-spring-tool-suite-4-4.15.1.RELEASE - demo/src/main/resources/application.properties - Spring Tool Suite 4
File Edit Navigate Search Project Run Window Help
Package Explorer X DemoApplication.java application.properties X Out... X
1 spring.datasource.url=jdbc:mysql://localhost:3306/restapi?useSSL=false&useUnicode=yes&characterEncoding=UTF-8&allowPublicKeyRetrieval=true&se^
2 spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect
3 #spring.datasource.url=jdbc:mysql://localhost:3306/restapi
4 spring.datasource.username=root
5 spring.datasource.password=Sravan99%
6 spring.jpa.hibernate.ddl-auto=update
7 server.port=8081
8 spring.data.rest.basePath=/api
9 spring.jpa.show-sql=true
10 spring.jpa.properties.hibernate.format_sql=true
11
12

Problems Javadoc Declaration Console X
demo - DemoApplication [Spring Boot App] C:\sts-4.15.1.RELEASE\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Aug 4, 2022, 11:52:55 AM) [pid: 1228]
:: Spring Boot :: (v2.7.0)

2022-08-04 11:52:56.555 INFO 1228 --- [           main] com.example.demo.DemoApplication      : Starting DemoApplication using Java 17.0.3 on Sravan with PID 1
2022-08-04 11:52:56.560 INFO 1228 --- [           main] com.example.demo.DemoApplication      : No active profile set, falling back to 1 default profile: "defau
2022-08-04 11:52:57.541 INFO 1228 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2022-08-04 11:52:57.621 INFO 1228 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 66 ms. Found 5 JPA r
2022-08-04 11:52:58.294 INFO 1228 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8081 (http)
2022-08-04 11:52:58.311 INFO 1228 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-08-04 11:52:58.311 INFO 1228 --- [           main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.63]
2022-08-04 11:52:58.550 INFO 1228 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[] : Initializing Spring embedded WebApplicationContext
2022-08-04 11:52:58.551 INFO 1228 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1936 ms
2022-08-04 11:52:59.074 INFO 1228 --- [           main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2022-08-04 11:52:59.154 INFO 1228 --- [           main] org.hibernate.Version                  : HHH000412: Hibernate ORM core version 5.6.9.Final
2022-08-04 11:52:59.368 INFO 1228 --- [           main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
2022-08-04 11:52:59.525 INFO 1228 --- [           main] com.zaxxer.hikari.HikariDataSource   : HikariPool-1 - Starting...
2022-08-04 11:52:59.832 INFO 1228 --- [           main] com.zaxxer.hikari.HikariDataSource   : HikariPool-1 - Start completed.
2022-08-04 11:52:59.857 INFO 1228 --- [           main] org.hibernate.dialect.Dialect       : HHH000400: Using dialect: org.hibernate.dialect.MySQL8Dialect
2022-08-04 11:53:00.892 INFO 1228 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator  : HHH000490: Using JtaPlatform implementation: [org.hibernate.eng
1 elements hidden by filter
```

Aws Deployment using S3

The screenshot shows the AWS S3 console interface. The left sidebar has a tree view with 'Buckets' selected. The main content area shows the properties for the bucket 'c3.capstoneproject.batch2.com'. The 'Properties' tab is active. Key details shown include:

- Bucket Overview:** AWS Region: US East (N. Virginia) us-east-1; Amazon Resource Name (ARN): arn:aws:s3:::c3.capstoneproject.batch2.com; Creation date: August 4, 2022, 13:28:31 (UTC+05:30)
- Bucket Versioning:** Bucket Versioning: Disabled. A note states: "Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures." An 'Edit' button is present.
- Other Options:** Multi-factor authentication (MFA) delete.

