**XYZstore**

**(A WEB-BASED E-Commerce APPLICATION)**

**BY**

**Aashish Kumar Thapa**

**T.U Roll No.: 8621/18**

Kantipur College of Management and Information Technology

*A Summer Project Documentation Submitted to*

**Faculty of Management, Tribhuvan University**

In partial fulfillment of the requirements for the degree of

**Bachelor of Information Management**

**May, 2022**

Table of Contents

[Chapter 1 : Introduction 1](#_Toc104217156)

[1.1 Background 1](#_Toc104217157)

[1.2 Objective of study 1](#_Toc104217158)

[1.3 Methodology 1](#_Toc104217159)

[1.3.1 Project Framework 1](#_Toc104217160)

[1.3.2 Tools and technology used 1](#_Toc104217161)

[Chapter 2 : Tasks and activities performed 2](#_Toc104217162)

[2.1 Analysis of task: 2](#_Toc104217163)

[2.1.1 Functional requirement analysis 2](#_Toc104217164)

[2.1.1.1 Admin Requirements: 2](#_Toc104217165)

[2.1.1.2 Customer Requirements: 3](#_Toc104217166)

[2.1.1.3 Other general requirements: 3](#_Toc104217167)

[2.2 Activities performed: 4](#_Toc104217168)

[2.2.1 Database design 4](#_Toc104217169)

[2.2.2 Architectural design 4](#_Toc104217170)

[2.2.3 Software components: 5](#_Toc104217171)

[2.2.3.1 Product management system 5](#_Toc104217172)

[2.2.3.2 Customer UI 9](#_Toc104217173)

[2.2.3.3 Product ordering system. 10](#_Toc104217174)

[2.2.3.4 Price tracking system 11](#_Toc104217175)

[2.2.3.5 Product analysis 12](#_Toc104217176)

[Chapter 3 : Conclusion 15](#_Toc104217177)

[3.1 Conclusion 15](#_Toc104217178)

**List of Figures:**

[Figure 2‑1 : Use case diagram involving the role of ‘administrator (Seller)’ 2](#_Toc104217181)

[Figure 2‑2 : Use case diagram involving the role of ‘customer’ 3](#_Toc104217182)

[Figure 2‑3 : ER-diagram of database 4](#_Toc104217183)

[Figure 2‑4 : Product view page 5](#_Toc104217184)

[Figure 2‑5 : product insertion form 6](#_Toc104217185)

[Figure 2‑6: product update view 7](#_Toc104217186)

[Figure 2‑7 : product delete view 8](#_Toc104217187)

[Figure 2‑8 : Customer home page 9](#_Toc104217188)

[Figure 2‑9 : product view page 9](#_Toc104217189)

[Figure 2‑10: Customer cart view 10](#_Toc104217190)

[Figure 2‑11 : admin order view 10](#_Toc104217191)

[Figure 2‑12 : price tracking view 11](#_Toc104217192)

[Figure 2‑13 : best product by order view 12](#_Toc104217193)

[Figure 2‑14 : best product by quantity view 13](#_Toc104217194)

[Figure 2‑15 : best brand view 14](#_Toc104217195)

# Introduction

## Background

This project is an E-commerce web application that create platform for customer to place an order for products over the internet using the web browser. The administrators (Seller) then respond to orders and deliver the product to the end user. This system tries to provide service similar to existing applications like Amazon, DARAZ to customer as well as conversion of customer’s data to information.

## Objective of study

Objective of the project is to create an easy to use website for doing transactions of the products, to keep track of the orders, to know best selling products, to analyze the bestselling brand, to keep track of the price of products.

## Methodology

### Project Framework

Reuse-oriented software model (Integration and configuration) is applied to achieve the desire objective with the use of web framework spring boot and JavaScript library react JS. This process model has obvious advantage of reducing the amount of software to be developed hence reducing cost and risks. It also leads to faster delivery of the software with the help of collections of objects that are developed as a package to be integrated with a component framework.

### Tools and technology used

* User Interface Designing using FIGMA
* Front-end : React JS library along with HTML , CSS , JavaScript
* Back-end : Spring Boot framework with MYSQL relational database management system
* Code Editor : Spring tool suite and visual studio

# Tasks and activities performed

## Analysis of task:

Expanding a business on an internet brings advantages like increase in size of market, easy to access service from anywhere. Above objective is achieved with web application, which contains functional requirements like product inventory management, easy to use front end UI for customer, order tracking system, as well as modules to bring information of products like best selling product, product price tracking system.

### Functional requirement analysis

The functional requirement describes what the system should do. The Expected users for the software are admin (Seller) and customer. The requirement is further categorized into:

#### Admin Requirements:

1. System access to only authorized administrator.
2. To be able to insert, update and delete the products information.
3. To check and keep track of order of customer.

