Prototype-Midterm

LITHAN

EDU CLaaS® DIGITAL ACCELERATOR

Student Name/ID Number:	Chathushi Jayarathna
Academic Year:	2022/23
Unit Assessor:	MS. Aravinder Kaur
Project Title:	CPL-Midterm
Issue Date:	04/03/2023
Submission Date:	16/03/2023
Internal Verifier Name:	
Date:	16/03/2023

Learner declaration

I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.

Student signature: Chathushi Jayarathna Date: 16/03/2023

01) A summary of the project's objectives, scope, and requirements, as outlined in the project brief or proposal.

Aceadora Tech has been engaged as a website developer for the development. As a project manager knowledge of project-making plans reporting is critical to communicating relevant and timely facts to key stakeholders. project making plans reviews deliver project managers more manipulation over their task and reduce the time spent balancing resource requirements with resources available. It additionally enables ensuring that the necessary assets and abilities are to be had on the right time and within the right location, enabling extra task success. data is fundamental to challenge manage and management. without the potential to communicate relevant and timely facts to key stakeholders there's no manipulation over the assignment and, ultimately, it'll fail. To attain challenge achievement, it is vital to accumulate and percentage statistics correctly, and the satisfactory manner to do that is the use of reviews. reports are created for the duration of the challenge lifecycle, and, whilst many view them as time-ingesting, exhausting, and painful, they may be one of the most critical assets a challenge manager has at their disposal.

1.1 Scope

The scope of this project includes the following:

- The implementation of a new software system
- Data migration from the old system to the new system
- User training and support
- System maintenance and upgrades

- Explain principles of problem management across its lifecycle.

- Use various tools, process and technologies to facilitate problem identification, investigation, analysis & resolution
- Explain various steps to investigate & diagnose problems.
- Prioritize & Categorize change requests.
- Prepare a solution to address the root cause of the problem.
- Document & monitor the problems
- Explain best practices in documenting problems

Users should be able to perform the following functions in the portal

- Register in the Portal
- Login to the Portal
- Post a product for Sale along with a Picture Upload
- Deactivate an Existing product
- Update their Profile after logging in.
- Add items to the cart
- View cart items

Administrators should be able to perform the following functions in the portal

- Register in the Portal
- Login to the portal
- View the List of Registered Users/partners
- Mark a User as Administrator
- Activate / Deactivate a product post
- Update their profile
- Transact the sales if the price is right

Both Users & Administrator

- Visit the Home Page
- View product Listing
- Search for a product by Name, Brand, Serial No & Price Range
- About Us Page
- Contact Us Page
- Terms & Conditions

1.2 Project out of Scope

- Upload the comments alternative and rating system to get the user's thoughts.
- Develop FAQ pages to remedy the issues frequently requested by the customers.

1.3 Functional Requirement

The following pages will be developed in the project.

- Landing Page.
- About Us Section.
- Contact Us Section
- Login Page.
- Registration Page.
- Thank You Page.
- Homepage.
- Search Product Function.
- Post Product Page.
- Product Detail Page.
- Update Profile Page.
- List Users Page.
- List Product Page.
- Add to cart page

1.4 Non-functional requirements

✓ **Safety requirements**

The database has the personal facts of users, due to that it secures with a robust password. also, the passwords of users are encrypted and stored in a database

✓ **Performance attributes**

- **Availability -** The network Portal is operational 24 hours a day

- **Correctness -** The consequences users are seeking out in a search function much like the results users need t get. additionally, whilst the user logged into the community portal correct user web page is showing

- **Maintainability -** The community portal is frequently monitored and maintained through the Admin

- **Usability -** Jumpstart network portal can include a bulky quantity of users and has a huge capacity to stock user records

1.5 Technical Environment Requirements

✓ **Hardware Requirements**

- Processor: Inter(R) Core (TM) i5
- RAM: 12GB
- 1TB Disk Space
- MySQL Server 8.0.30
- Apache Tomcat 9

✓ **Software Requirements**

- Development Environment - JDK 11
- Framework - Spring Boot
- Database Server - MySQL Server 8.0.30
- Firewall Settings - Port 9001-Http, Port 3306 - Database

✓ **System Integration Requirements**

- JDBC-JDBC Connector 8.0.11
- Spring security Dependency

✓ **Portability Requirements**

- Multi OS- Windows, Linux, and Mac
- Multiple Browsers - Chrome, Safari, Microsoft Edge
- Multiple Devices - Desktop, Tablet, Mobile

✓ **Maintainability Requirements**

  - Back up the database every 6 hours.
  - Vulnerability check – periodical scanning.
  - Each request should be processed within 10 seconds.

✓ **Performance Requirements**

  - Concurrent number of users- 100
  - Loading time – 15secs

1.6 Objectives

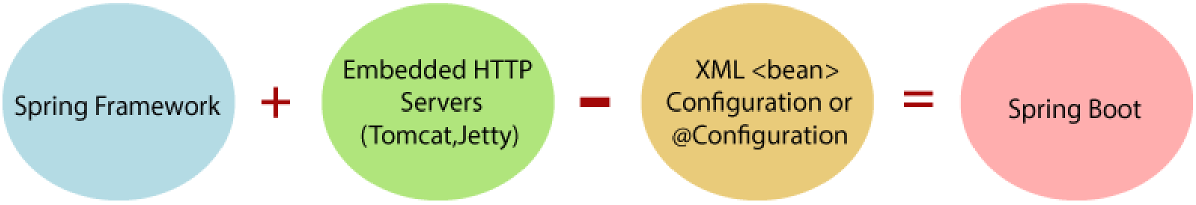
The objectives of this project are as follows:

- The grant objective of this project is considered to be the development of skills required for project planning, implementation, testing, and documentation.
- Identify and analyze business processes
- Creating a business process session.
  It defines the roles and obligations of the project management team participants.
- It guarantees that the project management group works in step with the business targets.
- It checks the feasibility of the schedule and consumer necessities.
- It determines project constraints.
- To improve the accuracy and efficiency of the customer data management process
- To increase the speed of customer data retrieval
- To reduce the time spent on manual data entry and management
- To improve the quality of customer service provided to customers

02) A brief description of the technologies, programming languages, and frameworks used to develop the website.

2.1 Front-End & Back-end - Spring boot.

Spring Boot is a Framework from "The Spring team" to ease the bootstrapping and development of recent Spring programs. It gives defaults for code and annotation configuration to short-begin new Spring projects within no time. It follows the "Opinionated Defaults Configuration" approach to avoid a lot of boilerplate code and configuration to enhance the development, Unit check, and Integration test process


Spring Framework
+
Embedded HTTP Servers (Tomcat,Jetty)
-
XML <bean> Configuration or @Configuration
=
Spring Boot

2.2 Database: MySQL

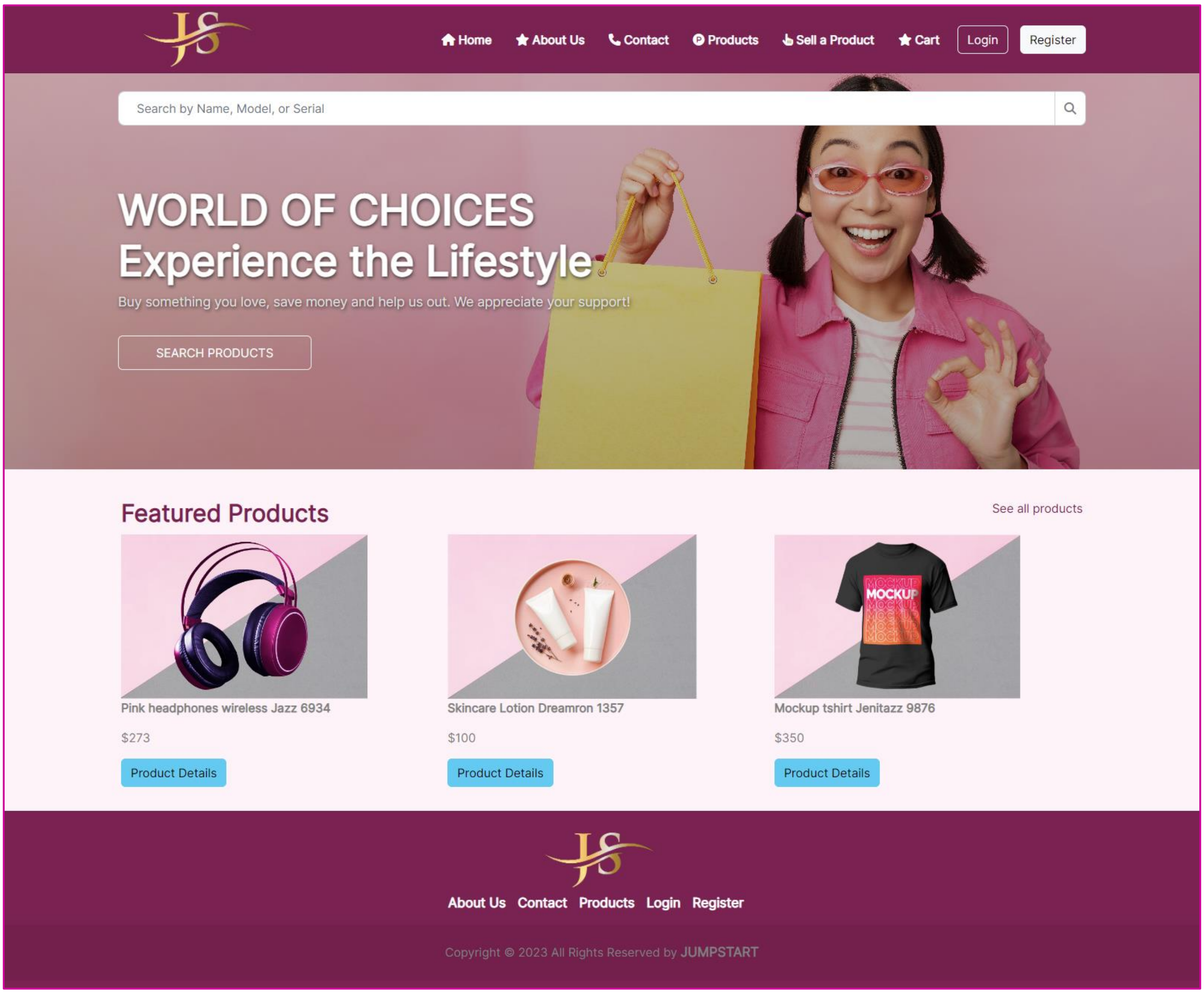
MySQL can be a social database manipulation framework (RDBMS) created by means of using Prophet that is based on prepared inquiry dialect. A database can be an organized series of facts. it is able to be something from a sincere buying list to a picture show or a place to preserve the limitless sums of data a company prepares. In specific, a social database can be complicated to keep collecting data and organizing it concurring to the social display. in this monitor, tables contain of traces and columns, and connections among statistics components all take after a strict coherent shape. An RDBMS is basically a set of software devices implemented to absolutely execute, oversee, and inquire this sort of database.

2.3 IDEs: Spring Tool Suite

Spring Tool Suite or STS is a java IDE designed for developing Spring-primarily based absolutely corporation packages. It’s a long way much less hard, faster, and handier. And most importantly it's miles primarily based totally on Eclipse IDE. STS is unfastened, open source, and powered by VMware. Spring equipment four is the following technology of spring equipment for the favorite coding surroundings.

03) Design and other design assets that illustrate the website's layout, color scheme, typography, and overall user interface design.

Landing Page



Home About Us Contact Products Sell a Product Cart Login Register

Search by Name, Model, or Serial

WORLD OF CHOICES
Experience the Lifestyle

Buy something you love, save money and help us out. We appreciate your support!

SEARCH PRODUCTS

Featured Products

See all products

Pink headphones wireless Jazz 6934
$273
Product Details

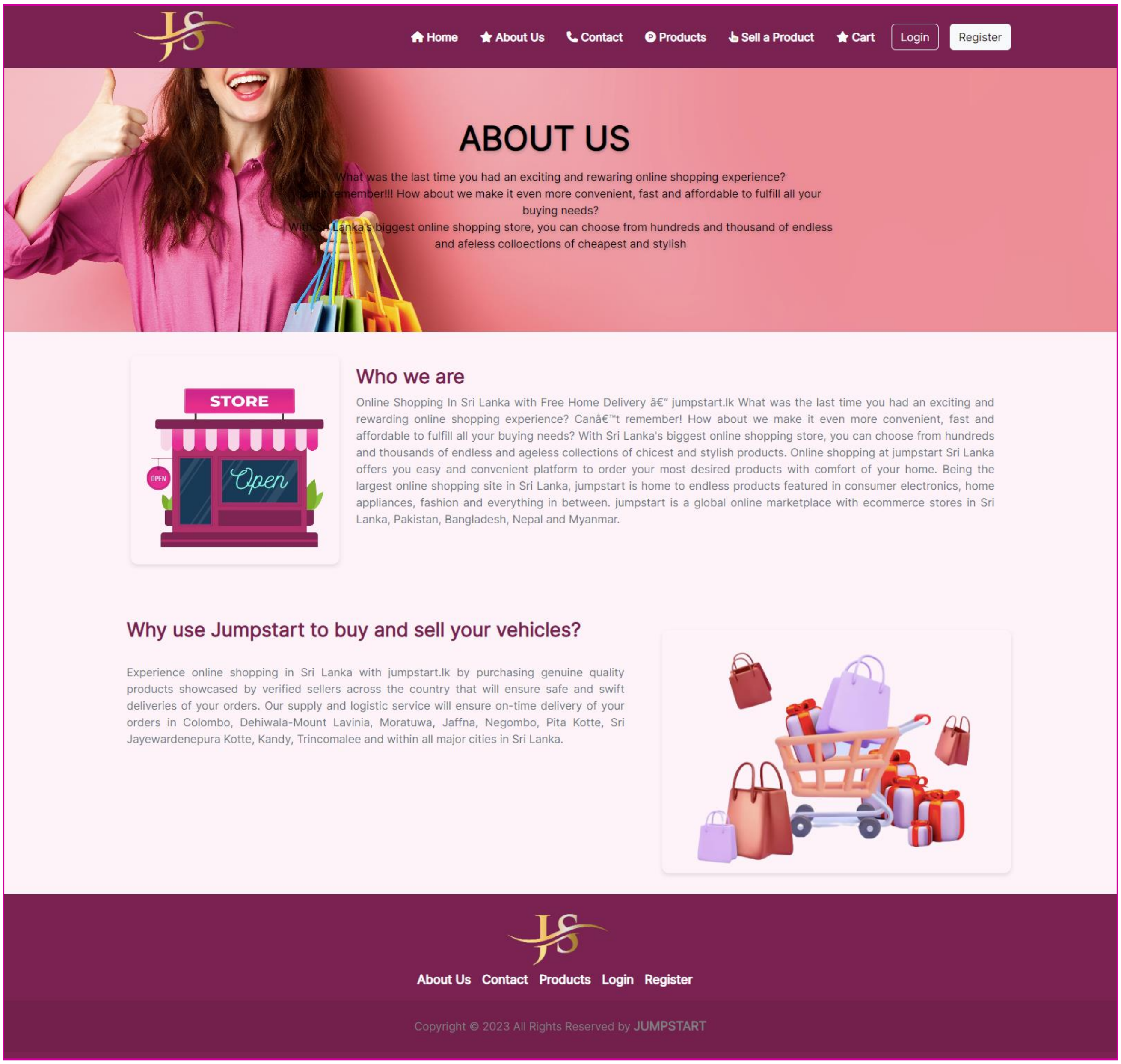
Skincare Lotion Dreamron 1357
$100
Product Details

