

EduBridge



A Project

Report On

Grocery Shopping

By

Divya M Palankar

B.E(2019)

Batch: 2021 – 5763

Center: Bengaluru

Under the Guidance of,

Nilesh Mokashi.

Technical Trainer

EduBridge

(School of coding)

Introduction:

Our project explains about the Ecommerce Shopping Website name Grocery Shopping. This project mainly explains the various actions related to shop online. You can search various products and shuffle according to category.

Our Project Include**Modules:-**

- Admin Module
- User Module

I have developed this Application in **Java, JSP, Servlets, Hibernate and MySQL**. It's a web-based projects so I have used **HTML, CSS, JavaScript and Bootstrap also**.

The main feature of the project is to make shopping easier. You can access this website from anywhere and everywhere.

In Admin module, admin can add the Category and Products in the specific category. He can also view the total number of users at present in the website. And category and product details also. Admin can add delete update the product accordingly.

In User module, user can access the website and buy products and products to productList and buy and shuffle by category. New user can register to shop.

Software Requirements:

Front end: Java/J2EE technologies (Servlet, JSP), HTML, CSS, JavaScript, Bootstrap, Hibernate.

Back end: MySQL workbench 8.0.23CE.

Middleware/Server: Apache Tomcat v7.0.32. IDE: Eclipse Mars IDE for Java EE Developers

Browser: Best result on Google Chrome

Operating System: Window 8.1.

Data Dictionary:

- Create Database springboot;
- Hibernate: @Entity
@Table(name = "contact_info")
public class Contact implements Serializable {
 @Id
 @GeneratedValue(strategy = GenerationType.*AUTO*)
 private int id;
 @Column
 private String name;
 @Column
 private String email;
 @Column
 private String contact_no;
 @Column
 private String subject;
 @Column
 private String message;
- Hibernate: @Entity
 @Table(name = "payment_info")
 public class Payment implements Serializable {
 @Id
 @GeneratedValue(strategy = GenerationType.*AUTO*)
 private int id;
 @Column
 private String name;
 @Column
 private String email;
 @Column
 private String contact_no;
 @Column
 private String address;
 @Column
 private String city;

```

@Column
private String pin;
@Column
private String payment_method;
@Column
private String card_type;
@Column
private String account_holdername;
@Column
private String cvv;
@Column
private String card_no;
@Column
private String exp_date;

```

- Hibernate: @Entity

```

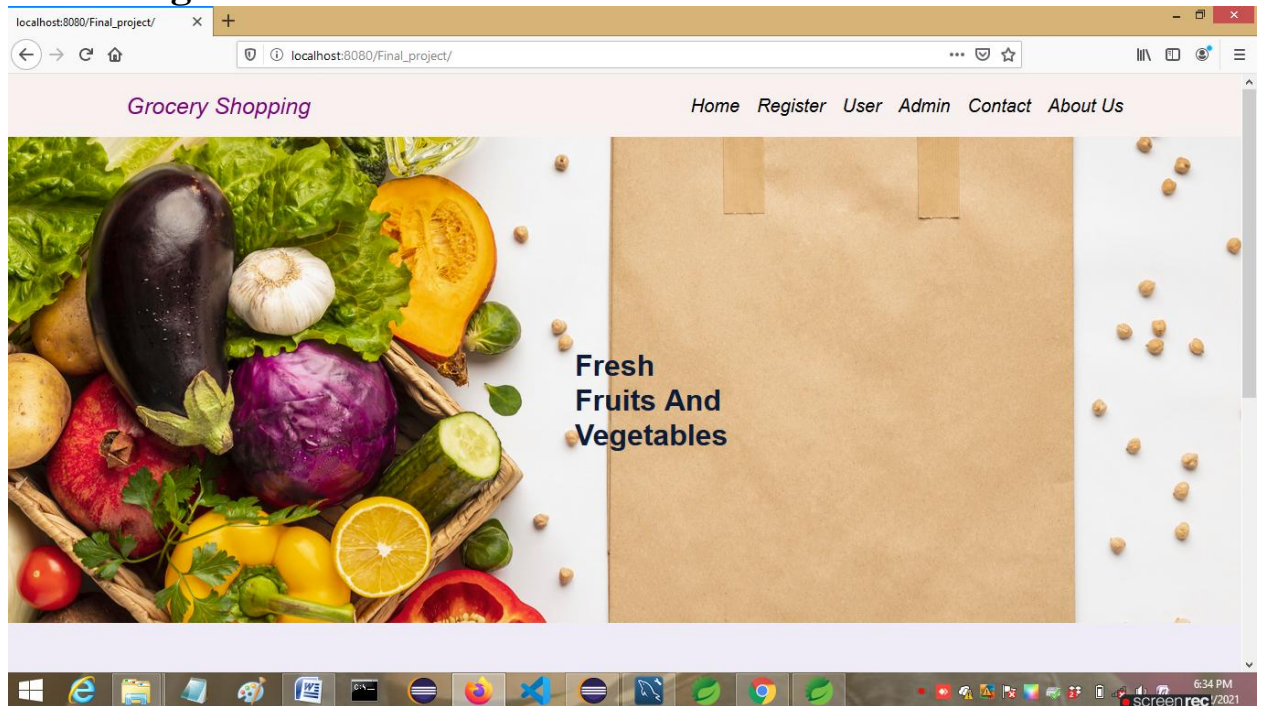
        @Table(name = "product_info")
        public class Product implements Serializable {
            @Id
            @GeneratedValue(strategy = GenerationType.AUTO)
            private int id;
            @Column
            private String product_name;
            @Column
            private String description;
            @Column
            private int quantity;
            @Column
            private String category;
            @Column
            private String cost;

