

# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018



A Mini Project Report on

## *Online Shopping Management System*

*Submitted in partial fulfillment of the requirements as a part of the DBMS Lab for the award of degree of*

## **Bachelor of Engineering in Information Science and Engineering**

Submitted by

**Aditi Somashekar**  
**1RN16IS004**

**Mohana Subramanian**  
**1RN16IS054**

**Faculty Incharge**  
**Ms. Anusha U**  
Assistant Professor  
Dept. of ISE, RNSIT

**Lab Incharge**  
**Mr. R Rajkumar**  
Assistant Professor  
Dept. of ISE, RNSIT



**Department of Information Science and Engineering**  
**RNS Institute of Technology**

Channasandra, Dr. Vishnuvardhan Road, RR Nagar Post,  
Bengaluru – 560 098  
2018 – 2019

# **RNS Institute of Technology**

Channasandra, Dr.Vishnuvardhan Road, RR Nagar Post,  
Bengaluru – 560 098

## **DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING**



### **CERTIFICATE**

This is to certify that the internship report entitled *Online Shopping Management* has been successfully completed by **Aditi Somashekar and Mohana Subramanian** bearing USN **1RN16IS004 and 1RN16IS054**, presently V semester student of **RNS Institute of Technology** in partial fulfillment of the requirements as a part of the DBMS Laboratory for the award of the degree *Bachelor of Engineering in Information Science and Engineering* under **Visvesvaraya Technological University, Belagavi** during academic year 2018 – 2019. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements as a part of DBMS Laboratory for the said degree.

---

**Ms. Anusha U A**  
Faculty Incharge

---

**Mr. R Rajkumar**  
Lab Incharge

---

**Dr. M V Sudhamani**  
Professor and HOD

### **External Viva**

**Name of the Examiners**

1. \_\_\_\_\_

2. \_\_\_\_\_

**Signature with date**

\_\_\_\_\_

\_\_\_\_\_

# **ABSTRACT**

This project is a web-based online shopping application developed in JAVA LANGUAGE using JSP as front end. The main aim of “ONLINE SHOPPING” is to improve the services of Customers and vendors.

The primary features of the project entitled “ONLINE SHOPPING” are high accuracy, design flexibility and easy availability. And also it uses database tables Representing entities and relationships between entities.

The central concept of the application is to allow the customer to shop virtually using the Internet and allow customers to view the items and articles of their desire from the store. The information pertaining to the products are stored on RDBMS at the server side (store). The end user of this product is a store where the application is hosted on the web and the administrator maintains the database. The application which is deployed at the customer database, the details of the items are brought forward from the database for the customer view based on the selection through the menu.

# ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank **Management of RNS Institute of Technology** for providing such a healthy environment to carry out this Project work.

We would like to thank our beloved Director **Dr. H N Shivashankar** for his confidence feeling words and support for providing facilities throughout the course.

We would like to express our thanks to our Principal **Dr. M K Venkatesha** for his support and inspired us towards the attainment of knowledge.

We wish to place on record our words of gratitude to **Dr. M V Sudhamani**, Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind our Project work.

We would like to express our profound and cordial gratitude to my Faculty incharge **Mr. R Rajkumar**, Assistant Professor, Department of Information Science and Engineering for their valuable guidance, constructive comments and continuous encouragement throughout the Project work.

We would like to express our profound and cordial gratitude to my guide **Ms. Anusha U** , Assistant Professor, Department of Information Science and Engineering for their valuable guidance in preparing Project report.

We would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped us to carry out the project work.

And lastly, We would hereby acknowledge and thank our parents who have been a source of inspiration and also instrumental in carrying out this Project work.

**Aditi Somashekar**  
**1RN16IS004**

**Mohana Subramanian**  
**1RN16IS054**

# **TABLE OF CONTENTS**

## **CERTIFICATE**

## **ABSTRACT**

**i**

## **ACKNOWLEDGMENT**

**ii**

## **TABLE OF CONTENTS**

**iii**

## **LIST OF FIGURES**

**iv**

## **ABBREVIATIONS**

**v**

## **1. INTRODUCTION**

**1**

### **1.1 Background**

**1**

### **1.2 Introduction to the title of the project**

**2**

## **2. E R DIAGRAM AND RELATIONAL SCHEMA DIAGRAM**

**3**

## **3. SYSTEM DESIGN**

**7**

### **3.1 Tables**

**7**

## **4. IMPLEMENTATION**

**10**

### **4.1 Front end and Back end used**

**8**

### **4.2 Discussion of code segments**

**12**

### **4.3 Applications of project Work**

**15**

### **4.4 Discussion of the Results**

**16**

## **5. CONCLUSION AND FUTURE ENCHANCEMENTS**

**22**

## **REFERENCES**

## LIST OF FIGURES

<b>Fig. No.</b>	<b>Descriptions</b>	<b>Page</b>
Fig 2.1	ER Diagram	3
Fig2.2	Schema diagram	5
Fig. 3.1.1	Brand table	11
Fig. 3.1.2	Collections table	11
Fig 3.1.3	Clothes table	12
Fig 3.1.4	Login table	12
Fig 3.1.5	Register table	13
Fig 4.4.1	Login page	14
Fig 4.4.2	Register page	16
Fig 4.4.3	home page	16
Fig 4.4.4	home page 2	17
Fig 4.4.5	home page 3	17
Fig 4.4.6	product display 1	18
Fig 4.4.7	summer product display 1	18
Fig 4.4.8	summer product display 2	19
Fig 4.4.9	winter product display 1	19
Fig 4.4.10	winter product display 2	20
Fig 4.4.11	spring product display 1	20
Fig 4.4.12	spring product display 2	21

# ABBREVIATIONS

DBMS:	Database Management Systems
GUI:	Graphical User Interface
IDE:	Integrated Development Environment
JSP:	Java Servlet Pages
HTML5 :	Hyper Text Markup Language Version 5
JDBC:	Java Database Connector
SQL:	Structured Query Language
ER:	Entity Relationship (diagram)
MPD:	Mobile Phone Database
CSS:	Cascading Style Sheets
ID:	Identifier
CE:	Classic Edition
API:	Application Programming Interface
RDBMS:	Relational Database management system

## Chapter 1

# INTRODUCTION

### 1.1 Background

A **database** is an organized collection of data, generally stored and accessed electronically from a computer system. Where databases are more complex they are often developed using formal design and modeling techniques.

The database management system (DBMS) is the software that interacts with end users, applications, the database itself to capture and analyze the data and provides facilities to administer the database. The sum total of the database, the DBMS and the associated applications can be referred to as a "database system". Often the term "database" is also used to loosely refer to any of the DBMS, the database system or an application associated with the database.

The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified and the database schema, which defines the database's logical structure. These three foundational elements help provide concurrency, security, data integrity and uniform administration procedures. Typical database administration tasks supported by the DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.



## 1.2 Introduction To Online Shopping

Online shopping is a form of electronic commerce which allows consumers to directly buy goods or services from a seller over the Internet using a web browser. Consumers find a product of interest by visiting the website of the retailer directly or by searching among alternative vendors using a shopping search engine, which displays the same product's availability and pricing at different e-retailers. As of 2016, customers can shop online using a range of different computers and devices, including desktop, computers, laptops, tablet computers and smartphones.