Mockup tshirt Jenitazz 9876
$350
Product Details

About Us Contact Products Login Register

Copyright © 2023 All Rights Reserved by JUMPSTART

About Us Page



ABOUT US

What was the last time you had an exciting and rewaring online shopping experience?
Can't remember!!! How about we make it even more convenient, fast and affordable to fulfill all your buying needs?
With Sri Lanka's biggest online shopping store, you can choose from hundreds and thousand of endless and afeless colloections of cheapest and stylish

Who we are

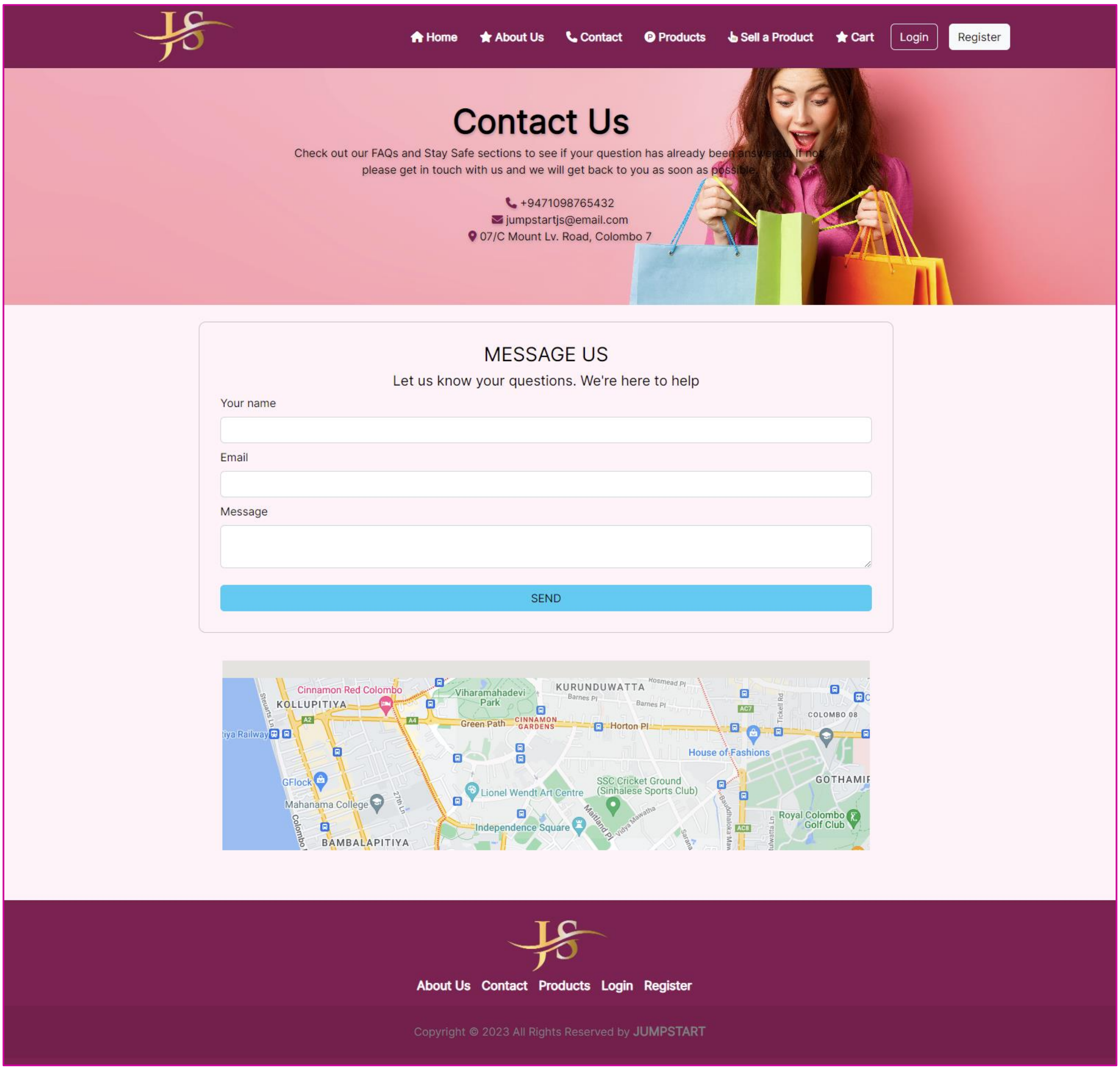
Online Shopping In Sri Lanka with Free Home Delivery â€“ jumpstart.lk What was the last time you had an exciting and rewarding online shopping experience? Canâ€™t remember! How about we make it even more convenient, fast and affordable to fulfill all your buying needs? With Sri Lanka's biggest online shopping store, you can choose from hundreds and thousands of endless and ageless collections of chicest and stylish products. Online shopping at jumpstart Sri Lanka offers you easy and convenient platform to order your most desired products with comfort of your home. Being the largest online shopping site in Sri Lanka, jumpstart is home to endless products featured in consumer electronics, home appliances, fashion and everything in between. jumpstart is a global online marketplace with ecommerce stores in Sri Lanka, Pakistan, Bangladesh, Nepal and Myanmar.

Why use Jumpstart to buy and sell your vehicles?

Experience online shopping in Sri Lanka with jumpstart.lk by purchasing genuine quality products showcased by verified sellers across the country that will ensure safe and swift deliveries of your orders. Our supply and logistic service will ensure on-time delivery of your orders in Colombo, Dehiwala-Mount Lavinia, Moratuwa, Jaffna, Negombo, Pita Kotte, Sri Jayewardenepura Kotte, Kandy, Trincomalee and within all major cities in Sri Lanka.

Copyright © 2023 All Rights Reserved by JUMPSTART

Contact Us



Contact Us

Check out our FAQs and Stay Safe sections to see if your question has already been answered. If not, please get in touch with us and we will get back to you as soon as possible.