```

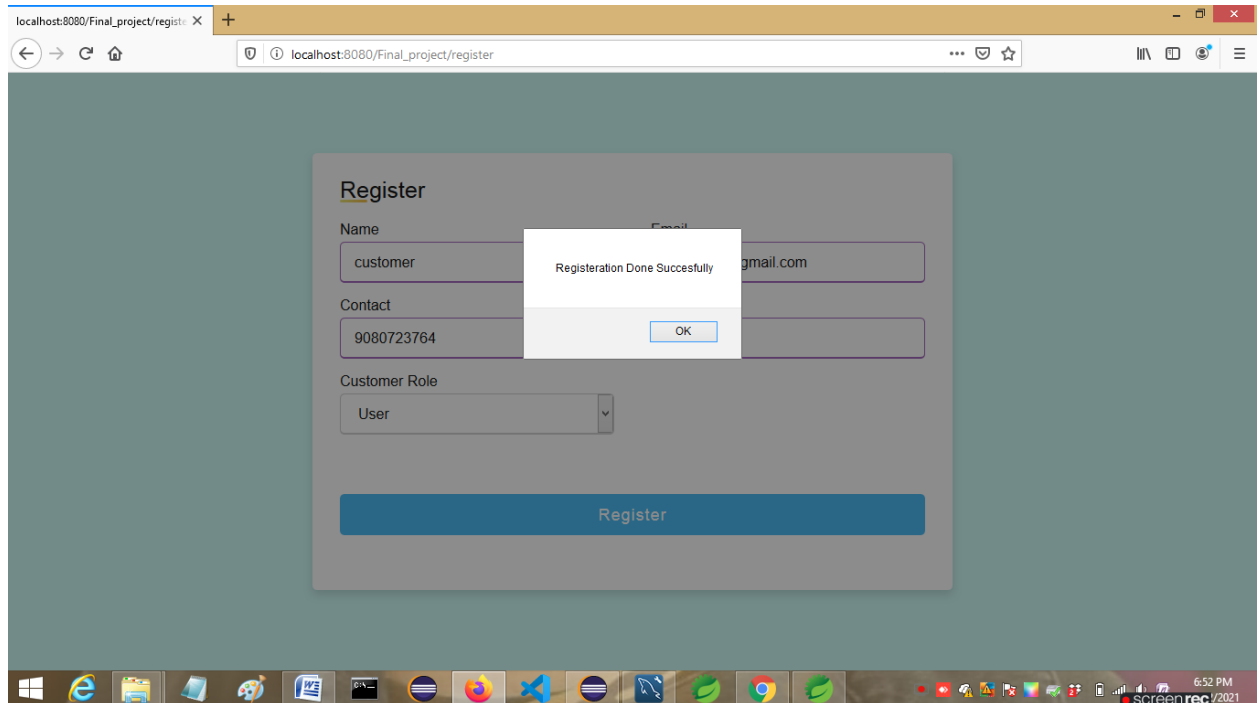
Hibernate

Screenshots:-

Home Page:



Register Page:



Data Added in Database Successfully:

The screenshot shows the MySQL Workbench interface. The left sidebar contains the 'MANAGEMENT' and 'INSTANCE' sections. The main window displays a SQL query editor with the following code:

```
68
69 • create database springboot;
70 • use springboot;
71 • show tables;
72 • select * from contact_info;
73 • select * from register_info
74
```

The 'Result Grid' shows the output of the queries:

id	contact	email	name	password	role
1	97865432	divya@gmail.com	divya	1234	user
2	98765432	divya@gmail.com	divya	password	user
3	9080723764	customer@gmail.com	customer	123456	user

The 'Output' section shows the execution log:

#	Time	Action	Message	Duration / Fetch
3	10:11:58	show tables	5 row(s) returned	0.031 sec / 0.000 sec
4	19:52:06	use springboot	0 row(s) affected	0.015 sec
5	19:52:12	show tables	5 row(s) returned	0.016 sec / 0.000 sec
6	19:52:37	select * from contact_info LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
7	19:53:43	show tables	5 row(s) returned	0.000 sec / 0.000 sec
8	19:54:02	select * from register_info LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

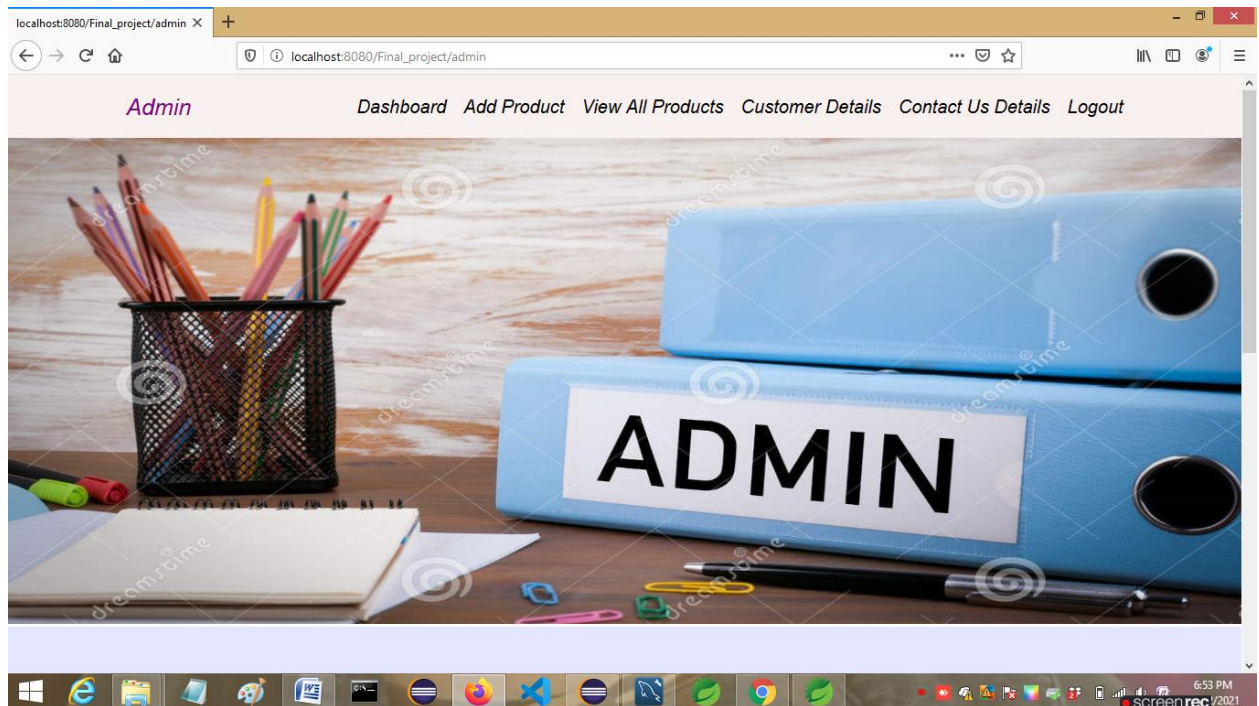
About Us Page:

The screenshot shows a web browser displaying the 'About Us' page of a web application. The page has a navigation bar with links: Home, Register, User, Admin, Contact, and About Us. The main content area is titled 'About Us' and contains the following text:

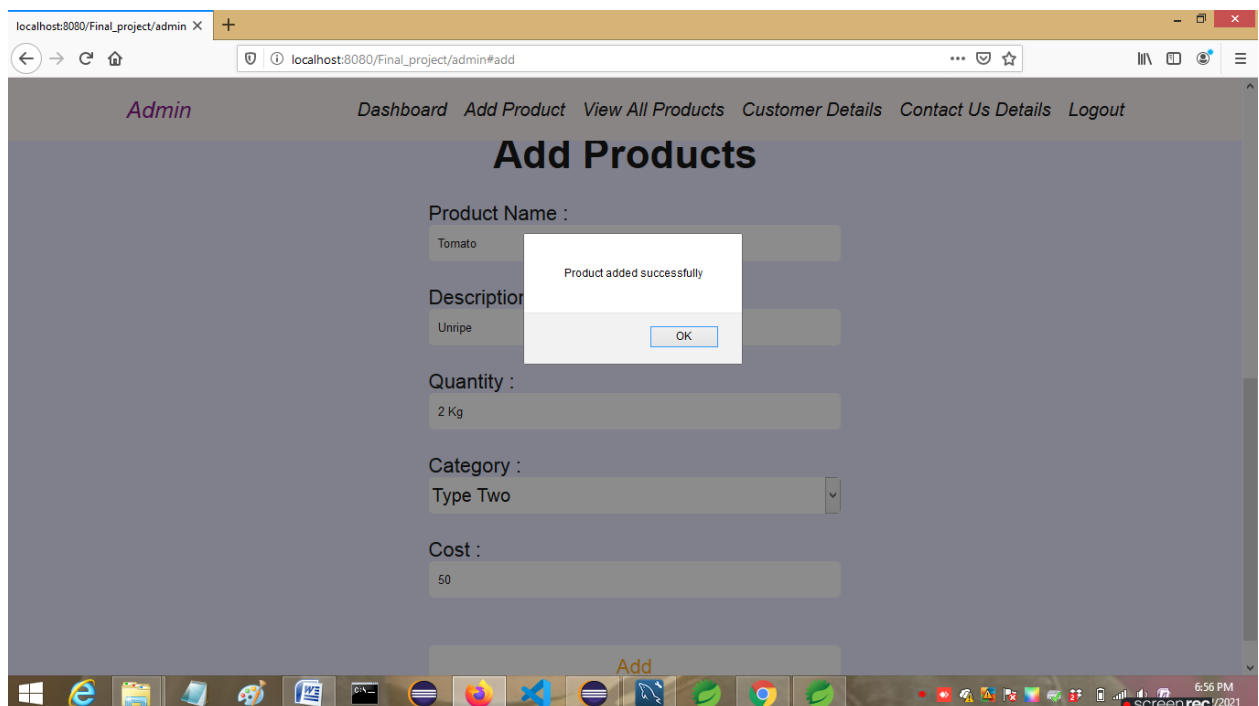
Did You Ever Imagine That The Freshest Of Fruits And Vegetables, Top Quality Pulses And Food Grains, Dairy Products And Hundreds Of Branded Items Could Be Handpicked And Delivered To Your Home, All At The Click Of A Button? India's First Comprehensive Online Megastore, Bigbasket.com, Brings A Whopping 20000+ Products With More Than 1000 Brands, To Over 4 Million Happy Customers. From Household Cleaning Products To Beauty And Makeup, Bigbasket Has Everything You Need For Your Daily Needs. Bigbasket.com Is Convenience Personified We've Taken Away All The Stress Associated With Shopping For Daily Essentials, And You Can Now Order All Your Household Products And Even Buy Groceries Online Without Travelling Long Distances Or Standing In Serpentine Queues. Add To This The Convenience Of Finding All Your Requirements At One Single Source, Along With Great Savings, And You Will Realize That Bigbasket-India's Largest Online Supermarket, Has Revolutionized The Way India Shops For Groceries. Online Grocery Shopping Has Never Been Easier. Need Things Fresh? Whether It's Fruits And Vegetables Or Dairy And Meat, We Have This Covered As Well! Get Fresh Eggs, Meat, Fish And More Online At Your Convenience. Hassle-free Home Delivery Options

We Have This Covered As Well! Get Fresh Eggs, Meat, Fish And More Online At Your Convenience. Hassle-free Home Delivery Options.