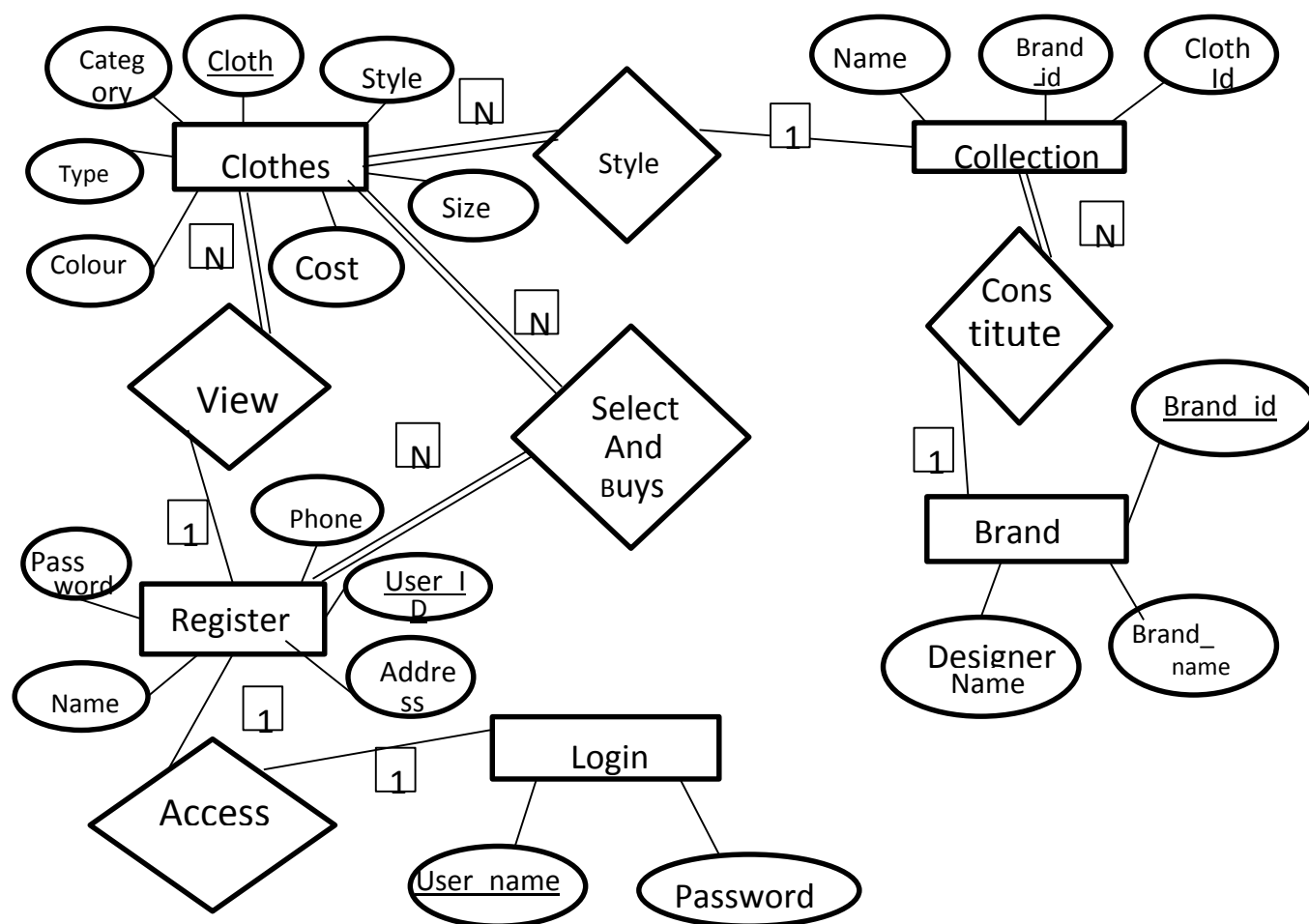
When an online store is set up to enable businesses to buy from another businesses, the process is called business-to-business online shopping. A typical online store enables the customer to browse the firm's range of products and services, view photos or images of the products, along with information about the product specifications, features and prices.

Online stores typically enable shoppers to use "search" features to find specific models, brands or items. Online customers must have access to the Internet and a valid method of payment in order to complete a transaction, such as a credit card, an Interac-enabled debit card, or a service

## CHAPTER-2

# ER DIAGRAM AND RELATIONAL SCHEMA DIAGRAM

### 2.1 Entity-Relationship Diagram



**Fig 2.1**

ER Diagram describes entities, attributes and relationships

Entity types like REGISTER and BRAND are in rectangular boxes.

- Relationships like ACCESS and VIEW are in diamond boxes, attached to entity types with straight lines.
- Attributes are shown in ovals, each attached by a straight line to entity or relationship type.

- Multivalued attributes are in double ovals.
- Key attributes (like CLOTH\_ID) are underlined. Component attributes of a composite attribute are attached to oval representing it.

### 2.1.1 ER DIAGRAM DESCRIPTION

1. COLLECTIONS:CLOTHES is of cardinality 1: N as one collection have have many styles of clothes. There is total participation of both entities as :
  - A cloth cannot exist without belonging to a collection.
  - A collection cannot exist without clothes that belong to it.
2. BRAND: COLLECTION is of cardinality 1:N as 1 brand can have many collections. Also each cloth belongs to a particular brand only. There is total participation of HARDWARE\_INFO and partial participation of PHONES as:
  - A collection cannot exist without belonging to a brand.
3. LOGIN: REGISTER is of cardinality 1:1 as 1 user can have only register for account. There is partial participation of login and register as:
  - one user can only register once.
  - Login cannot be done without registering.
3. CLOTHES:REGISTER is of cardinality N:N as many users can select and buy many clothes There is total participation of REGISTER and total participation of clothes

## 2.2 RELATIONAL SCHEMA DIAGRAM

### BRAND

Brand_Id	Brand_Name	Designer_Name
----------	------------	---------------

### REGISTER

<u>User_id</u>	Usr_Name	<u>Password</u>	Phone	Address
----------------	----------	-----------------	-------	---------

### LOGIN

User_Name	Password
-----------	----------

### CLOTHES

Catego ry	<u>Cloth_i d</u>	Type	Style	Colour	Size	Cost
--------------	----------------------	------	-------	--------	------	------

### COLLECTIONS

Name	Brand_id	Cloth -id
------	----------	-----------

Fig 2.2 Relational Schema Diagram

### General Constraints

- **NULL Constraint:** Attributes that are under NOT NULL constraints have to be filled compulsorily. Almost all the attributes in the project are under NOT NULL constraint.
- **Entity Integrity Constraint:** This constraint makes sure that no primary key can have a NULL value assigned to it. The primary keys involved in the project include:
  - CLOTH\_ID
  - USER\_NAME
  - USER\_ID

- **Referential Integrity Constraints:** A table in the back end of the project may have references pointing to an attribute in another table. For example: BRAND\_ID in the COLLECTIONS table refers to BRAND\_ID in BRAND table. The various tables are also linked with multiple foreign keys which are all set to cascade any update or delete operation on the attribute in the main table. The various Foreign Key attributes are:

- BRAND\_ID
- USER\_NAME
- PASSWORD

## Definition of Tables

- 1) **CLOTHES:** This is a master table that consists of details like cloth\_id, category, type, cost, size and style. Here only one gallery is taken into consideration with no branches.
- 2) **COLLECTION:** It is a table which is for type collections like summer, winter fall or spring. It consists of collection name, brand\_id, cloth\_id.
- 3) **REGISTER:** It is a table that consists of details of the user to register. It consists of details like phone, password, name, address etc.
- 4) **LOGIN:** It is a table that consists of details of the user to login after registering. It consists of details like user\_name and password.
- 5) **BRAND:** This table consists of details of the clothes. Includes brands like shein, lulu and sky etc. it consists of designer\_name, brand\_name and brand\_id.