☎ +9471098765432
✉ jumpstartjs@email.com
07/C Mount Lv. Road, Colombo 7

MESSAGE US

Let us know your questions. We're here to help

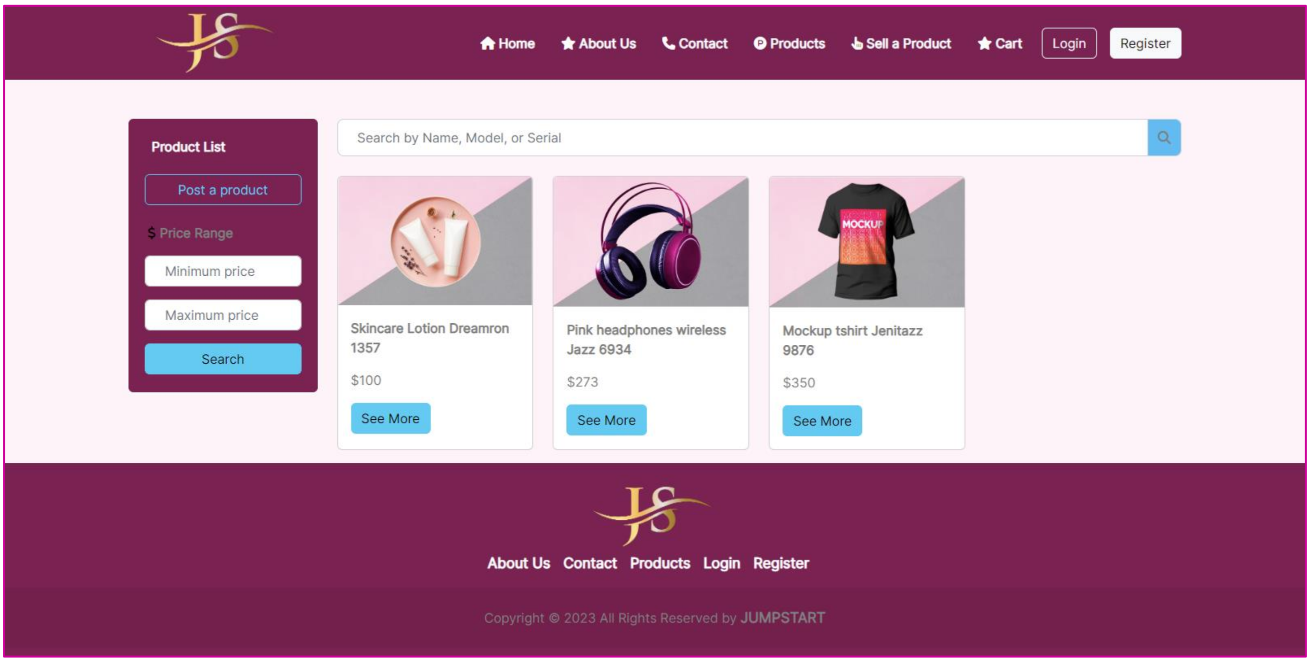
Your name

Email

Message

Copyright © 2023 All Rights Reserved by JUMPSTART

Products Page



Skincare Lotion Dreamron 1357

$100

Pink headphones wireless Jazz 6934

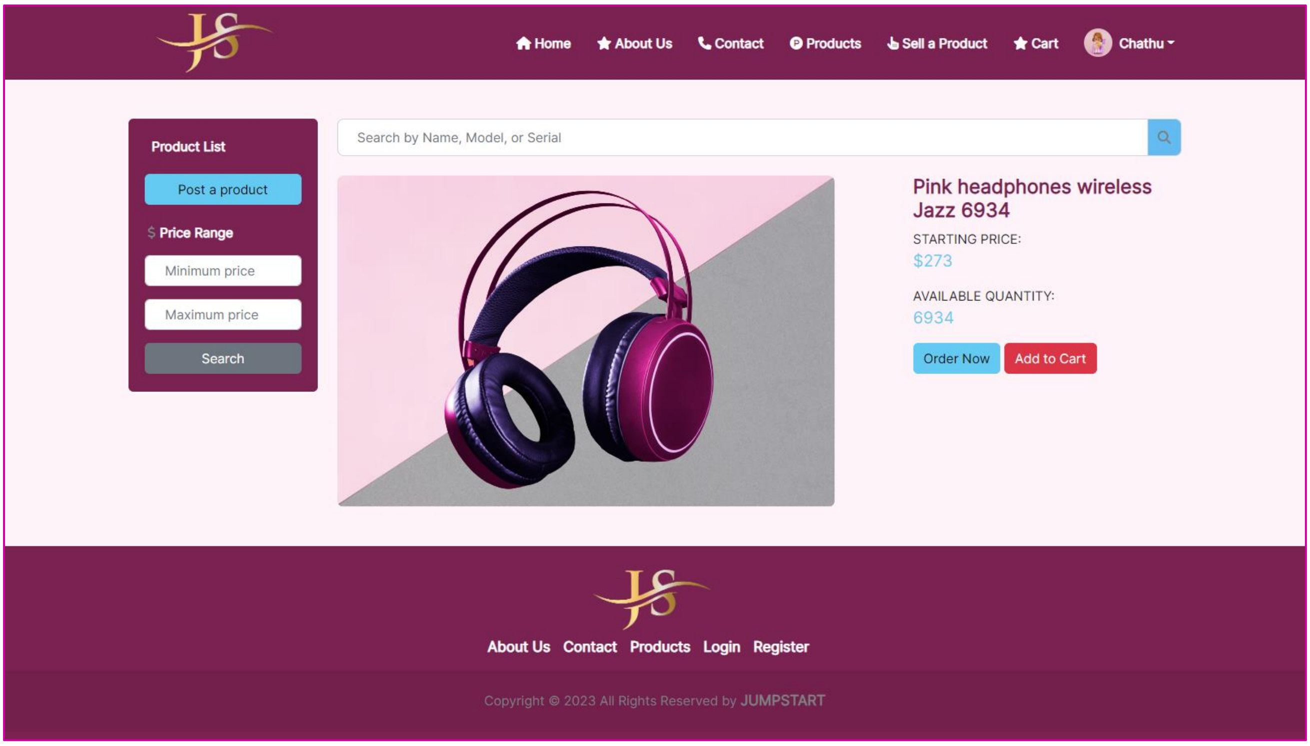
$273

Mockup tshirt Jenitazz 9876

$350

Copyright © 2023 All Rights Reserved by JUMPSTART

View Product



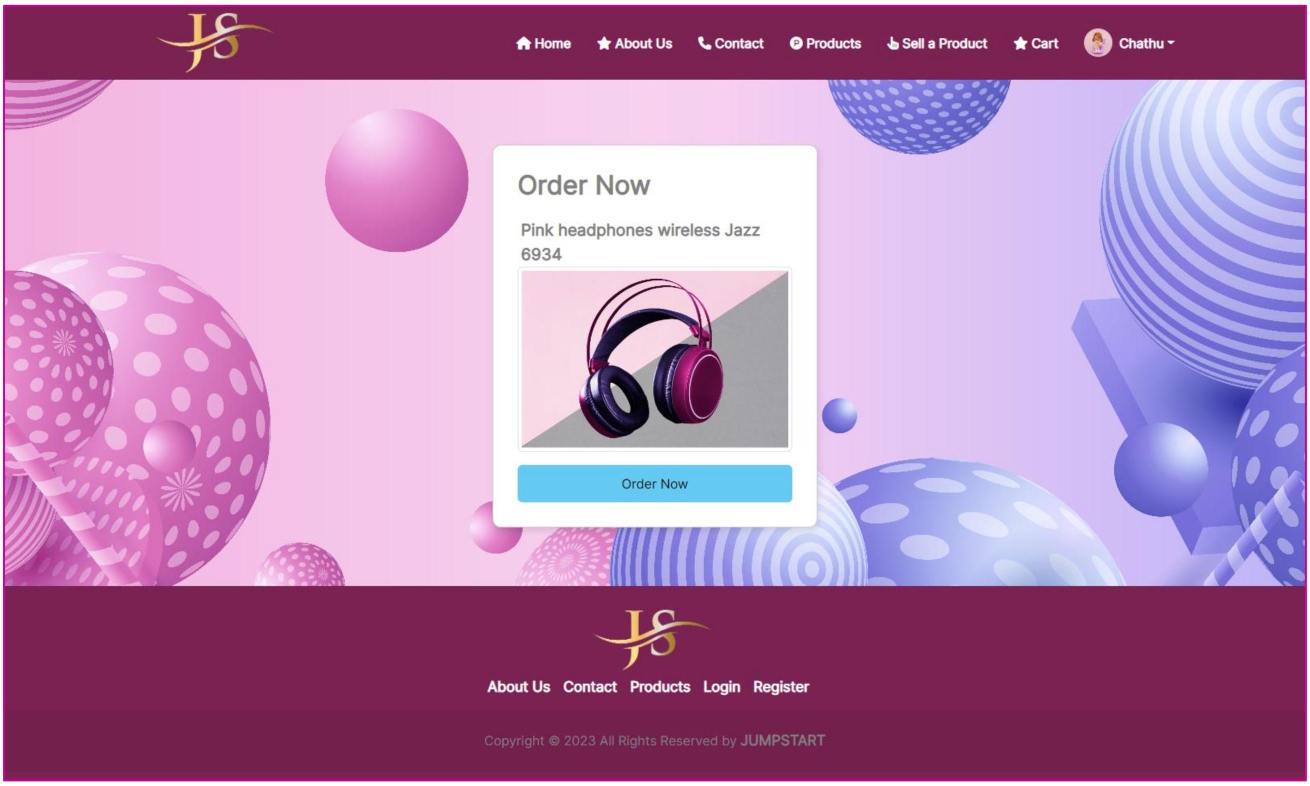
Pink headphones wireless Jazz 6934

STARTING PRICE:
$273

AVAILABLE QUANTITY:
6934

Copyright © 2023 All Rights Reserved by JUMPSTART

Order Product

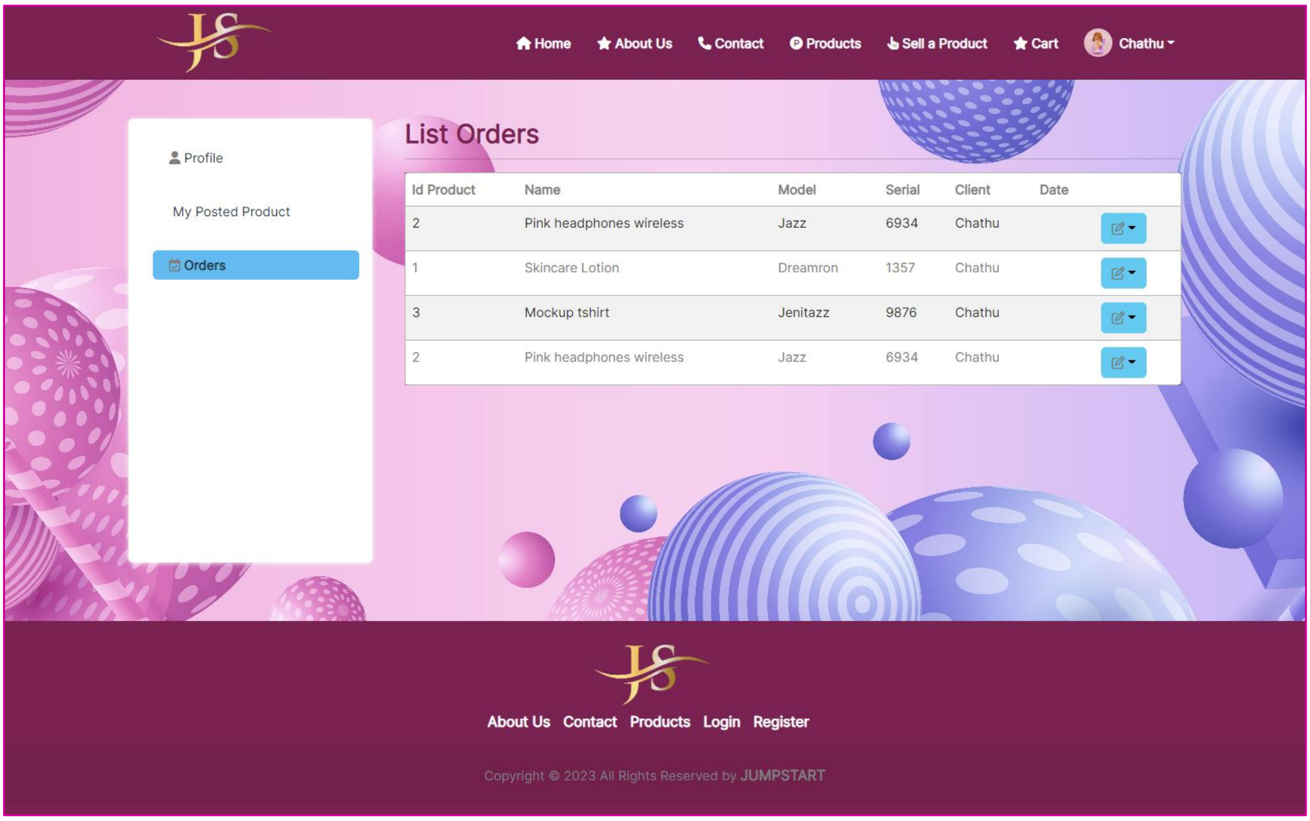

Home
About Us
Contact
Products
Sell a Product
Cart
Chathu

Order Now

Pink headphones wireless Jazz 6934

Order Now
About Us Contact Products Login Register
Copyright © 2023 All Rights Reserved by JUMPSTART

Order List

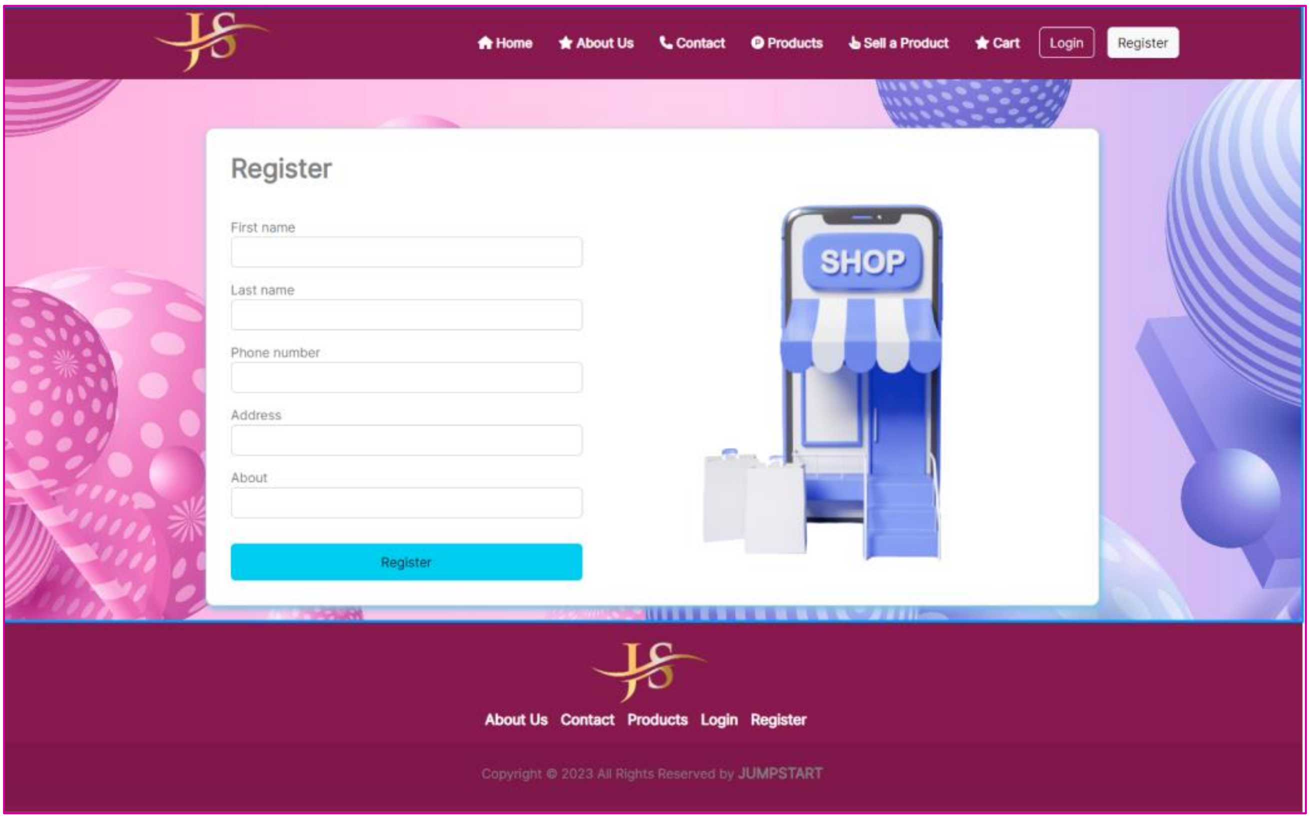

Home
About Us
Contact
Products
Sell a Product
Cart
Chathu
Profile
My Posted Product
Orders

List Orders

Id Product	Name	Model	Serial	Client	Date	
2	Pink headphones wireless	Jazz	6934	Chathu		
1	Skincare Lotion	Dreamron	1357	Chathu		
3	Mockup tshirt	Jenitazz	9876	Chathu		
2	Pink headphones wireless	Jazz	6934	Chathu		

About Us Contact Products Login Register
Copyright © 2023 All Rights Reserved by JUMPSTART