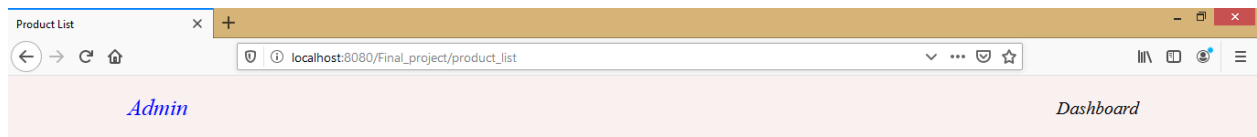
Admin Page:



Add Product Page:



Product List Page:

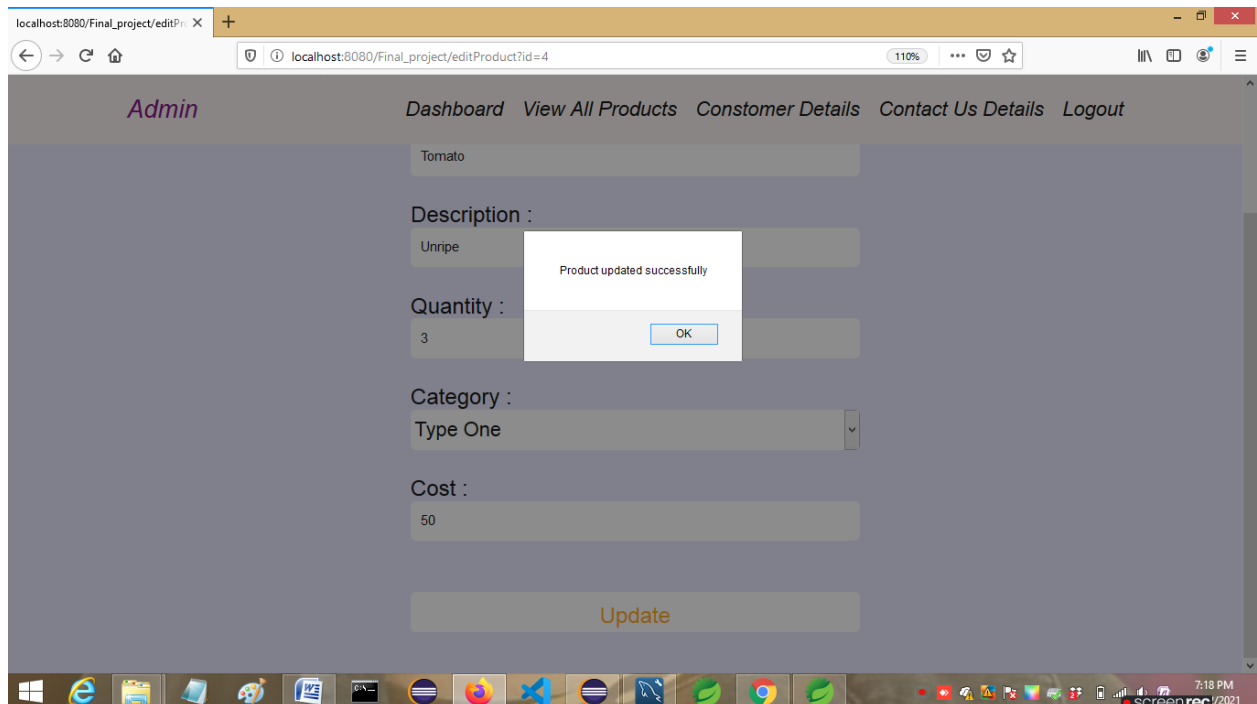


PRODUCT LIST

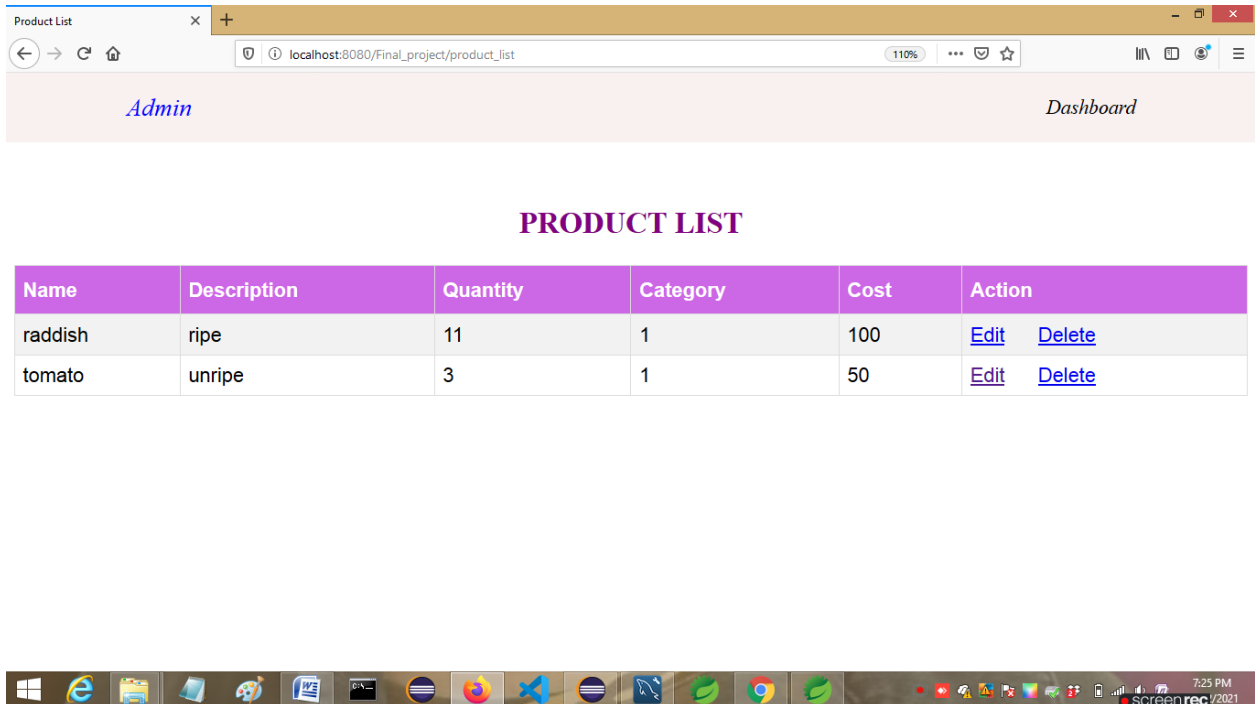