## CHAPTER-3

### SYSTEM DESIGN

#### 3.1 TABLES

##### 3.1.1 Brand Table

```
create table brand(  
brand_id numeric(10),  
brand_name varchar(20),  
designer_name varchar(20),  
primary key (brand_id));
```

##### TABLE DESCRIPTION

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BRAND	BRAND_ID	Number	-	10	0	1	-	-	-
	BRAND_NAME	Varchar2	20	-	-	-	✓	-	-
	DESIGNER_NAME	Varchar2	20	-	-	-	✓	-	-
1 - 3									

Fig 3.1.1

##### 3.1.2 Collections Table

```
create table collections(  
name varchar(20),  
brand_id numeric(10),  
cloth_id numeric(10),  
foreign key (brand_id) references brand (brand_id) on delete cascade,  
foreign key (cloth_id) references clothes (cloth_id) on delete cascade);
```

##### TABLE DESCRIPTION

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
COLLECTIONS	NAME	Varchar2	20	-	-	-	✓	-	-
	BRAND_ID	Number	-	10	0	-	✓	-	-
	CLOTHES_ID	Number	-	10	0	-	✓	-	-
1 - 3									

Fig 3.1.2

### 3.1.3 Clothes Table

```
create table cloths(
category varchar(20),
cloth_id numeric(10),
type char(20),
style varchar(20),
colour varchar(20),
size numeric(10),
cost numeric(10),
collection_name varchar(15),
primary key(cloth_id));
```

#### TABLE DESCRIPTION

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CLOTHES	CLOTHES_ID	Number	-	10	0	1	-	-	-
	CATEGORY	Varchar2	10	-	-	-	✓	-	-
	STYLE	Varchar2	10	-	-	-	✓	-	-
	SIZES	Varchar2	10	-	-	-	✓	-	-
	COST	Varchar2	10	-	-	-	✓	-	-
	COLOUR	Varchar2	10	-	-	-	✓	-	-
	IMAGES	Varchar2	30	-	-	-	✓	-	-
1 - 7									

Fig 3.1.3

### 3.1.4 Login

```
create table login(
username varchar(20),
password varchar(20),
primary key(username));
```

#### TABLE DESCRIPTION

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
LOGIN	USERNAME	Varchar2	20	-	-	1	-	-	-
	PASSWORD	Varchar2	20	-	-	-	✓	-	-
1 - 2									

Fig 3.1.4

### 3.1.5 Register Table

```
create table register(
user_id numeric(10),
name char(25),
password varchar(20),
phone numeric(10),
address varchar(20),
primary key(user_id),
foreign key(name) references login(username) on delete cascade);
```

#### TABLE DESCRIPTION

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
REGISTER	USERNAME	Varchar2	10	-	-	1	-	-	-
	EMAIL	Varchar2	20	-	-	-	✓	-	-
	PASSWORD	Varchar2	10	-	-	-	✓	-	-
	ADDRESS	Varchar2	20	-	-	-	✓	-	-
	PHONE	Varchar2	10	-	-	-	✓	-	-
1 - 5									



# CHAPTER 4

## IMPLEMENTATION

### 4.1 FRONT END AND BACK END USED

#### 4.1.1 Front End Used

##### **JavaServer Pages (JSP)**

It is a technology that helps software developers create dynamically generated web pages based on HTML, XML, or other document types.

To deploy and run JavaServer Pages, a compatible web server with a servlet container, such as Apache Tomcat or Jetty, is required. Architecturally, JSP may be viewed as a highlevel abstraction of Java servlets. JSPs are translated into servlets at runtime, therefore JSP is a Servlet; each JSP servlet is cached and re-used until the original JSP is modified.

JSP allows Java code and certain pre-defined actions to be interleaved with static web markup content, such as HTML, with the resulting page being compiled and executed on the server to deliver a document. The compiled pages, as well as any dependent Java libraries, contain Java bytecode rather than machine code. Like any other Java program, they must be executed within a Java virtual machine (JVM) that interacts with the server's host operating system to provide an abstract, platform-neutral environment.

JSPs are usually used to deliver HTML and XML documents, but through the use of OutputStream, they can deliver other types of data as well.

## 4.1.2 BACK END USED

### SQL Structured Query Language

It is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data where there are relations between different entities/variables of the data. SQL offers two main advantages over older read/write APIs like ISAM or VSAM: first, it introduced the concept of accessing many records with one single command; and second, it eliminates the need to specify *how* to reach a record, e.g. with or without an index

Originally based upon relational algebra and tuple relational calculus, SQL consists of many types of statements,<sup>[9]</sup> which may be informally classed as sublanguages, commonly: a data query language (DQL),<sup>[a]</sup> a data definition language (DDL),<sup>[b]</sup> a data control language (DCL), and a data manipulation language (DML).<sup>[c][10]</sup> The scope of SQL includes data query, data manipulation (insert, update and delete), data definition (schema creation and modification), and data access control. Although SQL is often described as, and to a great extent is, a declarative language (4GL), it also includes procedural elements.

---

## 4.2 DISCUSSION OF CODE SEGMENTS

### 4.2.1 Query to insert into register table

```
public int addUser(RegisterModel rm) throws SQLException
{
    String sql="insert into register values(?,?,?)";
        PreparedStatement ps = con.prepareStatement(sql);
        ps.setString(1, rm.getUsername());
        ps.setString(2,rm.getEmail());
        ps.setString(3,rm.getPassword());
        return ps.executeUpdate();
}
```

### 4.2.2 Query to insert into login table

```
public int addUser1(LoginModel lm) throws SQLException
{
    String s="insert into login values(?,?)";
    PreparedStatement ps1 = con.prepareStatement(s);
    ps1.setString(1, lm.getUsername());
    ps1.setString(2,lm.getPassword());
    return ps1.executeUpdate();
}
```

### 4.2.3 Query to select username and password

```
public boolean login(LoginModel lm) throws SQLException
{
    String sql="select username,password from login where username=? and
password=?";
    PreparedStatement prepareStatement = con.prepareStatement(sql);
    prepareStatement.setString(1, lm.getUsername());
    prepareStatement.setString(2,lm.getPassword());
    ResultSet executeQuery = prepareStatement.executeQuery();
    return executeQuery.next();
}
```