The browser tabs at the top show other open windows like 'AWS LAB: Program Overview', 'AWS (Hands-On-Lab)', 'c3.capstoneproject.batch2.com', 'IAM Management Console', and 'Shop For Home'.

EC2 Instance

The screenshot shows the AWS EC2 Management Console interface. On the left, a navigation sidebar lists services like EC2, VPC, Lambda, S3, RDS, IAM, Route 53, S3 Glacier, Storage Gateway, and DynamoDB. The main area displays a table of instances with one entry:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
Ec2-instance	i-07ce2ddf99fc2470e	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-52-23-

Below the table, a detailed view for the instance 'i-07ce2ddf99fc2470e (Ec2-instance)' is shown. The 'Details' tab is selected, displaying the following information:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-07ce2ddf99fc2470e (Ec2-instance)	52.23.187.225 open address	172.31.24.197
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-52-23-187-225.compute-1.amazonaws.com open address
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-24-197.ec2.internal	ip-172-31-24-197.ec2.internal	-
Answer private resource DNS name	Instance type	
IPv4 (A)	t2.micro	

At the bottom of the page, there are links for Feedback, Unified Settings, Copyright notice (© 2022, Amazon Internet Services Private Ltd. or its affiliates.), Privacy, Terms, and Cookie preferences.

Load Balancer

The screenshot shows the AWS EC2 Management Console interface. The left sidebar is collapsed, and the main area displays the 'Create Load Balancer' configuration page. The 'Basic Configuration' section is visible, showing details for a load balancer named 'LoadBalancer'. The configuration includes:

Name	Value
Name	LoadBalancer
ARN	arn:aws:elasticloadbalancing:us-east-1:559242741107:loadbalancer/app/LoadBalancer/bdd9594c15dbbc32
DNS name	LoadBalancer-1989575933.us-east-1.elb.amazonaws.com (A Record)
State	Provisioning
Type	application
Scheme	internet-facing
IP address type	ipv4

The 'Listeners' tab is selected in the navigation bar below the table. The top of the page shows a search bar and a list of other services like S3, RDS, IAM, Route 53, etc.

Auto Scaling

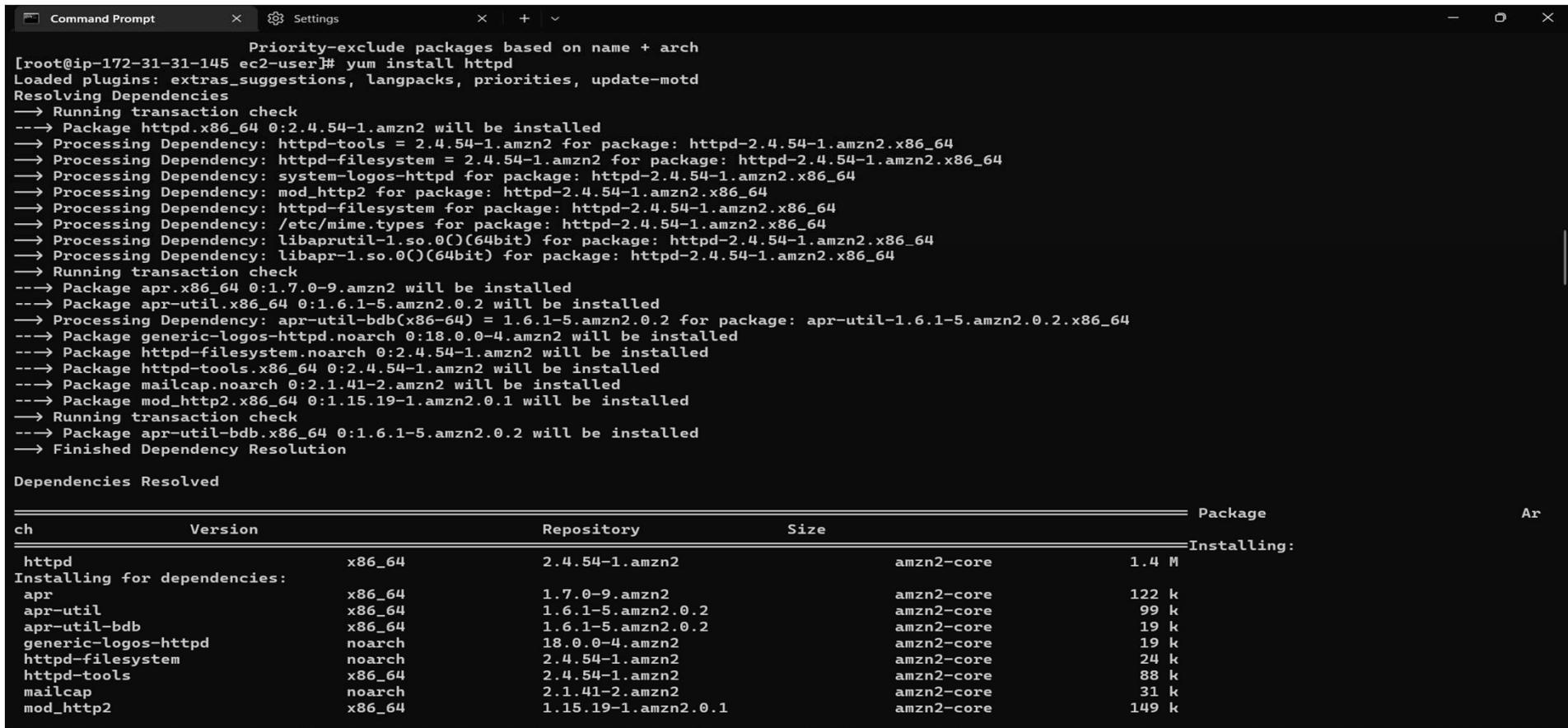
The screenshot shows the AWS Management Console interface for Auto Scaling groups. The left sidebar navigation includes services like EC2, VPC, Lambda, S3, RDS, IAM, Route 53, S3 Glacier, Storage Gateway, and DynamoDB. The main content area displays the 'Auto Scaling groups (1/1) info' section. A success message at the top states: 'GL-ASG, 1 Scaling policy created successfully. Group metrics collection is enabled.' Below this, a table lists the single Auto Scaling group 'GL-ASG'. The table columns include Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Available. The 'Desired capacity' is set to 2, and 'Min' and 'Max' are set to 1 and 10 respectively. The 'Available' status is 'us-east-1'. At the bottom of the table, there are tabs for Details, Activity, Automatic scaling, Instance management, Monitoring, and Instance refresh. The 'Details' tab is selected. Under 'Group details', the 'Desired capacity' is listed as 2, and the 'Auto Scaling group name' is GL-ASG. The 'Minimum capacity' is listed as 1, and the 'Date created' is Thu Aug 04 2022 15:31:51 GMT+0530 (India Standard Time).

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Available
GL-ASG	GL-instance Version Default	2	-	2	1	10	us-east-1

Group details

Desired capacity 2	Auto Scaling group name GL-ASG
Minimum capacity 1	Date created Thu Aug 04 2022 15:31:51 GMT+0530 (India Standard Time)

Connecting EC2 using Command Prompt