Registration



Register

First name

Last name

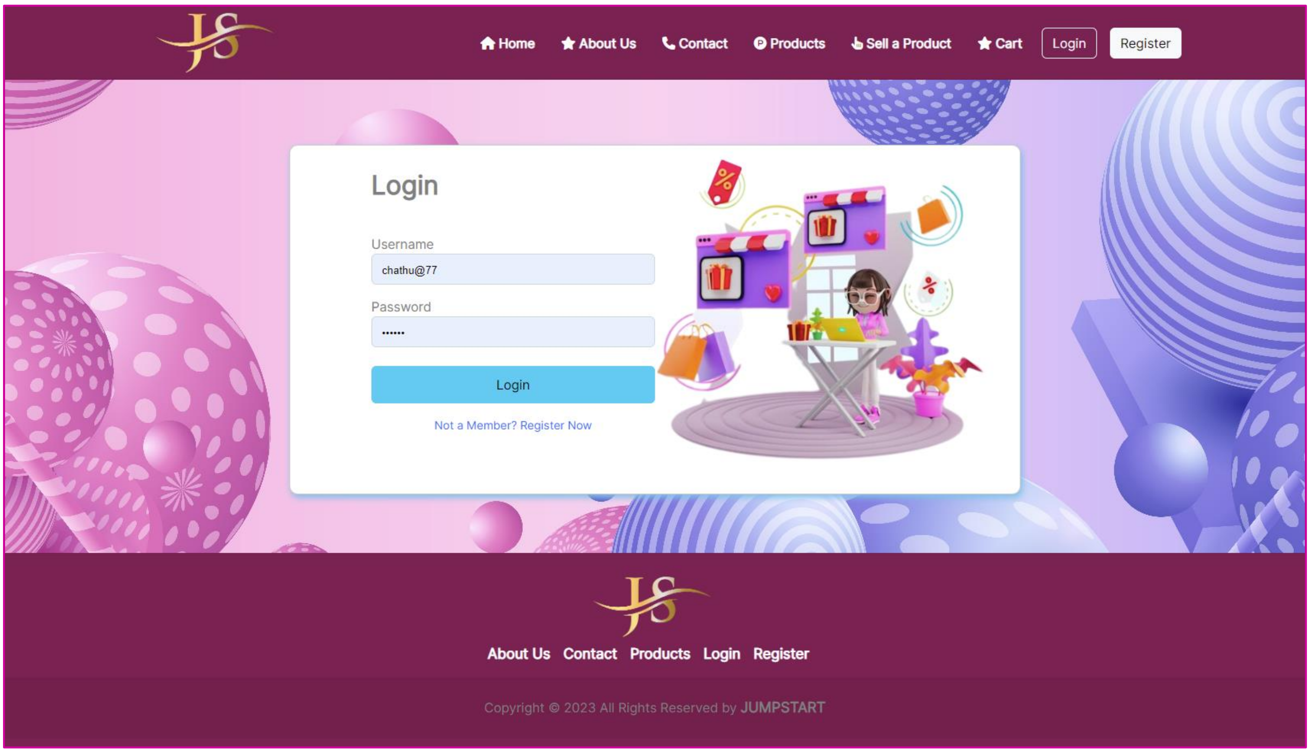
Phone number

Address

About

Copyright © 2023 All Rights Reserved by JUMPSTART

Login


Home
About Us
Contact
Products
Sell a Product
Cart
Login
Register

Login

Username

chathu@77

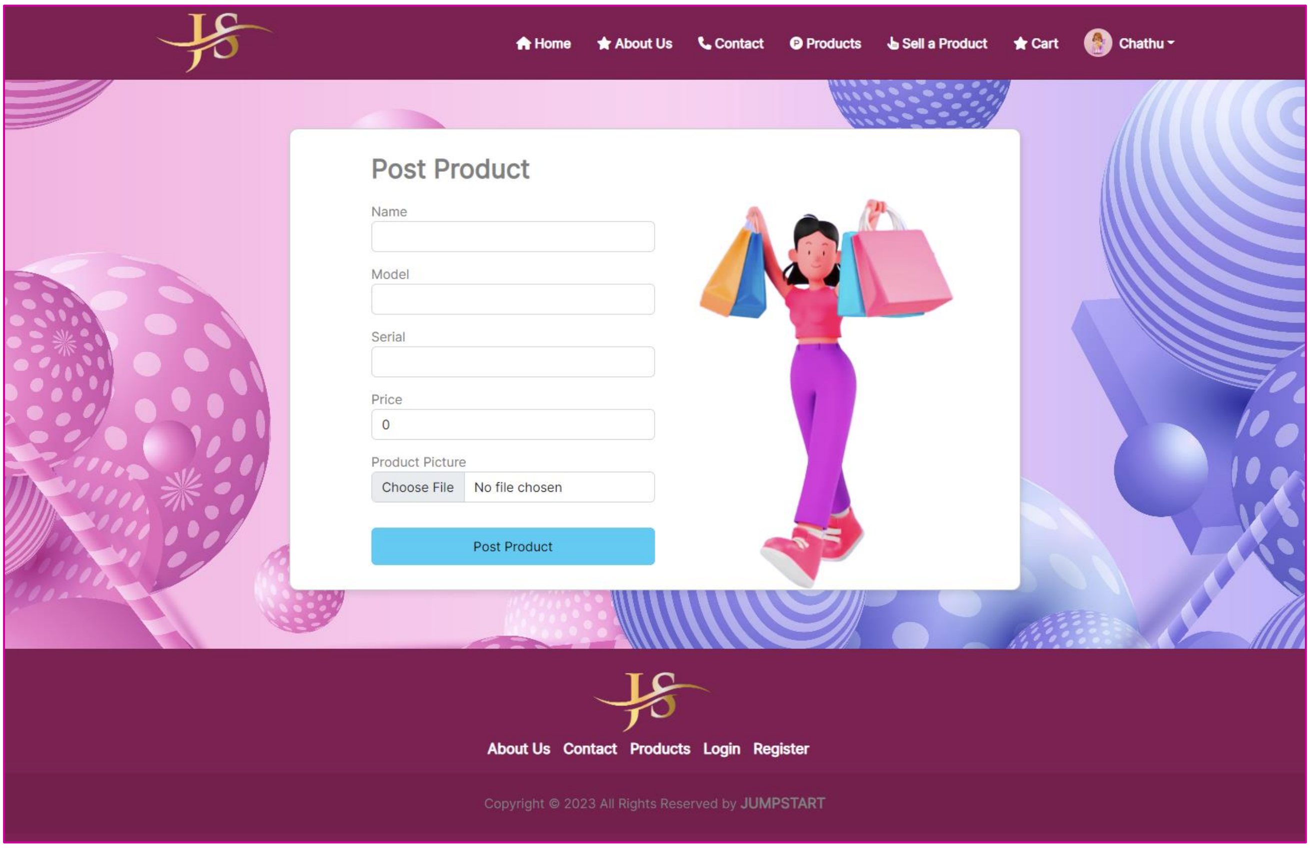
Password

••••••

Login
Not a Member? Register Now
About Us Contact Products Login Register

Copyright © 2023 All Rights Reserved by JUMPSTART

Post Product



Post Product

Name

Model

Serial

Price

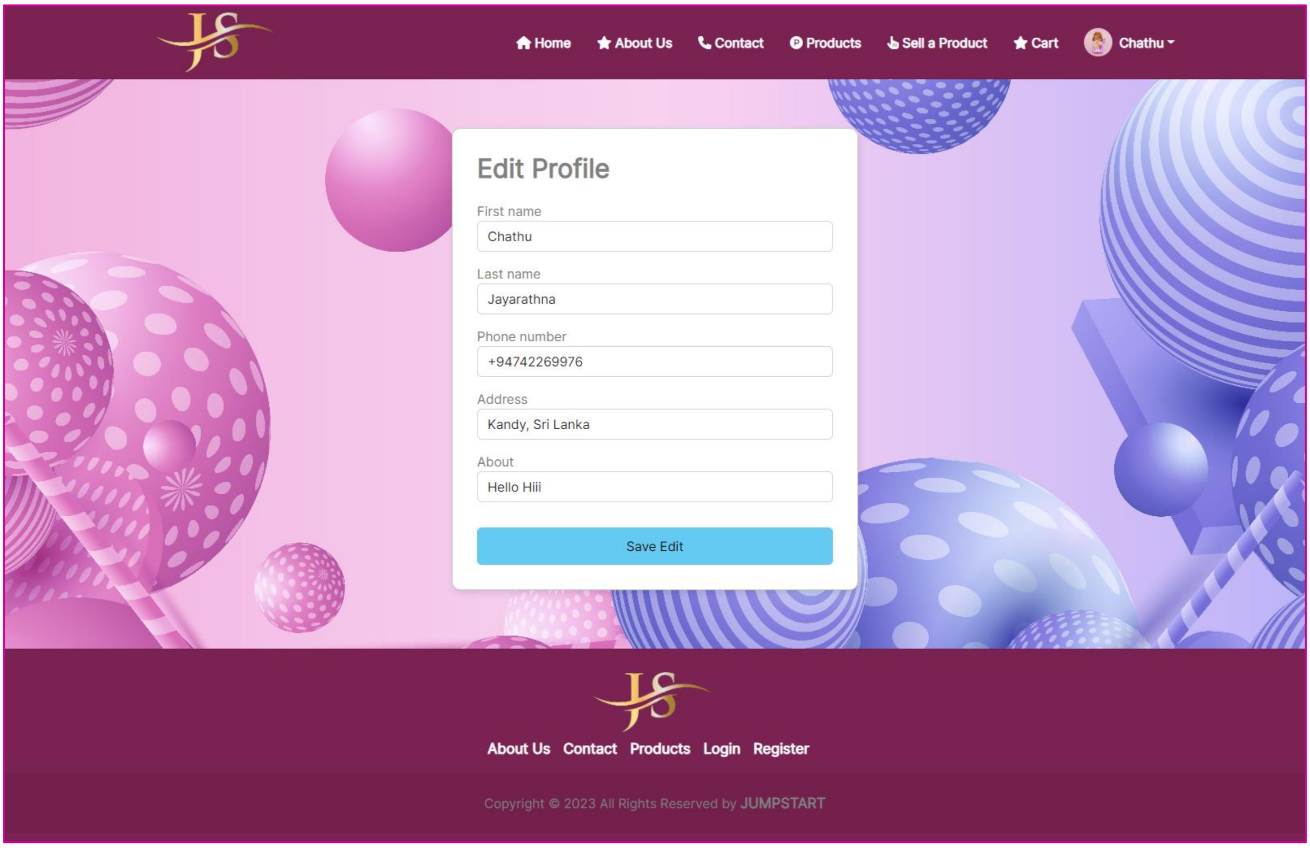
0

Product Picture

Choose File No file chosen

Copyright © 2023 All Rights Reserved by JUMPSTART

Edit Profile



Edit Profile

First name

Chathu

Last name

Jayarathna

Phone number

+94742269976

Address

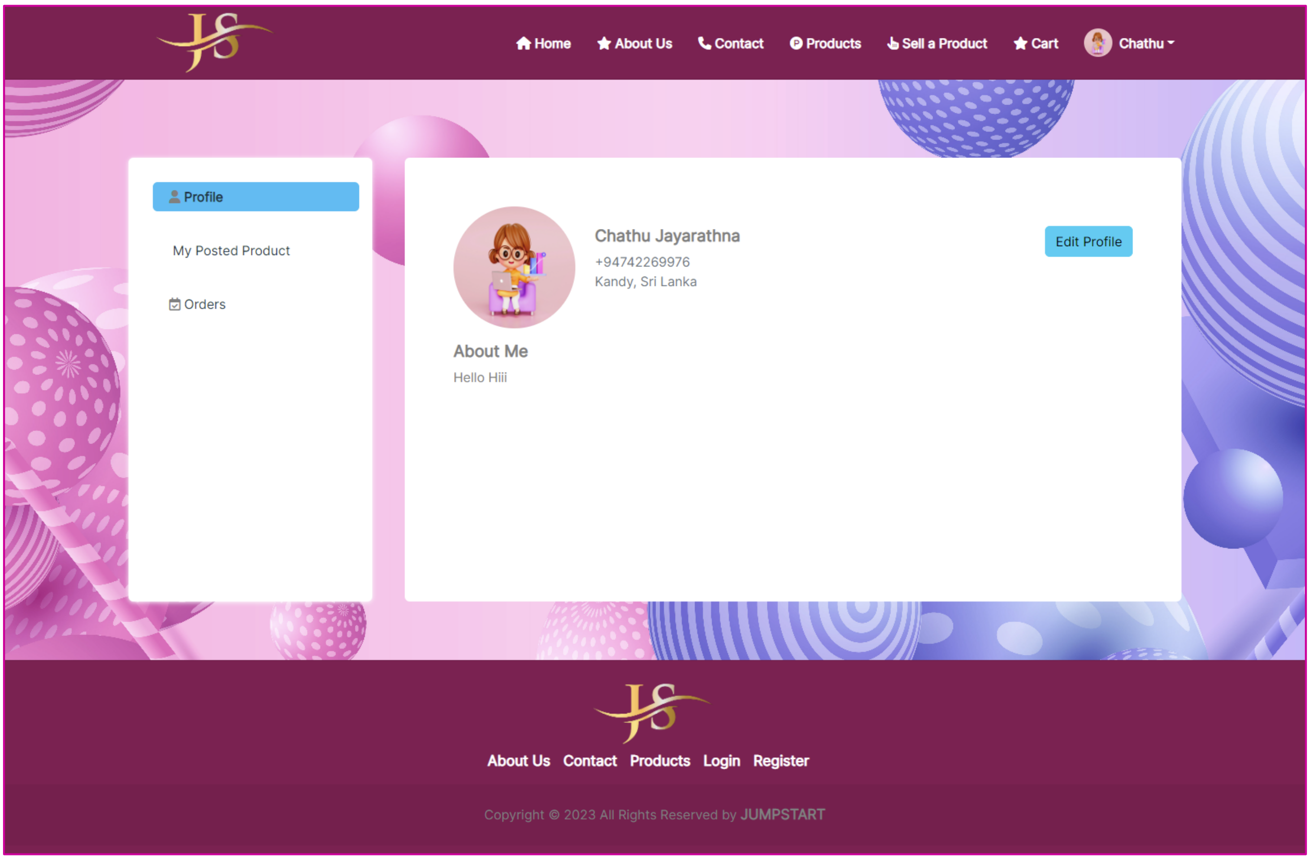
Kandy, Sri Lanka

About

Hello Hiii

Copyright © 2023 All Rights Reserved by JUMPSTART

Profile



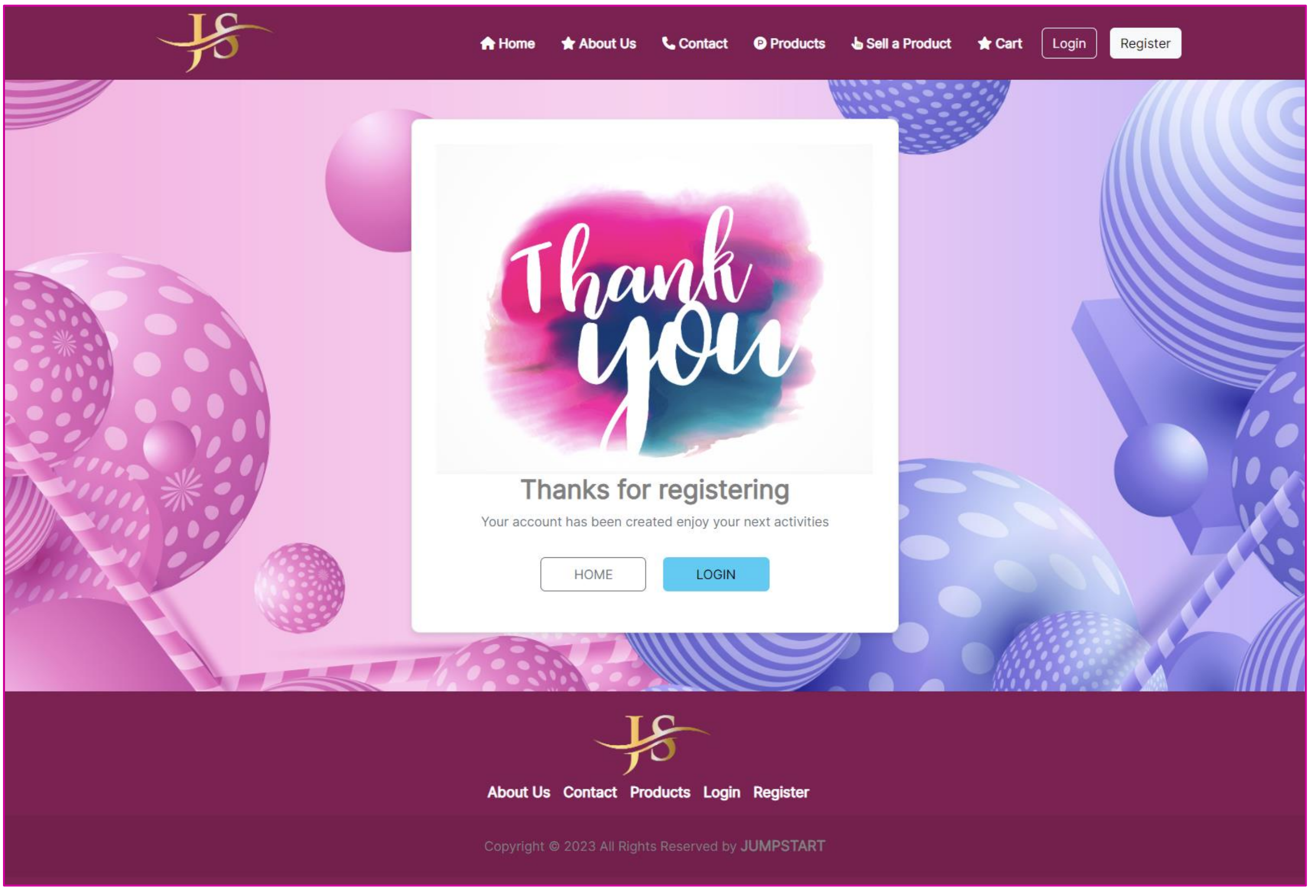
Chathu Jayarathna
+94742269976
Kandy, Sri Lanka