### 4.2.4 Query to select from clothes

```
public List<Clothes> fetchClothes() throws SQLException
{
    String sql="select * from clothes";
    PreparedStatement stmt = con.prepareStatement(sql);
    ResultSet rs = stmt.executeQuery();
}
```

```
List<Clothes> data=new ArrayList<>();
while(rs.next()) {
    Clothes c=new Clothes();
    c.setClothes_id(rs.getInt("cloth_id"));
    c.setCategory(rs.getString("category"));
    c.setType(rs.getString("type"));
    c.setStyle(rs.getString("style"));
    c.setColour(rs.getString("colour"));
    c.setSize(rs.getInt("size"));
    c.setCost(rs.getInt("cost"));
    data.add(c);
}
```

#### 4.2.5 Query to select from cloth with cloth\_id

```
public Clothes fetchById(int id)throws SQLException
{
    String sql="select * from clothes where id=?";
    PreparedStatement stmt = con.prepareStatement(sql);
    stmt.setInt(1, id);
    ResultSet rs = stmt.executeQuery();
    Clothes c=new Clothes();
    while(rs.next()) {
        c.setClothes_id(rs.getInt("cloth_id"));
        c.setCategory(rs.getString("category"));
        c.setType(rs.getString("type"));
        c.setStyle(rs.getString("style"));
        c.setColour(rs.getString("colour"));
        c.setSize(rs.getInt("size"));
        c.setCost(rs.getInt("cost"));
    }
}
```

#### 4.2.6 Query to select clothes from collection

```
public List<Clothes> fetchCollection(String str) throws SQLException
{
    String sql="select * from clothes where collection_name=?";
    PreparedStatement stmt = con.prepareStatement(sql);
    stmt.setString(1,str);
    ResultSet rs = stmt.executeQuery();
    List<Clothes> data=new ArrayList<>();
    while(rs.next()) {
        Clothes c=new Clothes();
        c.setClothes_id(rs.getInt("cloth_id"));
        c.setCategory(rs.getString("category"));
        c.setType(rs.getString("type"));
        c.setStyle(rs.getString("style"));
    }
}
```

```
        c.setColour(rs.getString("colour"));
        c.setSize(rs.getInt("size"));
        c.setCost(rs.getInt("cost"));
        data.add(c);}
```

#### 4.2.7 Query to select clothes from Brand

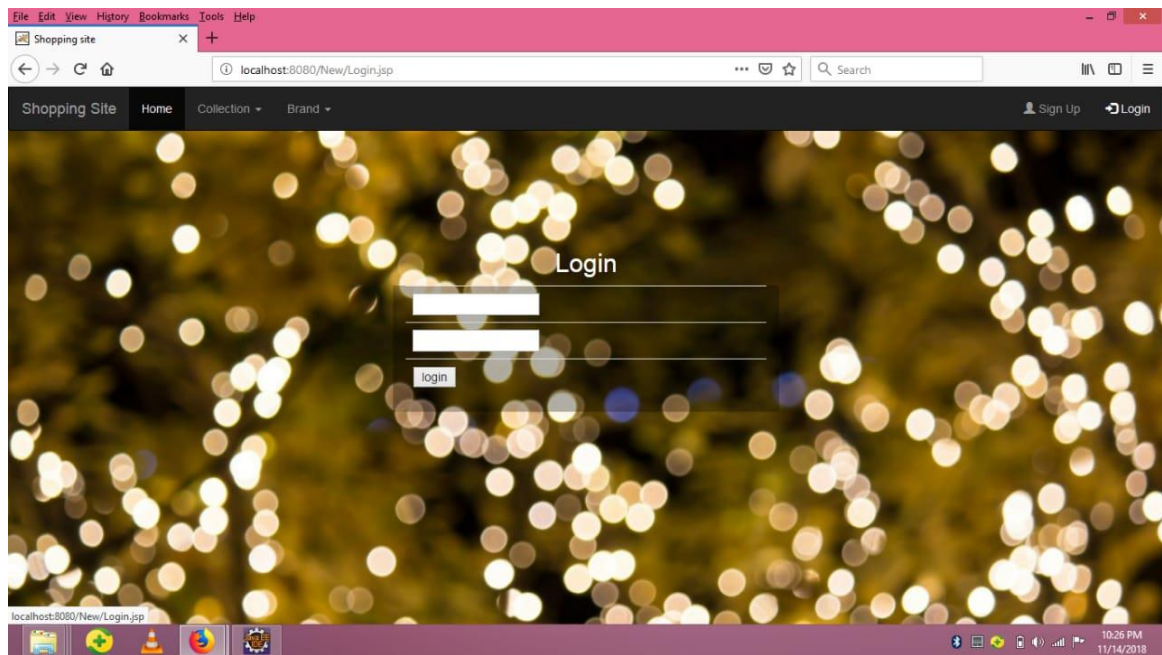
```
public List<Clothes> fetchCollection1(String str) throws SQLException
{
    String sql="select * from clothes where brand_name=?";
    PreparedStatement stmt = con.prepareStatement(sql);
    stmt.setString(1,str);
    ResultSet rs = stmt.executeQuery();
    List<Clothes> data=new ArrayList<>();
    while(rs.next()) {
        Clothes c=new Clothes();
        c.setClothes_id(rs.getInt("cloth_id"));
        c.setCategory(rs.getString("category"));
        c.setType(rs.getString("type"));
        c.setStyle(rs.getString("style"));
        c.setColour(rs.getString("colour"));
        c.setSize(rs.getInt("size"));
        c.setCost(rs.getInt("cost")); data.add(c);
    }
}
```

---

## 4.3 APPLICATIONS OF PROJECT WORK

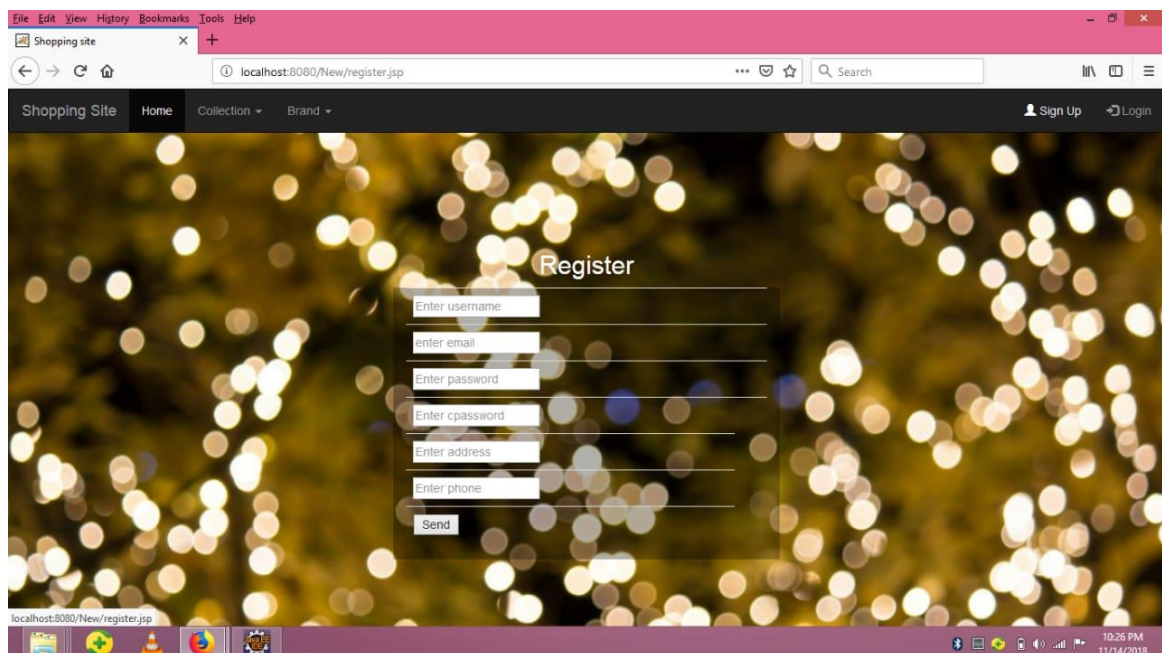
- Viewing Products
- Selecting them from a wide range of Brands and Collections