```
Command Prompt x Settings x + ▾ Priority-exclude packages based on name + arch
[root@ip-172-31-31-145 ec2-user]# yum install httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
→ Running transaction check
--> Package httpd.x86_64 0:2.4.54-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
--> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
--> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
--> Package httpd-filesystem.noarch 0:2.4.54-1.amzn2 will be installed
--> Package httpd-tools.x86_64 0:2.4.54-1.amzn2 will be installed
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
--> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed
--> Running transaction check
--> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

repo        ch      Version       Repository      Size   Package          Ar
repo        ch      Version       Repository      Size   Package          Ar
repo        ch      Version       Repository      Size   Package          Ar
httpd      x86_64  2.4.54-1.amzn2      amzn2-core    1.4 M  Installing:
Installing for dependencies:
apr         x86_64  1.7.0-9.amzn2      amzn2-core    122 k
apr-util    x86_64  1.6.1-5.amzn2.0.2  amzn2-core    99 k
apr-util-bdb x86_64  1.6.1-5.amzn2.0.2  amzn2-core    19 k
generic-logos- httpd.noarch 18.0.0-4.amzn2      amzn2-core    19 k
httpd-filesystem.noarch 2.4.54-1.amzn2      amzn2-core    24 k
httpd-tools   x86_64  2.4.54-1.amzn2      amzn2-core    88 k
mailcap     noarch  2.1.41-2.amzn2      amzn2-core    31 k
mod_http2   x86_64  1.15.19-1.amzn2.0.1  amzn2-core   149 k
```

Connecting EC2 using Command Prompt