About Me

Hello Hiii

Copyright © 2023 All Rights Reserved by JUMPSTART

Registration Thank You

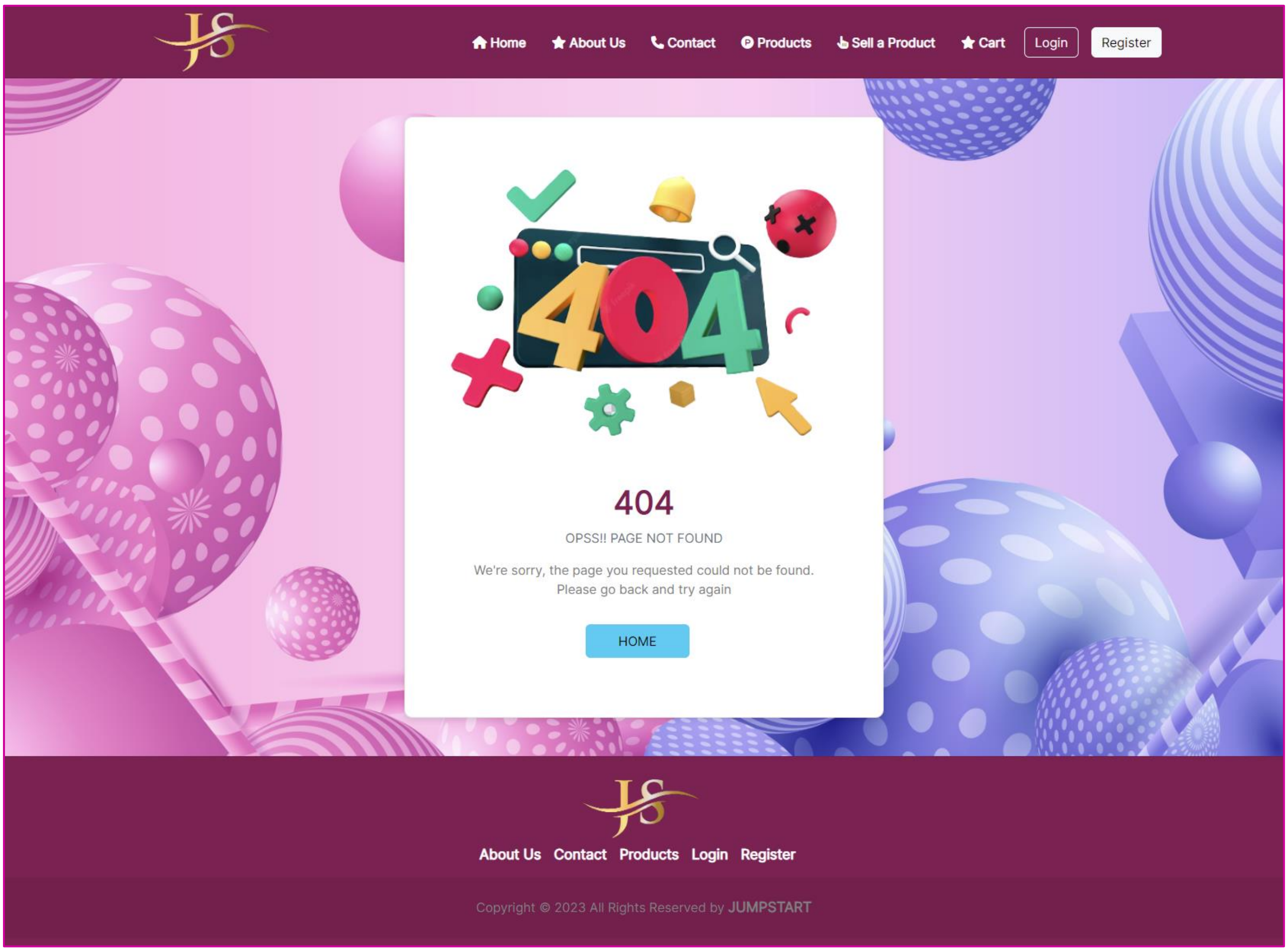


Thanks for registering

Your account has been created enjoy your next activities

Copyright © 2023 All Rights Reserved by JUMPSTART

Error Page



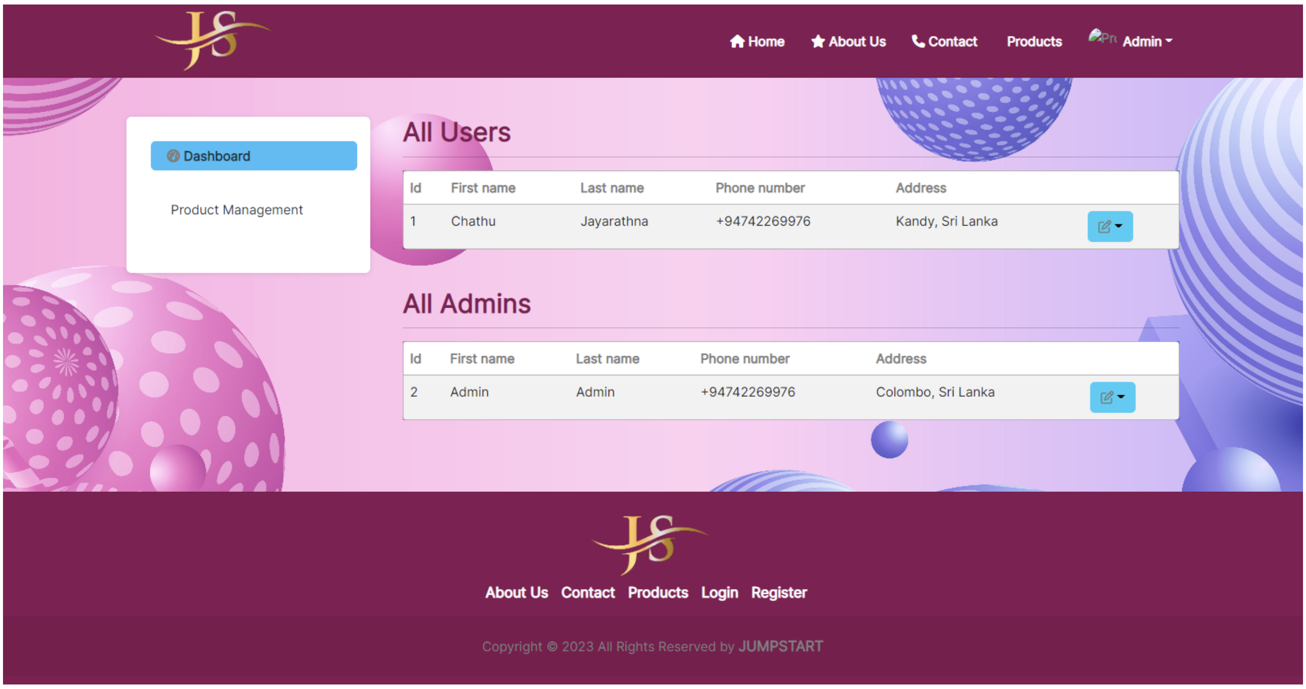
404

OPSS!! PAGE NOT FOUND

We're sorry, the page you requested could not be found.
Please go back and try again

Copyright © 2023 All Rights Reserved by JUMPSTART

Admin Dashboard



All Users

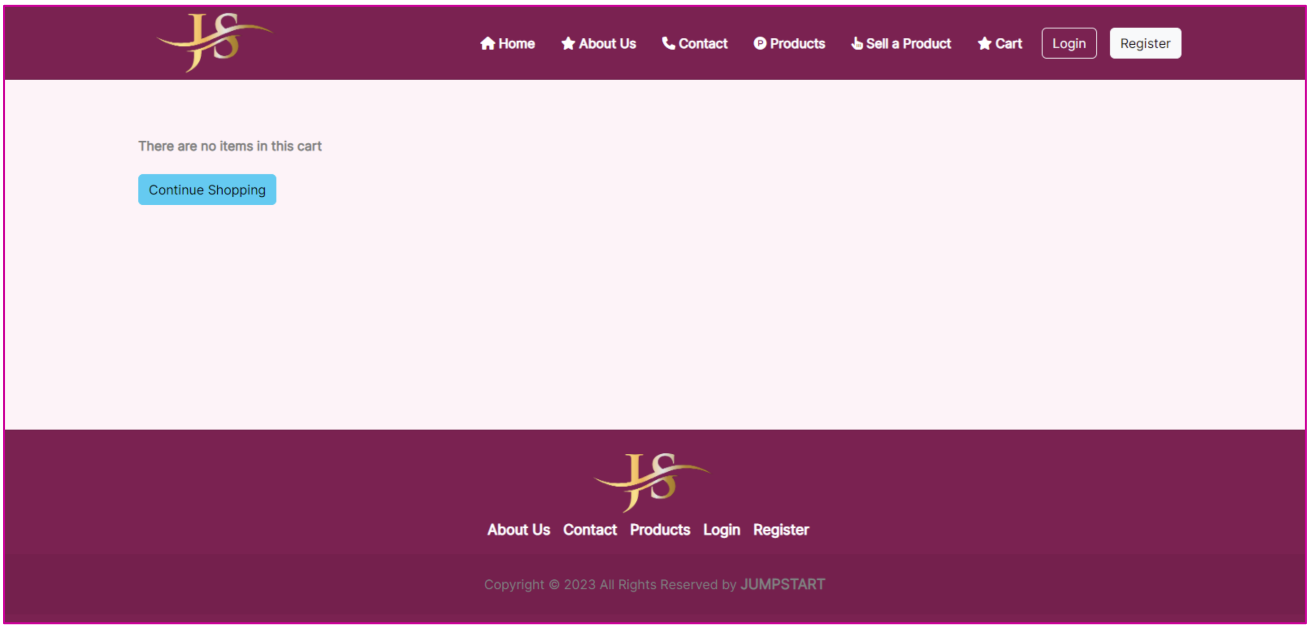
Id	First name	Last name	Phone number	Address
1	Chathu	Jayarathna	+94742269976	Kandy, Sri Lanka

All Admins

Id	First name	Last name	Phone number	Address
2	Admin	Admin	+94742269976	Colombo, Sri Lanka

Copyright © 2023 All Rights Reserved by JUMPSTART

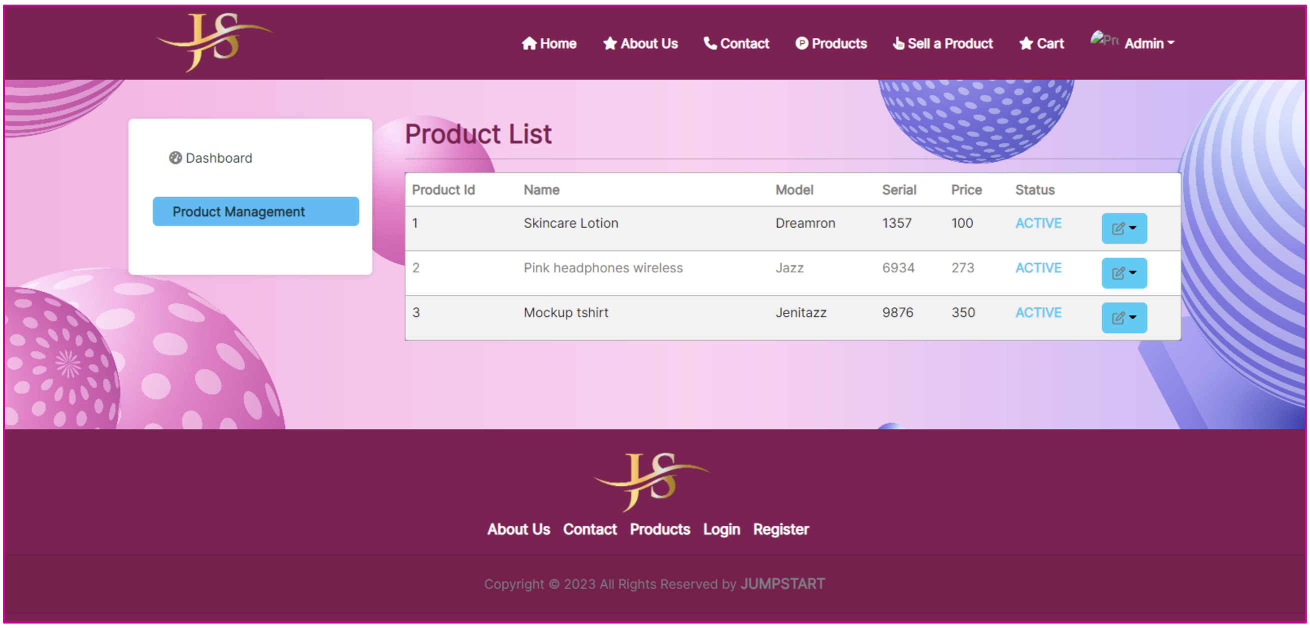
Cart Before Add Items



There are no items in this cart

Copyright © 2023 All Rights Reserved by JUMPSTART

Product Management



Product List

Product Id	Name	Model	Serial	Price	Status
1	Skincare Lotion	Dreamron	1357	100	ACTIVE
2	Pink headphones wireless	Jazz	6934	273	ACTIVE
3	Mockup tshirt	Jenitazz	9876	350	ACTIVE