## 4.3 DISCUSSION OF RESULT



**Fig 4.4.1 Login page**

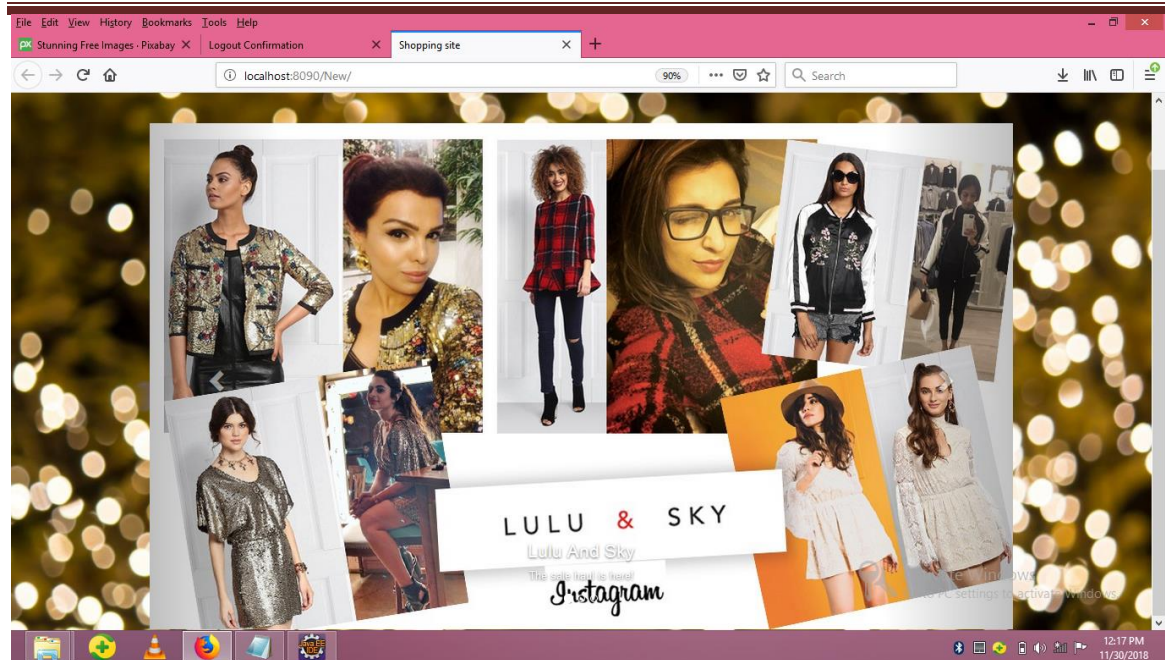
- This displays the login page for the user to login.



**Fig 4.4.2 Register page**

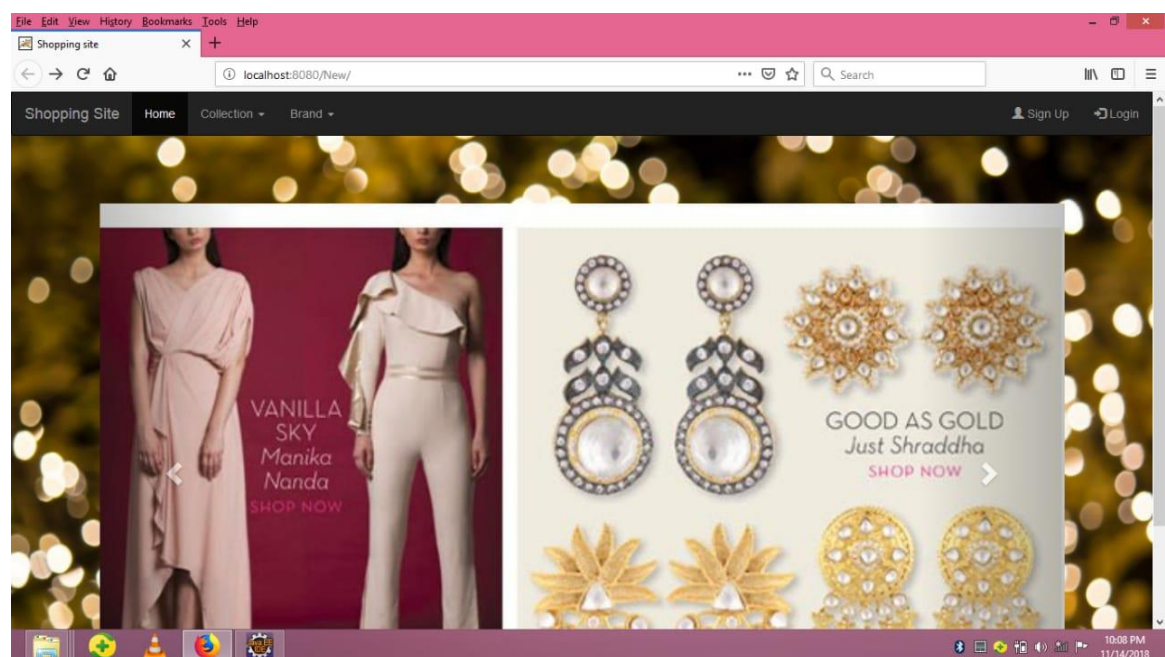
- This displays the register page for the user to register