```
Microsoft Windows [Version 10.0.21996.1]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sree>cd Downloads

C:\Users\sree\Downloads>ssh -i "capstoneproject.pem" ec2-user@ec2-54-226-128-176.compute-1.amazonaws.com
The authenticity of host 'ec2-54-226-128-176.compute-1.amazonaws.com (54.226.128.176)' can't be established.
ECDSA key fingerprint is SHA256:LtXeF13yUYse9dWyA2ACYMZgR27EgA75kAlFZnnz4qw.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-226-128-176.compute-1.amazonaws.com,54.226.128.176' (ECDSA) to the list of known hosts.

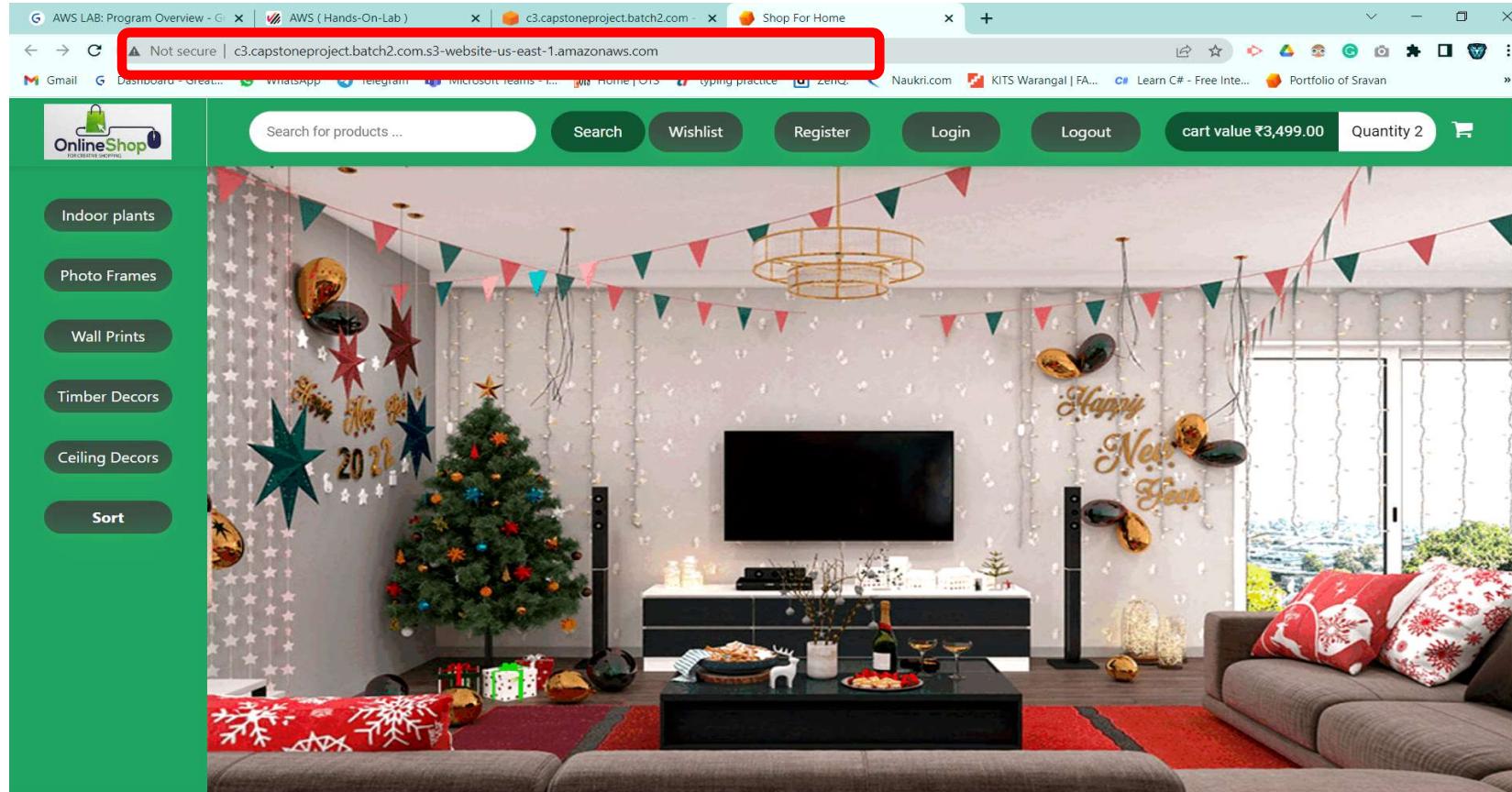
   _|_ / 
  -| (   /  Amazon Linux 2 AMI
   \_\_|_ | 