Copyright © 2023 All Rights Reserved by JUMPSTART

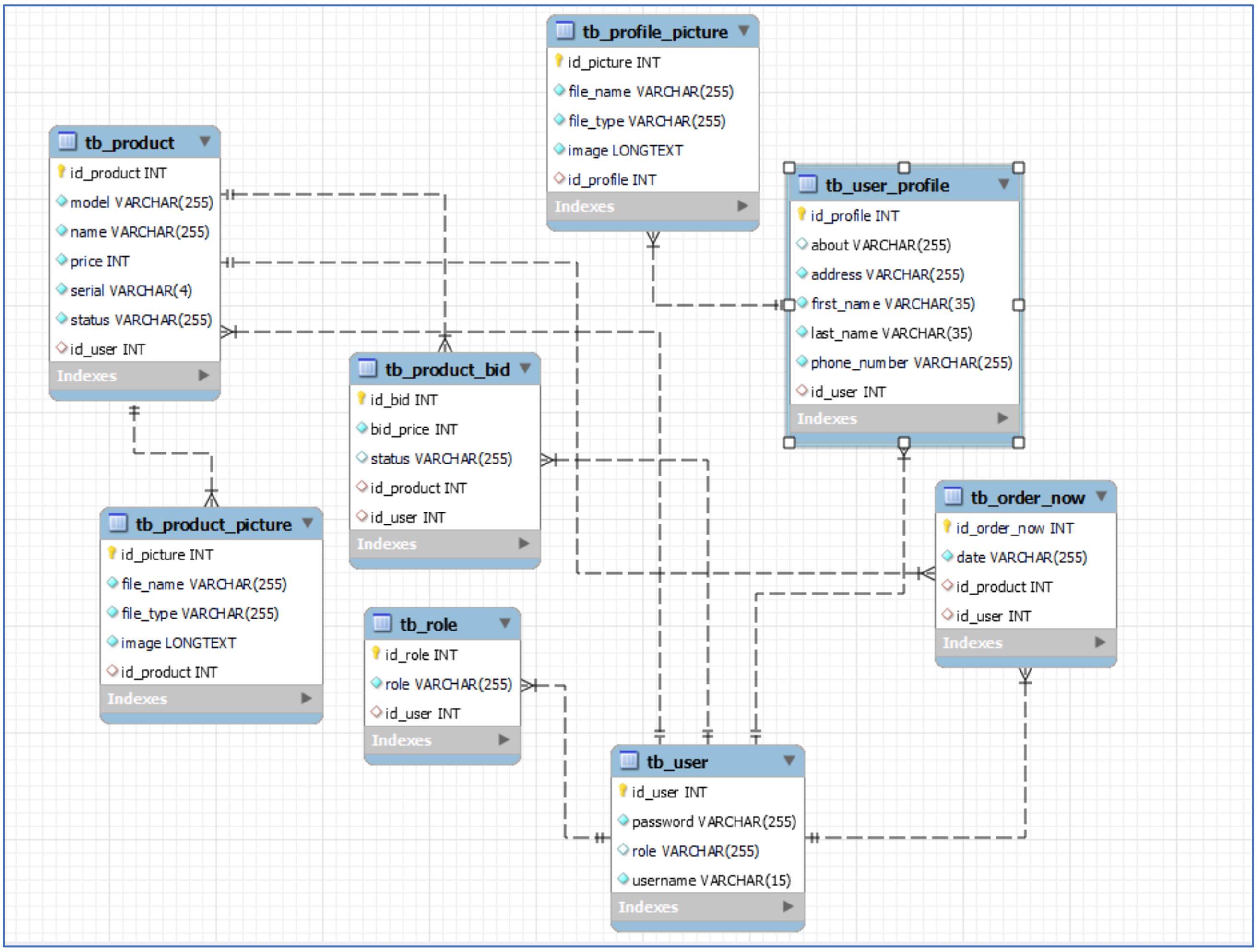
04) A description of the website's core functionality and features that have been implemented so far, along with any that are planned for the remaining development period.

- Landing Page – Anyone can search product by using the product name, serial number, or band and can see suggested products
- Header – Navigation for home, about us, contact us, all products, sell products, cart, logo
- Footer – Navigation for important links, Copyright, Logo
- About Us Section – About jumpstart like history mission vision, about team members
- Contact Us Section – Contact details of jumpstart like Address, email address, phone number send messages location map
- Login Page – Registered users, admin and partners can log in to their profile/ Dashboard by inputting their email address and password
- Registration Page – Anyone can register by inputting, Email address, First Name, Last Name, Phone number, About, and uploading an image
- Thank You Page – Thank you page will appear after successfully registering to jumpstart. This page includes thank you message and buttons to log in and go back to the home page
- Post Product Page – Partners can post products using this post product page. A form containing details such as products name, brand serial number, price, and image upload is included here
- Product Detail Page – Using this page anyone can see product details such as product name, Brand, Serial Number, and Price and can add the product to the cart.
- Update Profile Page – Any registered person can update their profile. A form containing details such as First Name, Last Name, Phone Number, Address, and about me is included here
- List Users Page – This page is for admin. Admin can see all registered people using this page. This page contains the registered people’s

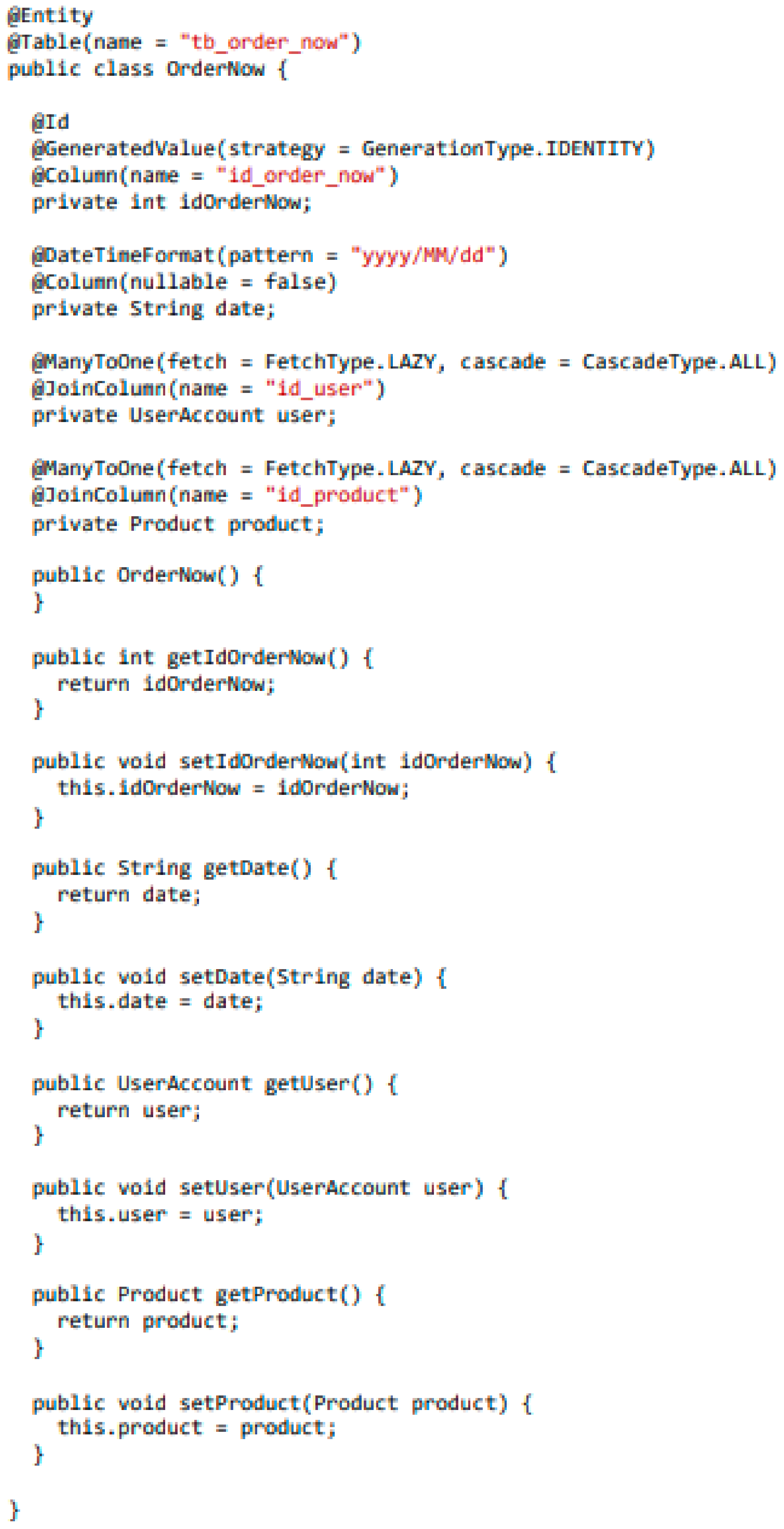
details table. Admin can view, Edit Delete people and admin can assign registered people as admins

- List Product Page – A table including details of the product such as product ID, Product Name, Brand, Serial Number, Price, Product Status (Available or not or Sold), Action (Edit, Delete)
- Cart – Here user can see added product list, and the total price and proceed to the pay button
- Terms & Conditions Page – All terms & Conditions of jumpstart

05) **Sample code snippets and database schemas that showcase the website's back-end functionality**.