**Fig 4.4.3 HomePage1**

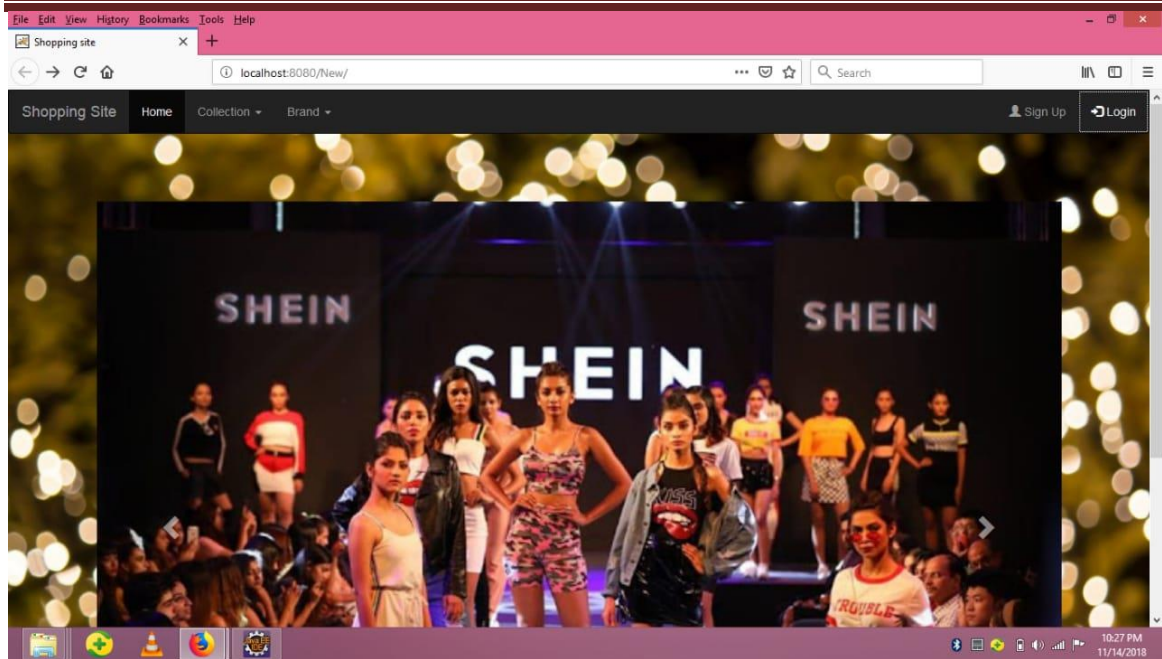
- Displays the brand lulu and sky in the home page.



**Fig 4.4.4 HomePage2**

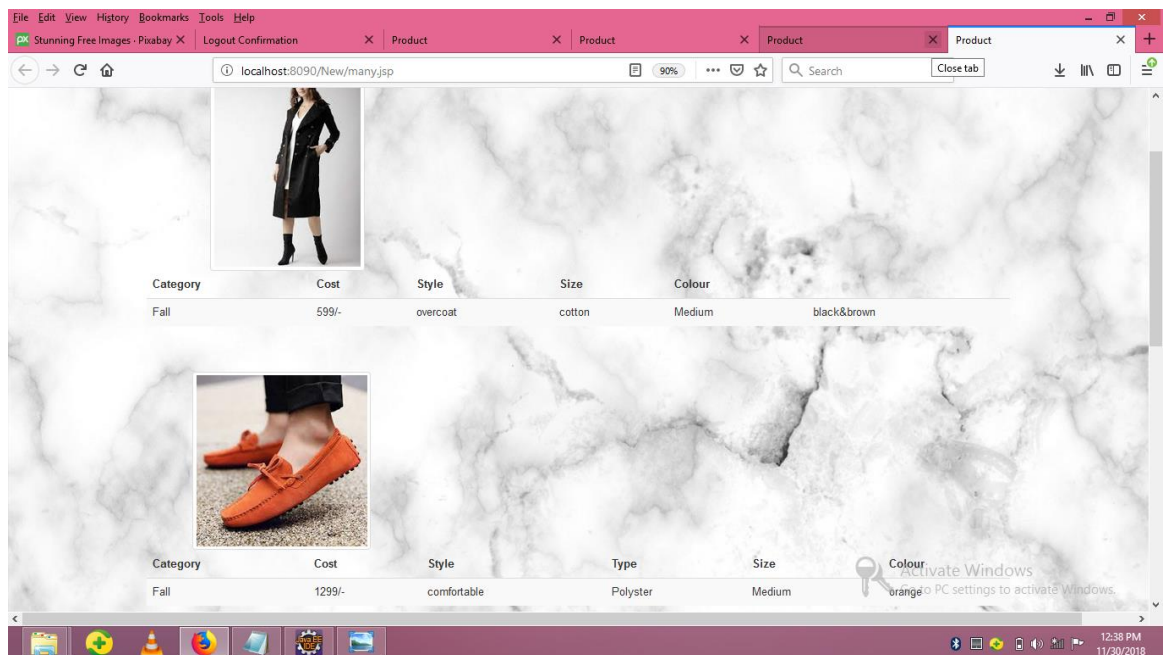
- This displays the brand Pernia Pop Up Shop in the home page





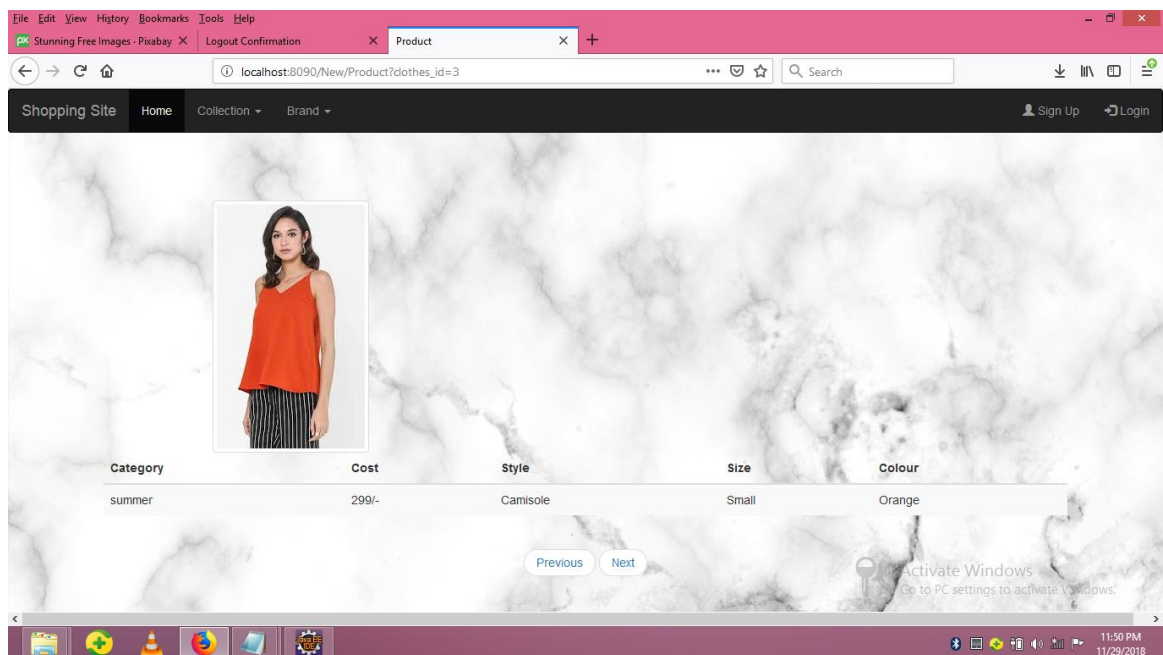
**Fig 4.4.5 HomePage3**

- Displays the brand shein in the home page.