Name	Description	Quantity	Category	Cost	Action
raddish	ripe	11	1	100	Edit Delete
tomato	unripe	2	1	50	Edit Delete



Update Product Page:



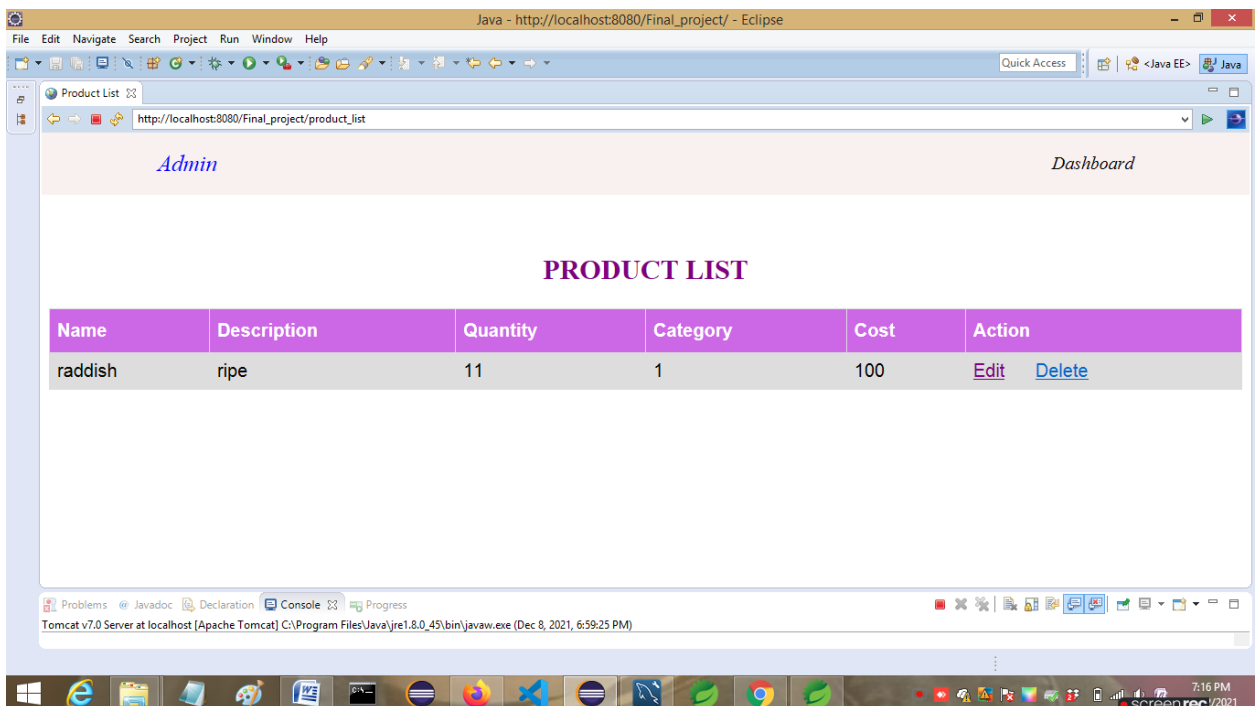
Updated Product List:



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/Final_project/product_list'. The page features a header with 'Admin' on the left and 'Dashboard' on the right. The main content area is titled 'PRODUCT LIST' and contains a table with the following data:

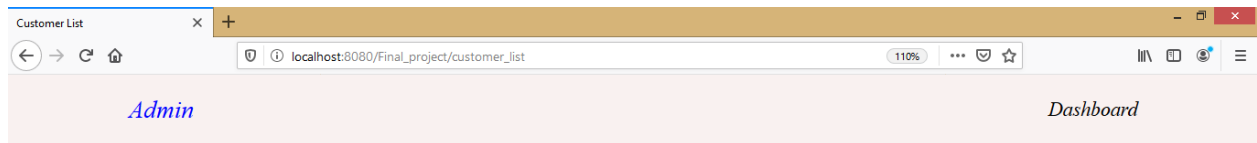
Name	Description	Quantity	Category	Cost	Action
raddish	ripe	11	1	100	Edit Delete
tomato	unripe	3	1	50	Edit Delete

Delete Product List:



The screenshot shows the Eclipse IDE with a browser window open at 'http://localhost:8080/Final_project/product_list'. The browser displays the same 'PRODUCT LIST' table as the previous screenshot. The IDE's console at the bottom shows the message: 'Tomcat v7.0 Server at localhost [Apache Tomcat] C:\Program Files\Java\jre1.8.0_45\bin\javaw.exe (Dec 8, 2021, 6:59:25 PM)'.

Customer Details:

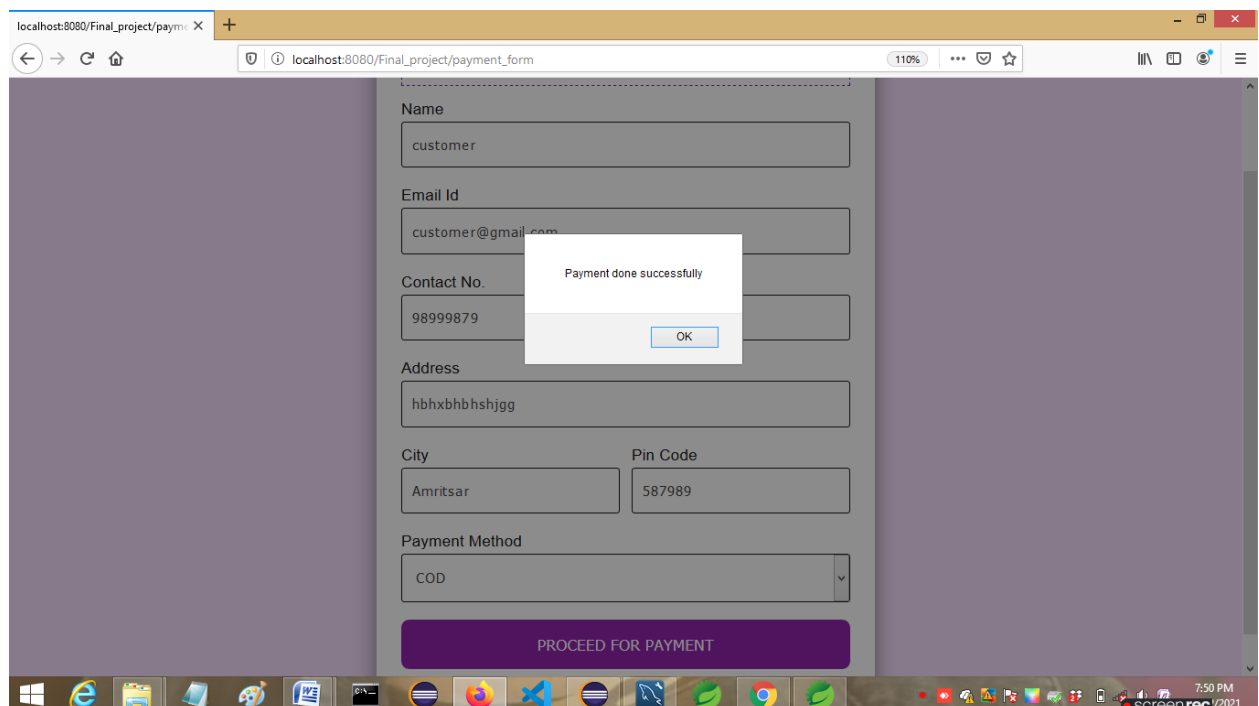


CUSTOMER DETAILS

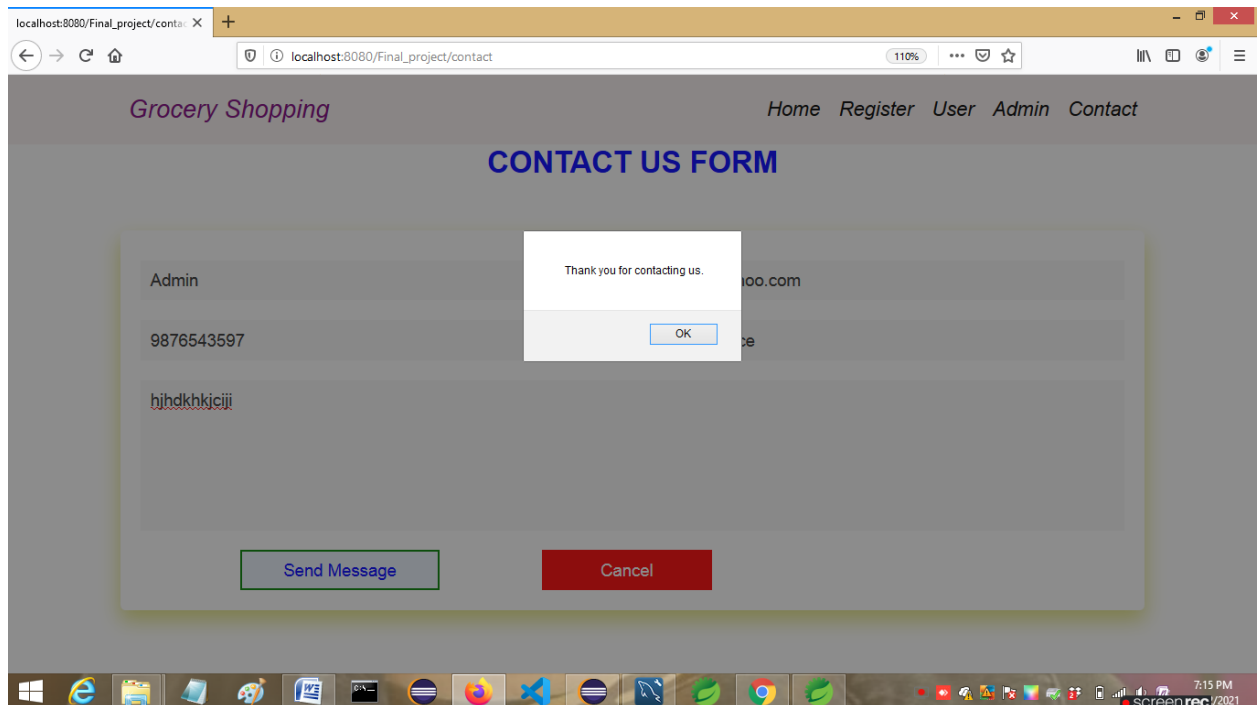
Name	Contact	Email	Password	Action
divya	97865432	divya@gmail.com	1234	Delete
divya	98765432	divya@gmail.com	password	Delete
customer	9080723764	customer@gmail.com	123456	Delete



Payment Page:



Contact Us Page:



Thank you