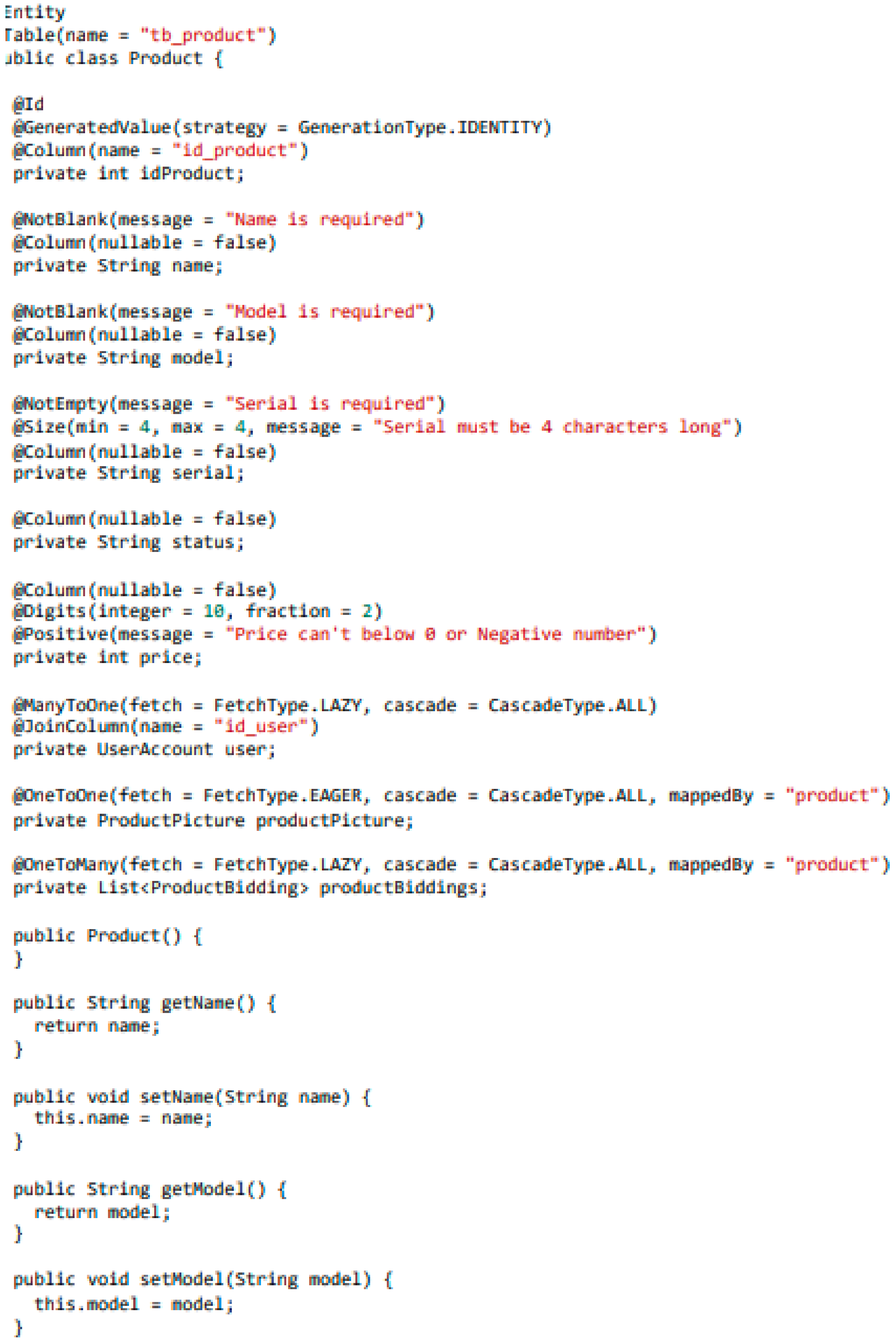


Database for Orders



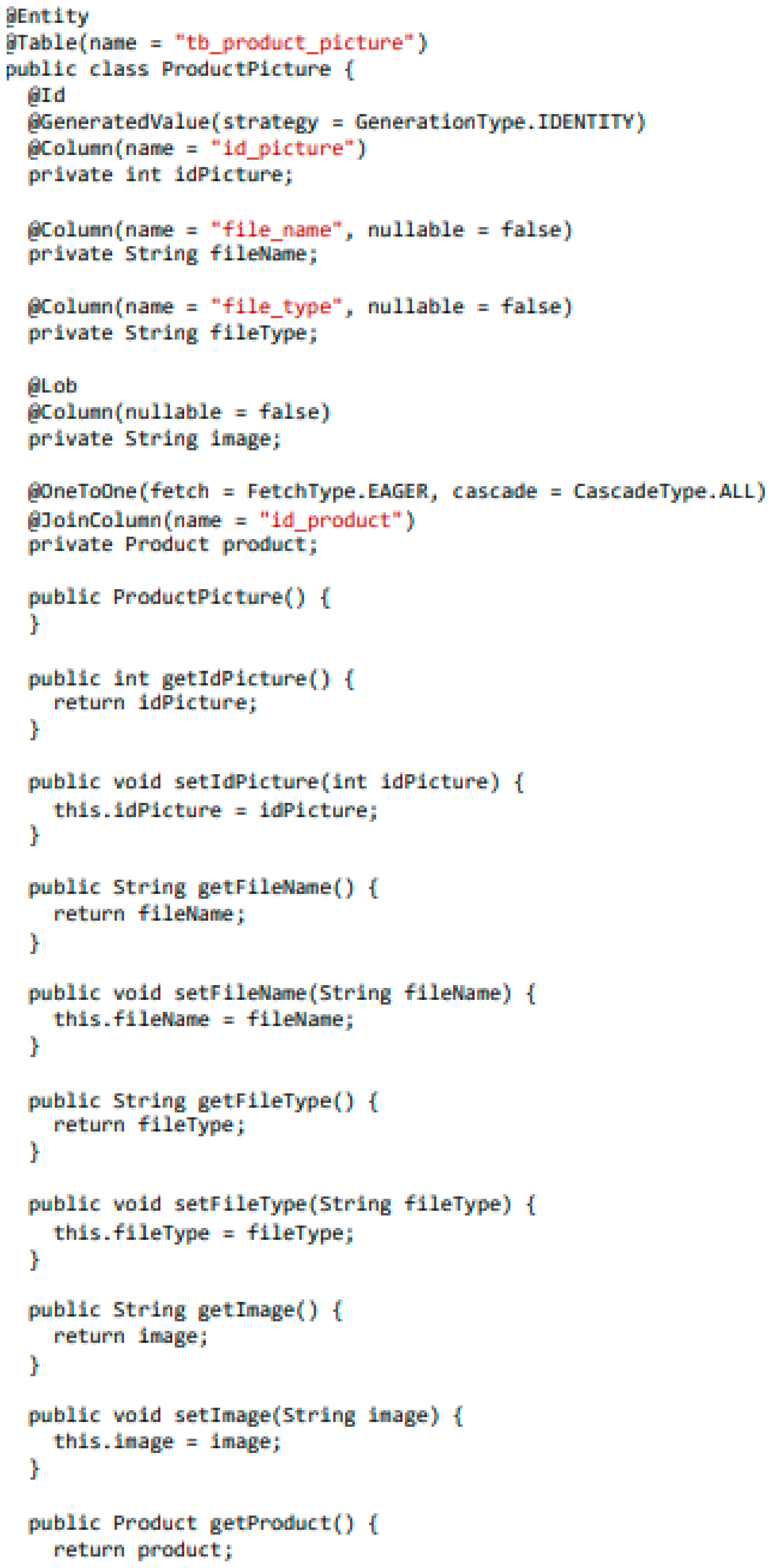
```
@Entity
@Table(name = "tb_order_now")
public class OrderNow {

  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_order_now")
  private int idOrderNow;

  @DateTimeFormat(pattern = "yyyy/MM/dd")
  @Column(nullable = false)
  private String date;

  @ManyToOne(fetch = FetchType.LAZY, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_user")
  private UserAccount user;

  @ManyToOne(fetch = FetchType.LAZY, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_product")
  private Product product;

  public OrderNow() {
  }

  public int getIdOrderNow() {
    return idOrderNow;
  }

  public void setIdOrderNow(int idOrderNow) {
    this.idOrderNow = idOrderNow;
  }

  public String getDate() {
    return date;
  }

  public void setDate(String date) {
    this.date = date;
  }

  public UserAccount getUser() {
    return user;
  }

  public void setUser(UserAccount user) {
    this.user = user;
  }

  public Product getProduct() {
    return product;
  }

  public void setProduct(Product product) {
    this.product = product;
  }

}
```

Database for Products



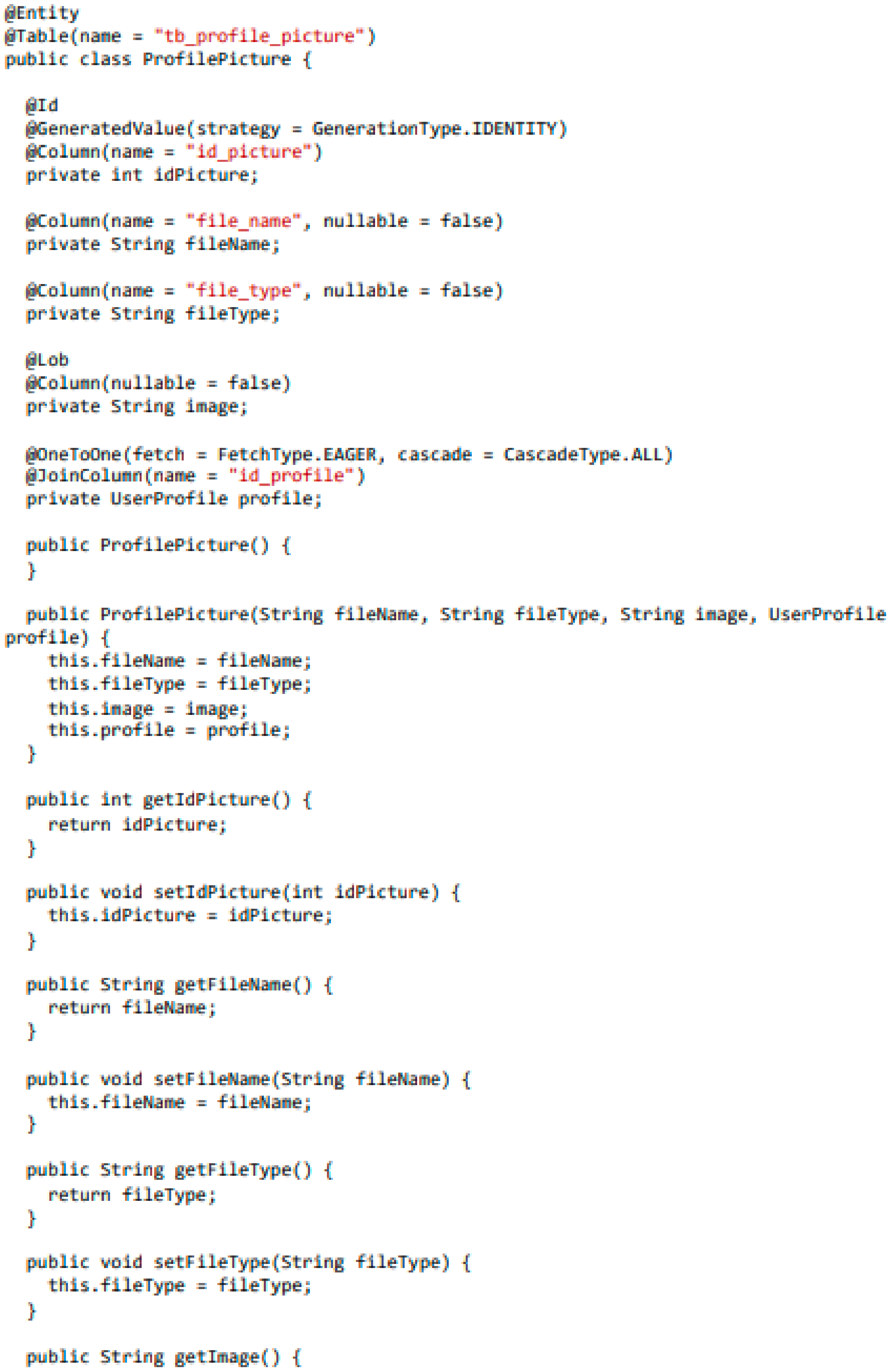
```
Entity
Table(name = "tb_product")
ublic class Product {

  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_product")
  private int idProduct;

  @NotBlank(message = "Name is required")
  @Column(nullable = false)
  private String name;

  @NotBlank(message = "Model is required")
  @Column(nullable = false)
  private String model;

  @NotEmpty(message = "Serial is required")
  @Size(min = 4, max = 4, message = "Serial must be 4 characters long")
  @Column(nullable = false)
  private String serial;

  @Column(nullable = false)
  private String status;

  @Column(nullable = false)
  @Digits(integer = 10, fraction = 2)
  @Positive(message = "Price can't below 0 or Negative number")
  private int price;

  @ManyToOne(fetch = FetchType.LAZY, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_user")
  private UserAccount user;

  @OneToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL, mappedBy = "product")
  private ProductPicture productPicture;

  @OneToMany(fetch = FetchType.LAZY, cascade = CascadeType.ALL, mappedBy = "product")
  private List<ProductBidding> productBiddings;

  public Product() {
  }

  public String getName() {
    return name;
  }

  public void setName(String name) {
    this.name = name;
  }

  public String getModel() {
    return model;
  }

  public void setModel(String model) {
    this.model = model;
  }
  public String getSerial() {
    return serial;
  }

  public void setSerial(String serial) {
    this.serial = serial;
  }

  public int getPrice() {
    return price;
  }

  public void setPrice(int price) {
    this.price = price;
  }

  public String getStatus() {
    return status;
  }

  public void setStatus(String status) {
    this.status = status;
  }

  public UserAccount getUser() {
    return user;
  }

  public void setUser(UserAccount user) {
    this.user = user;
  }

  public ProductPicture getProductPicture() {
    return productPicture;
  }

  public void setProductPicture(ProductPicture productPicture) {
    this.productPicture = productPicture;
  }

  public int getIdProduct() {
    return idProduct;
  }

  public void setIdProduct(int idProduct) {
    this.idProduct = idProduct;
  }

  public List<ProductBidding> getProductBiddings() {
    return productBiddings;
  }

  public void setProductBiddings(List<ProductBidding> productBiddings) {
    this.productBiddings = productBiddings;
  }

}
```

Database for Product Pictures



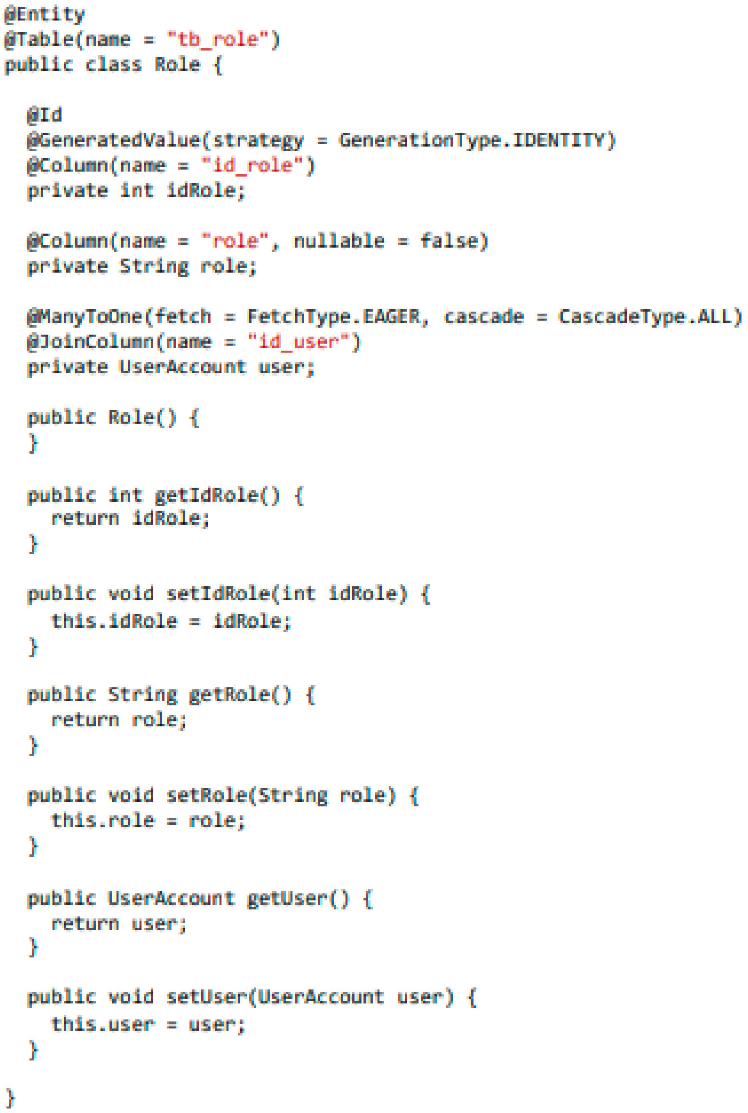
```
@Entity
@Table(name = "tb_product_picture")
public class ProductPicture {
  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_picture")
  private int idPicture;

  @Column(name = "file_name", nullable = false)
  private String fileName;

  @Column(name = "file_type", nullable = false)
  private String fileType;

  @Lob
  @Column(nullable = false)
  private String image;

  @OneToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_product")
  private Product product;

  public ProductPicture() {
  }

  public int getIdPicture() {
    return idPicture;
  }

  public void setIdPicture(int idPicture) {
    this.idPicture = idPicture;
  }

  public String getFileName() {
    return fileName;
  }

  public void setFileName(String fileName) {
    this.fileName = fileName;
  }

  public String getFileType() {
    return fileType;
  }

  public void setFileType(String fileType) {
    this.fileType = fileType;
  }

  public String getImage() {
    return image;
  }

  public void setImage(String image) {
    this.image = image;
  }

  public Product getProduct() {
    return product;
```