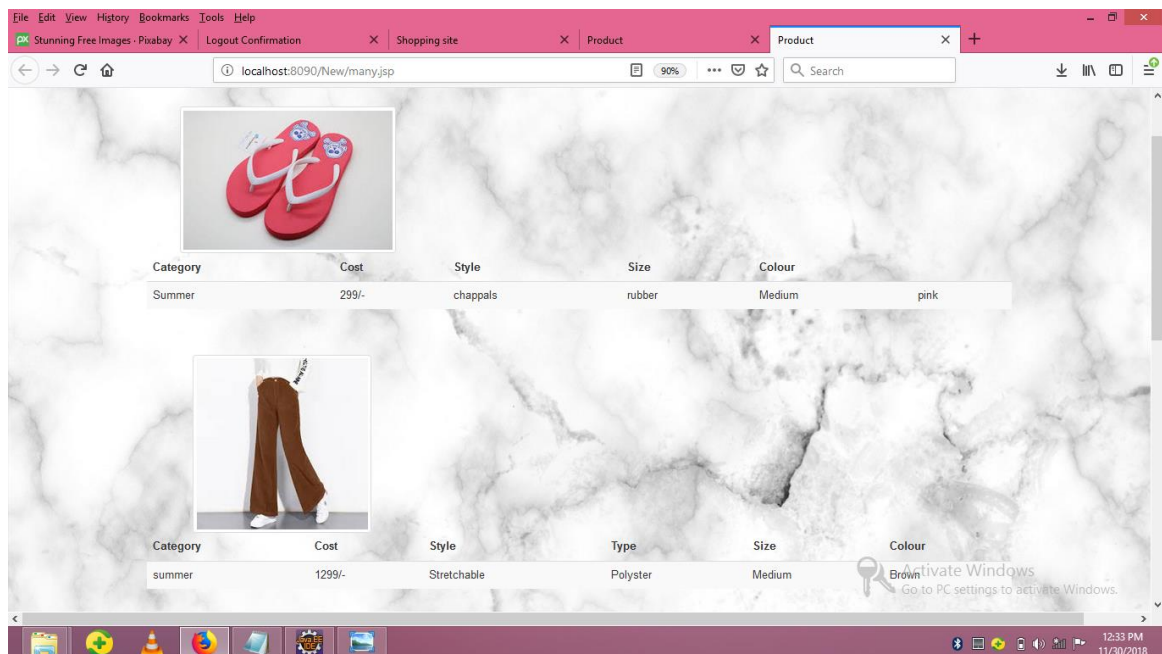
**Fig 4.4.6 Product display1**

- Displays the products under the collection fall



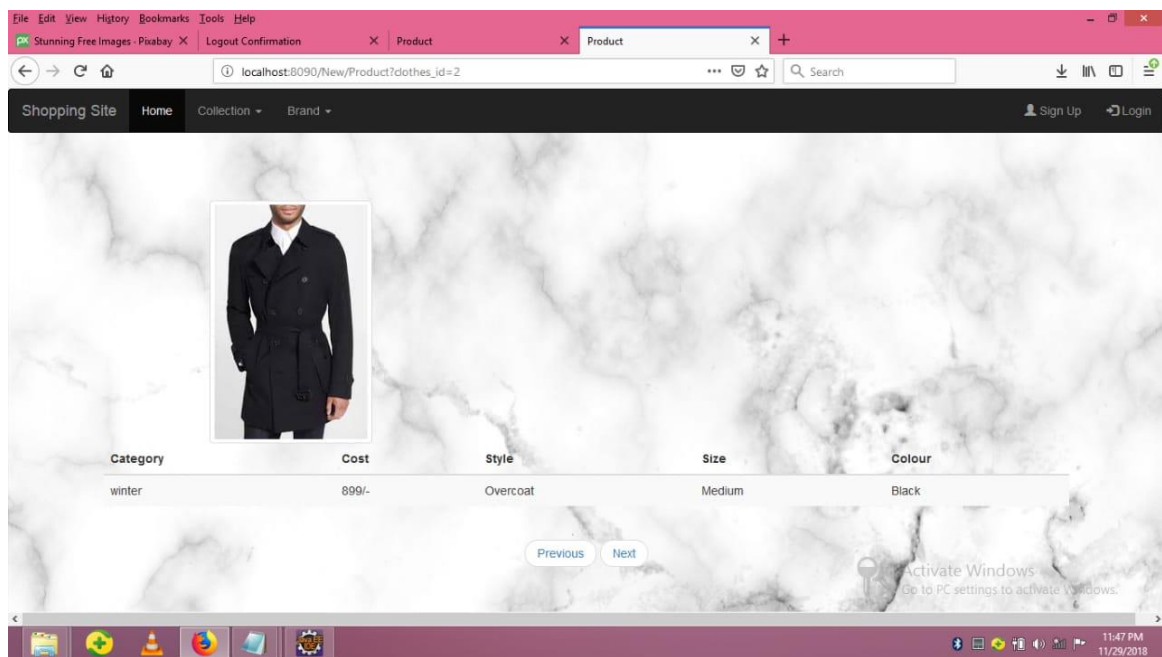
**Fig 4.4.7 Summer product display1**

➤ Display the product under the collection summer

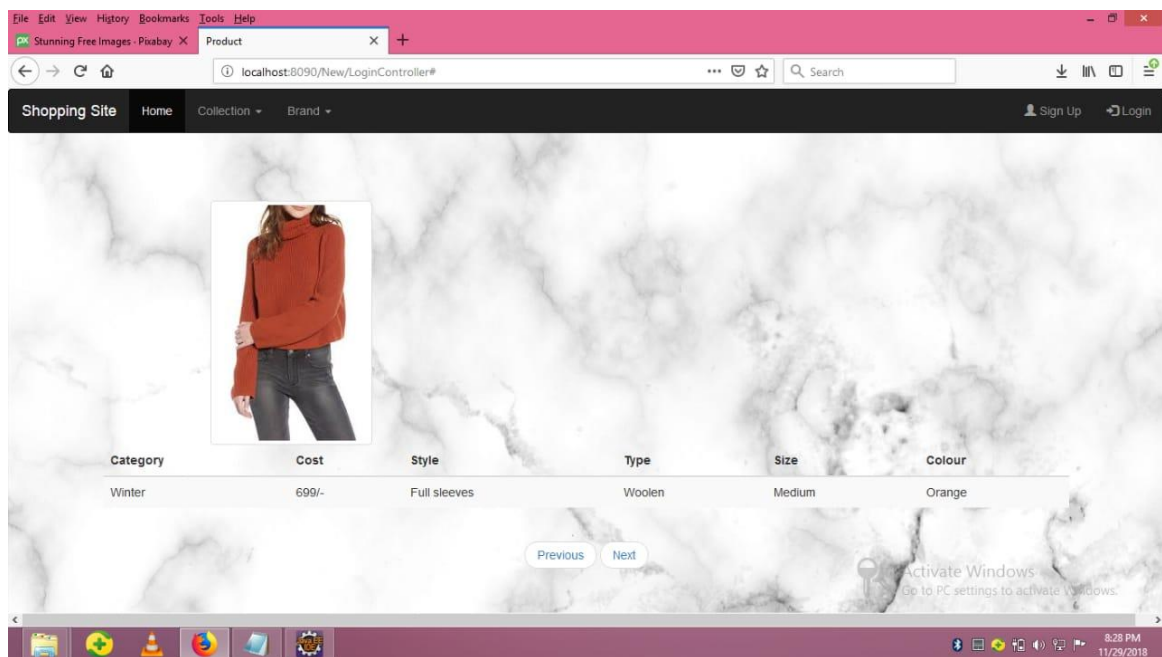


**Fig 4.4.8 Summer product display2**

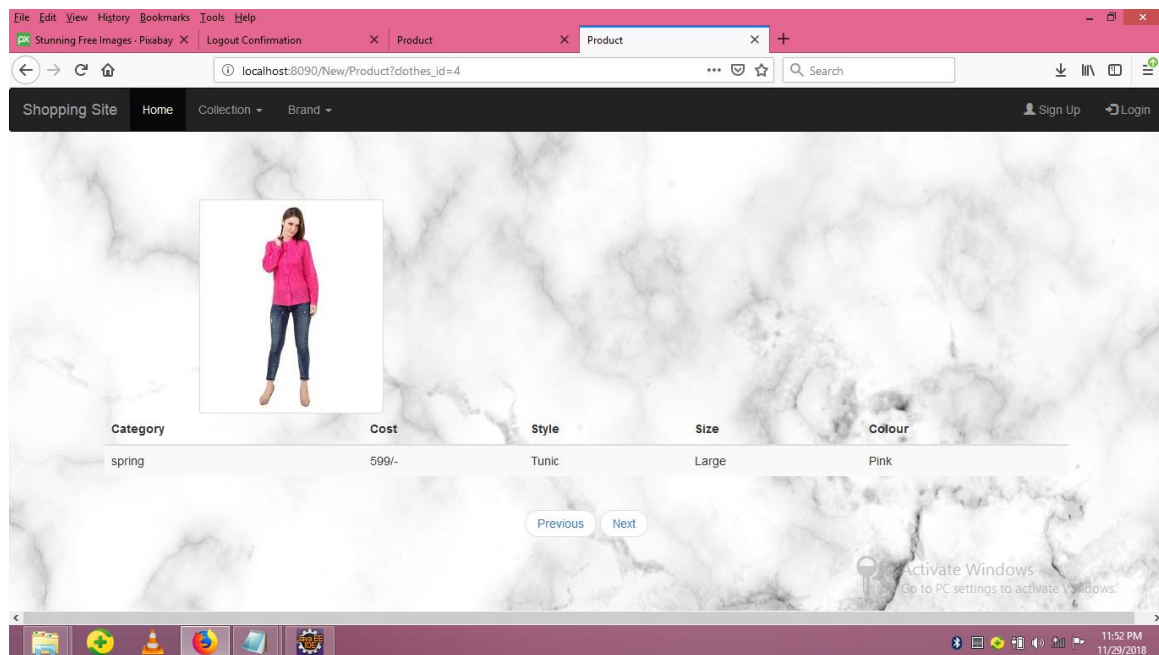
Display the product under the collection summer

**Fig 4.4.9 Winter product display1**

- Displays the product under the collection winter

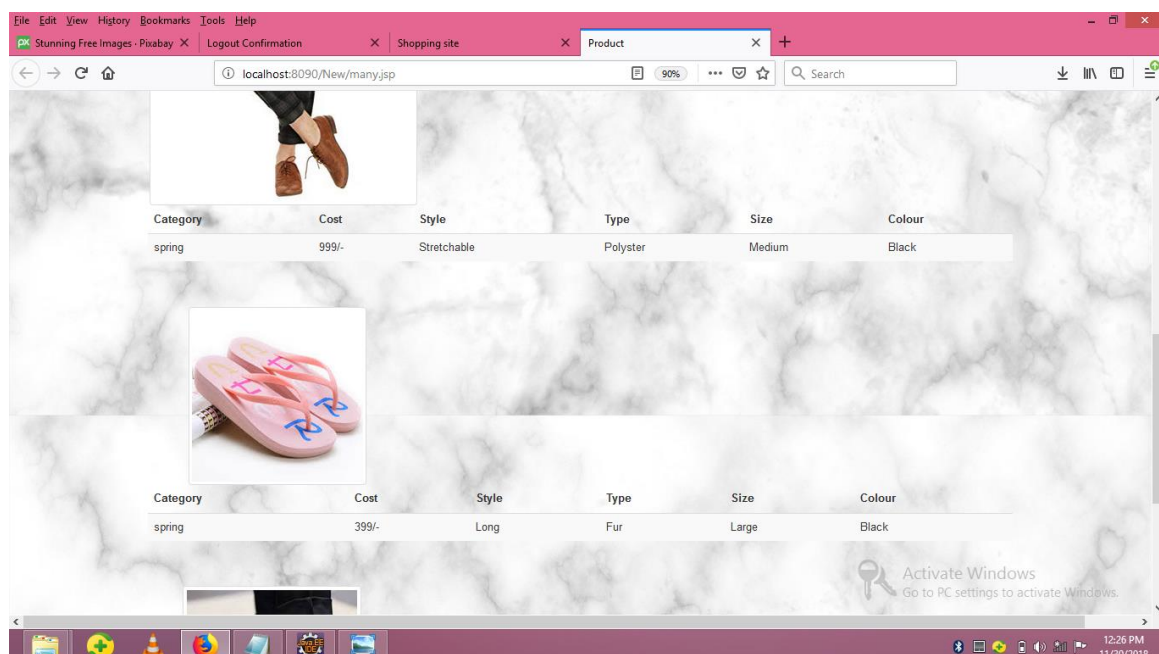
**Fig 4.4.10 Winter Product Display2**

- Displays the product under the collection winter



**Fig 4.4.11 Spring product display1**

- Display the product under the collection Spring



**Fig 4.4.12 Spring product display2**

- Display the product under the collection Spring

## **CHAPTER 5**

### **CONCLUSION AND FUTURE ENHANCEMENTS**

#### **CONCLUSION**

This project is only a humble venture to satisfy the needs and to make the structure and working of online shopping simpler and user-friendly. Several user friendly coding have also been adopted. This project shall prove to be a powerful package in satisfying all the requirements of the college.

Given the right guidance and support, we hope to enhance the it's applications and availability.

At the end it is concluded that we have made effort on the following points:

- Made statement of the aims and objectives of the project.
- The description of Purpose, Scope and applicability.
- We define the problem on which we are working the project.
- We describe the requirement Specifications of the system and the actions that can be done on these things.
- We designed user interface related to system.
- Finally the system is implemented and tested accordingly

#### **FUTURE ENHANCEMENTS**

- Printing generated bills
- More facilities for online shopping management like including discounts based on festival season and clearance sale
- Creating an App on play store for our website
- Payment through credit cards and online payment options



## REFERENCES

### Online Reference

- [www.google.com](http://www.google.com)
- [www.stackoverflow.com](http://www.stackoverflow.com)
- [www.bootstrap.com](http://www.bootstrap.com)
- [www.java.sun.com](http://www.java.sun.com)