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-31-145 ~]$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-c num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user]
          [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-c num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] file ...
[ec2-user@ip-172-31-31-145 ~]$ sudo su
[root@ip-172-31-31-145 ec2-user]# yum
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
You need to give some command
Usage: yum [options] COMMAND

List of Commands:

check      Check for problems in the rpmdb
check-update  Check for available package updates
clean      Remove cached data
deplist    List a package's dependencies
distribution-synchronization Synchronize installed packages to the latest available versions
downgrade  downgrade a package
erase      Remove a package or packages from your system
fs         Acts on the filesystem data of the host, mainly for removing docs/languages for minimal hosts.
fssnapshot Creates filesystem snapshots, or lists/deletes current snapshots.
groups     Display, or use, the groups information
help       Display a helpful usage message
```

Aws Deployment



Git Repository

The screenshot shows a GitHub repository page for 'A-Sravan / C3CapstoneProjectBatch2'. The repository is private. The 'Code' tab is selected. The master branch has 2 branches and 0 tags. The repository contains files: backend/demo, frontend, BulkUpload1.csv, Capstone batch2 ppt.pptx, Database Schema.png, and data.sql.sql. A commit from 'A-Sravan capstoneproject' was made 3 hours ago. The repository has 0 stars, 1 watching, and 0 forks. There are no releases or packages published.

A-Sravan/C3CapstoneProjectBatch2

github.com/A-Sravan/C3CapstoneProjectBatch2/tree/master

Search or jump to... Pull requests Issues Marketplace Explore

A-Sravan / C3CapstoneProjectBatch2 Private

Code Issues Pull requests Actions Projects Security Insights Settings

master 2 branches 0 tags

This branch is 1 commit ahead, 1 commit behind main.

Contribute

A-Sravan capstoneproject

backend/demo	capstoneproject	3 hours ago
frontend	capstoneproject	3 hours ago
BulkUpload1.csv	capstoneproject	3 hours ago
Capstone batch2 ppt.pptx	capstoneproject	3 hours ago
Database Schema.png	capstoneproject	3 hours ago
data.sql.sql	capstoneproject	3 hours ago

a508dd1 3 hours ago 1 commit

Add a README with an overview of your project. Add a README

About

No description, website, or topics provided.

0 stars 1 watching 0 forks

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Conclusion

- Technology has made significant progress over the years to provide consumers a better online shopping experience and will continue to do so for years to come.
- The Application has been developed with free of errors and at the same time it is efficient and less time consuming.
- The purpose of this project was to develop a web application for purchasing items from a shop.
- With most of the E-retailers selling branded goods and having flexible return policies they are being well trusted by the users.
- Online Shopping is better for both consumer and sellers.

THANK YOU