Database for profile pictures



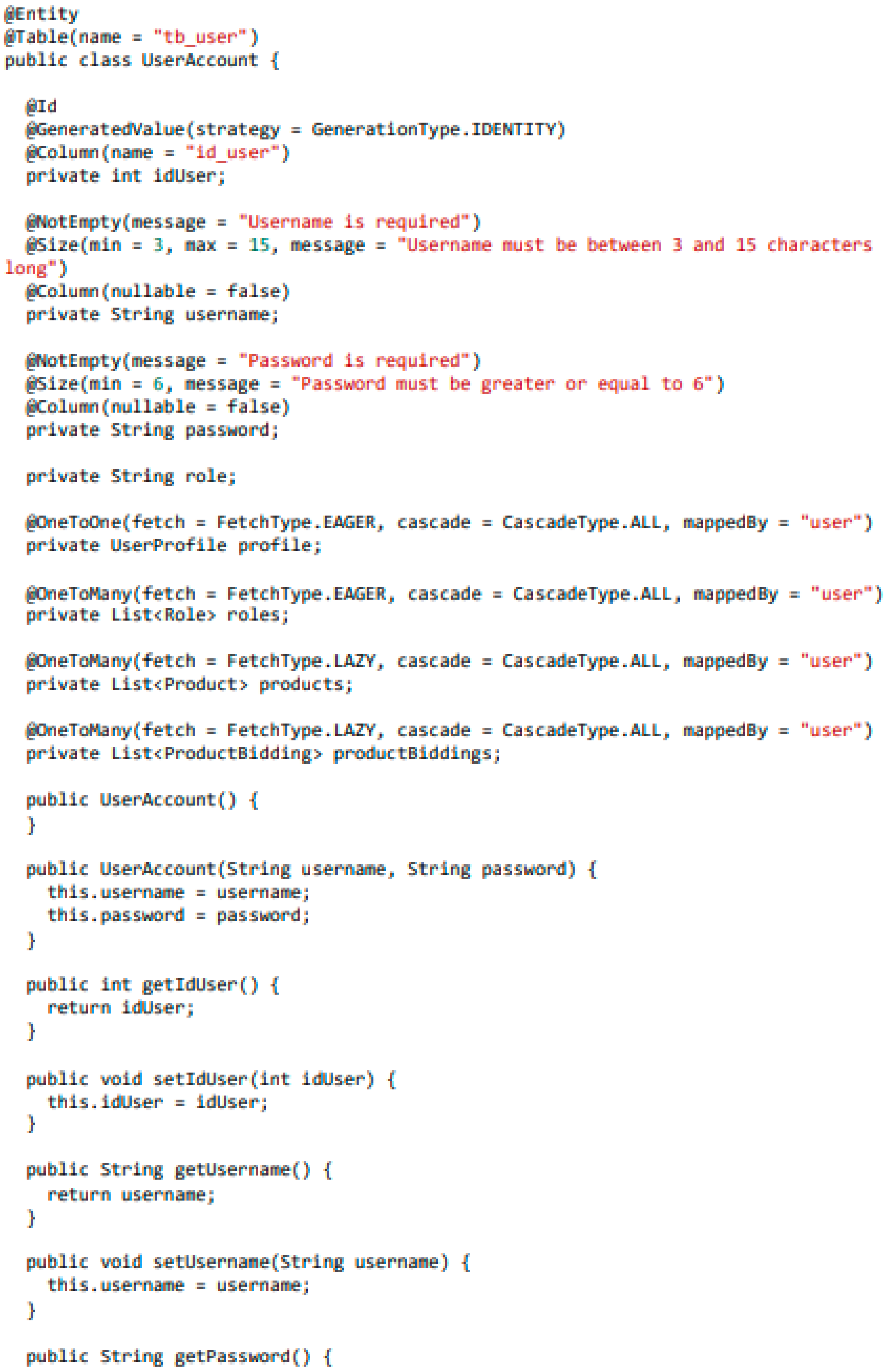
```
@Entity
@Table(name = "tb_profile_picture")
public class ProfilePicture {

  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_picture")
  private int idPicture;

  @Column(name = "file_name", nullable = false)
  private String fileName;

  @Column(name = "file_type", nullable = false)
  private String fileType;

  @Lob
  @Column(nullable = false)
  private String image;

  @OneToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_profile")
  private UserProfile profile;

  public ProfilePicture() {
  }

  public ProfilePicture(String fileName, String fileType, String image, UserProfile
profile) {
    this.fileName = fileName;
    this.fileType = fileType;
    this.image = image;
    this.profile = profile;
  }

  public int getIdPicture() {
    return idPicture;
  }

  public void setIdPicture(int idPicture) {
    this.idPicture = idPicture;
  }

  public String getFileName() {
    return fileName;
  }

  public void setFileName(String fileName) {
    this.fileName = fileName;
  }

  public String getFileType() {
    return fileType;
  }

  public void setFileType(String fileType) {
    this.fileType = fileType;
  }

  public String getImage() {
```

Database for Role



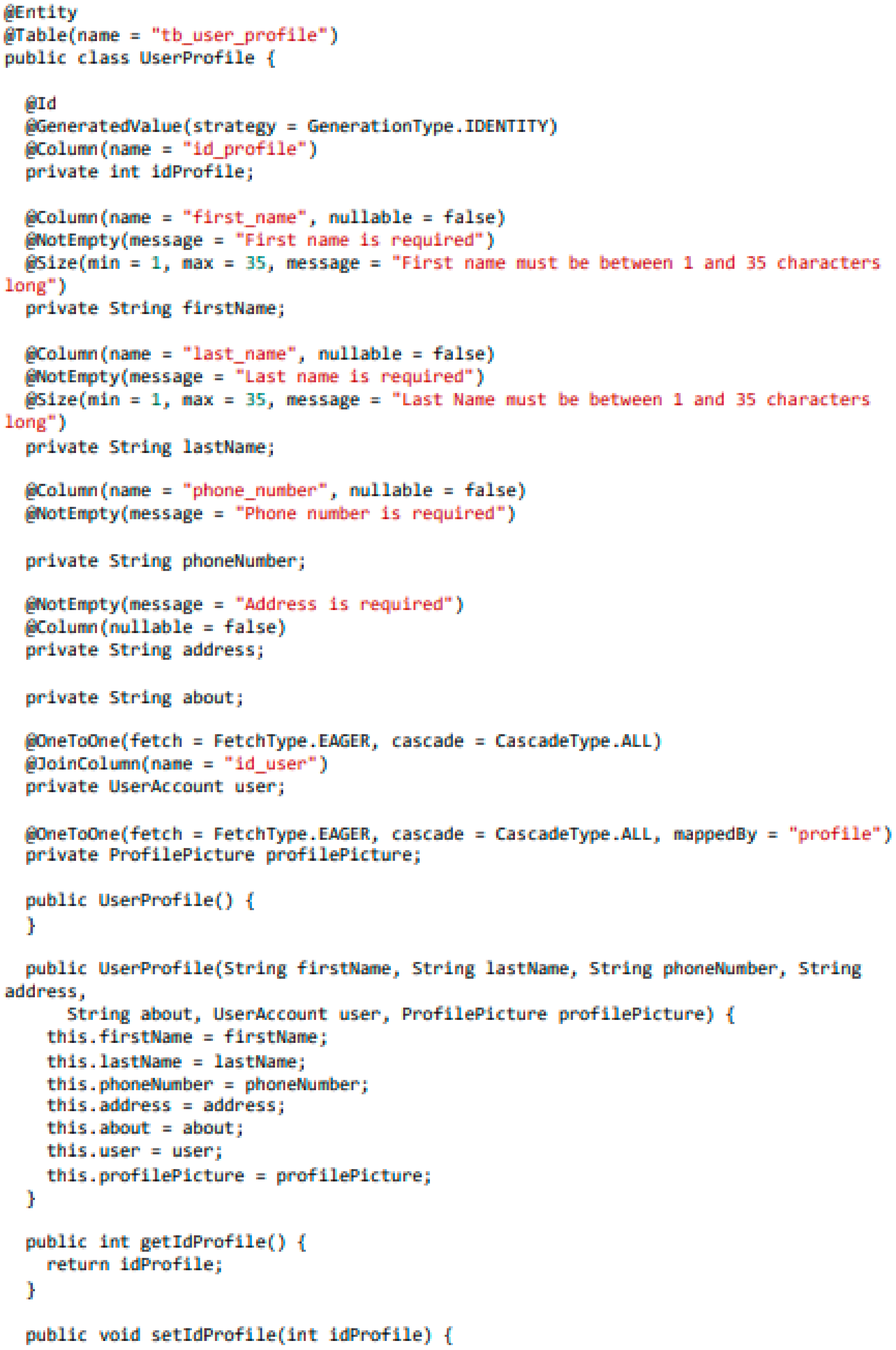
```
@Entity
@Table(name = "tb_role")
public class Role {

  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_role")
  private int idRole;

  @Column(name = "role", nullable = false)
  private String role;

  @ManyToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_user")
  private UserAccount user;

  public Role() {
  }

  public int getIdRole() {
    return idRole;
  }

  public void setIdRole(int idRole) {
    this.idRole = idRole;
  }

  public String getRole() {
    return role;
  }

  public void setRole(String role) {
    this.role = role;
  }

  public UserAccount getUser() {
    return user;
  }

  public void setUser(UserAccount user) {
    this.user = user;
  }

}
```

Database for accounts



```
@Entity
@Table(name = "tb_user")
public class UserAccount {

  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_user")
  private int idUser;

  @NotEmpty(message = "Username is required")
  @Size(min = 3, max = 15, message = "Username must be between 3 and 15 characters
long")
  @Column(nullable = false)
  private String username;

  @NotEmpty(message = "Password is required")
  @Size(min = 6, message = "Password must be greater or equal to 6")
  @Column(nullable = false)
  private String password;

  private String role;

  @OneToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL, mappedBy = "user")
  private UserProfile profile;

  @OneToMany(fetch = FetchType.EAGER, cascade = CascadeType.ALL, mappedBy = "user")
  private List<Role> roles;

  @OneToMany(fetch = FetchType.LAZY, cascade = CascadeType.ALL, mappedBy = "user")
  private List<Product> products;

  @OneToMany(fetch = FetchType.LAZY, cascade = CascadeType.ALL, mappedBy = "user")
  private List<ProductBidding> productBiddings;

  public UserAccount() {
  }

  public UserAccount(String username, String password) {
    this.username = username;
    this.password = password;
  }

  public int getIdUser() {
    return idUser;
  }

  public void setIdUser(int idUser) {
    this.idUser = idUser;
  }

  public String getUsername() {
    return username;
  }

  public void setUsername(String username) {
    this.username = username;
  }

  public String getPassword() {
    return password;
  }

  public void setPassword(String password) {
    this.password = password;
  }

  public List<Role> getRoles() {
    return roles;
  }

  public void setRoles(List<Role> roles) {
    this.roles = roles;
  }

  public UserProfile getProfile() {
    return profile;
  }

  public void setProfile(UserProfile profile) {
    this.profile = profile;
  }

  public List<Product> getProducts() {
    return products;
  }

  public void setProducts(List<Product> products) {
    this.products = products;
  }

  public List<ProductBidding> getProductBiddings() {
    return productBiddings;
  }

  public void setProductBiddings(List<ProductBidding> productBiddings) {
    this.productBiddings = productBiddings;
  }

  public String getRole() {
    return role;
  }

  public void setRole(String role) {
    this.role = role;
  }

}
```

Database for user accounts/ registration



```
@Entity
@Table(name = "tb_user_profile")
public class UserProfile {

  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  @Column(name = "id_profile")
  private int idProfile;

  @Column(name = "first_name", nullable = false)
  @NotEmpty(message = "First name is required")
  @Size(min = 1, max = 35, message = "First name must be between 1 and 35 characters
long")
  private String firstName;

  @Column(name = "last_name", nullable = false)
  @NotEmpty(message = "Last name is required")
  @Size(min = 1, max = 35, message = "Last Name must be between 1 and 35 characters
long")
  private String lastName;

  @Column(name = "phone_number", nullable = false)
  @NotEmpty(message = "Phone number is required")

  private String phoneNumber;

  @NotEmpty(message = "Address is required")
  @Column(nullable = false)
  private String address;

  private String about;

  @OneToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL)
  @JoinColumn(name = "id_user")
  private UserAccount user;

  @OneToOne(fetch = FetchType.EAGER, cascade = CascadeType.ALL, mappedBy = "profile")
  private ProfilePicture profilePicture;

  public UserProfile() {
  }

  public UserProfile(String firstName, String lastName, String phoneNumber, String
address,
      String about, UserAccount user, ProfilePicture profilePicture) {
    this.firstName = firstName;
    this.lastName = lastName;
    this.phoneNumber = phoneNumber;
    this.address = address;
    this.about = about;
    this.user = user;
    this.profilePicture = profilePicture;
  }

  public int getIdProfile() {
    return idProfile;
  }

  public void setIdProfile(int idProfile) {
    this.idProfile = idProfile;
  }

  public String getFirstName() {
    return firstName;
  }

  public void setFirstName(String firstName) {
    this.firstName = firstName;
  }

  public String getLastName() {
    return lastName;
  }

  public void setLastName(String lastName) {
    this.lastName = lastName;
  }

  public String getPhoneNumber() {
    return phoneNumber;
  }

  public void setPhoneNumber(String phoneNumber) {
    this.phoneNumber = phoneNumber;
  }

  public String getAddress() {
    return address;
  }

  public void setAddress(String address) {
    this.address = address;
  }

  public String getAbout() {
    return about;
  }

  public void setAbout(String about) {
    this.about = about;
  }

  public UserAccount getUser() {
    return user;
  }

  public void setUser(UserAccount user) {
    this.user = user;
  }

  public ProfilePicture getProfilePicture() {
    return profilePicture;
  }

  public void setProfilePicture(ProfilePicture profilePicture) {
    this.profilePicture = profilePicture;
  }

}
```

06) A detailed roadmap or timeline for the remaining tasks and deliverables, along with any changes or updates to the original project plan.

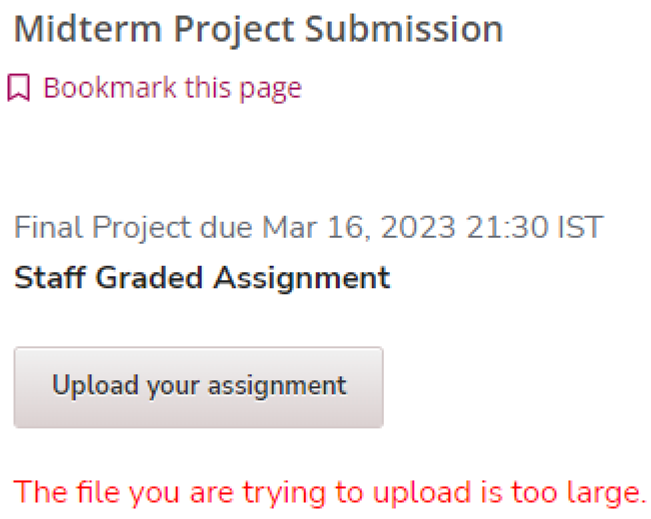
Task	Start Date	End Date
Add payment system	15/03/2023	16/03/2023
Change some colors of some parts of the application	16/03/2023	16/03/2023
Implement email system	16/03/2023	17/03/2023
Sold items for partners	17/03/2023	18/03/2023
Bulk email	18/03/2023	19/03/2023
Live Chat	19/03/2023	20/03/2023
Some part of cart system	20/03/2023	21/03/2023

Source Code

https://drive.google.com/drive/folders/1Az5gu4ldK7XvdUEYDQ3xKT4CwiaCrkFa?usp=sharing

Note:

The system does not allow to submit the word document with source code attached.



Midterm Project Submission

Final Project due Mar 16, 2023 21:30 IST

Staff Graded Assignment

The file you are trying to upload is too large.