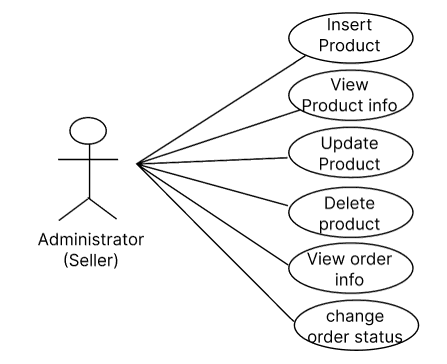


Figure 2‑1 : Use case diagram involving the role of ‘administrator (Seller)’

#### Customer Requirements:

1. To be able to view and place order of a multiple product at a time
2. To be able to maintain cart where multiple product can be placed and removed if necessary before placing order.
3. To be able to search the product with product name, categories and filter the result with price range they want.

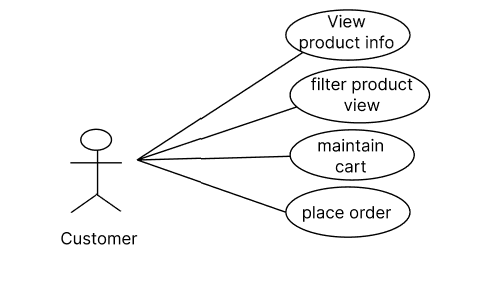


Figure 2‑2 : Use case diagram involving the role of ‘customer’

#### Other general requirements:

1. The system shall keep track of price of a product when it is updated, where it provide information of when price has gone lowest, highest and all prices according to updated timestamp.
2. The system shall generate the information of highest sold product and highest sold brand.

## Activities performed:

### Database design

To make a dynamic website, it requires a database to store data about products, orders of products, customer information. For this project relational database MYSQL is used which store data in separate table which is expressed in following ER diagram.

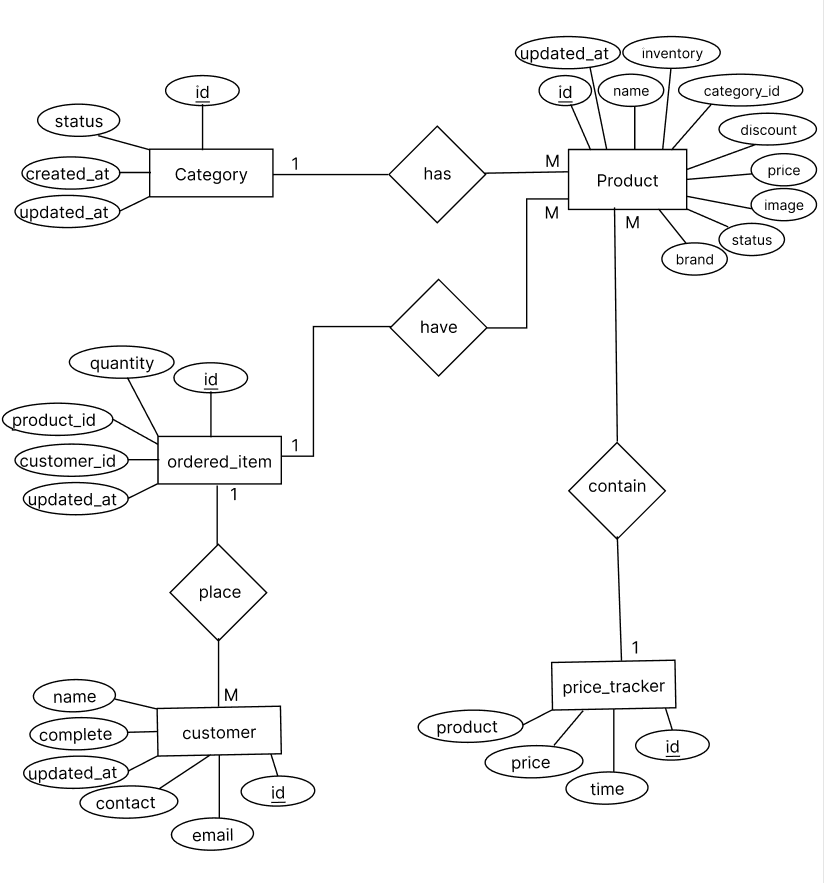


Figure ‑ : ER-diagram of database

### Architectural design

The architectural pattern followed for admin end is MVC pattern where as for customer UI, REST architectural style is used, which then communicate with single page application made by using library of JavaScript react JS.

### Software components:

#### Product management system

Its primary objective is to allow admin to view, insert, delete and update the product information in or from the database.

##### View Product

As multiple products may belong to same category and to reduce data redundancy category is placed in separate table and foreign key is used by product table. The view page of product displays data as table which by default only displays five distinct products as shown in Figure 2‑4. The image column contains anchor tag which display the image associated with image name in database on new tab of browser. To view other product a search bar is designed which on submit search the products by name and display empty table when it found nothing.

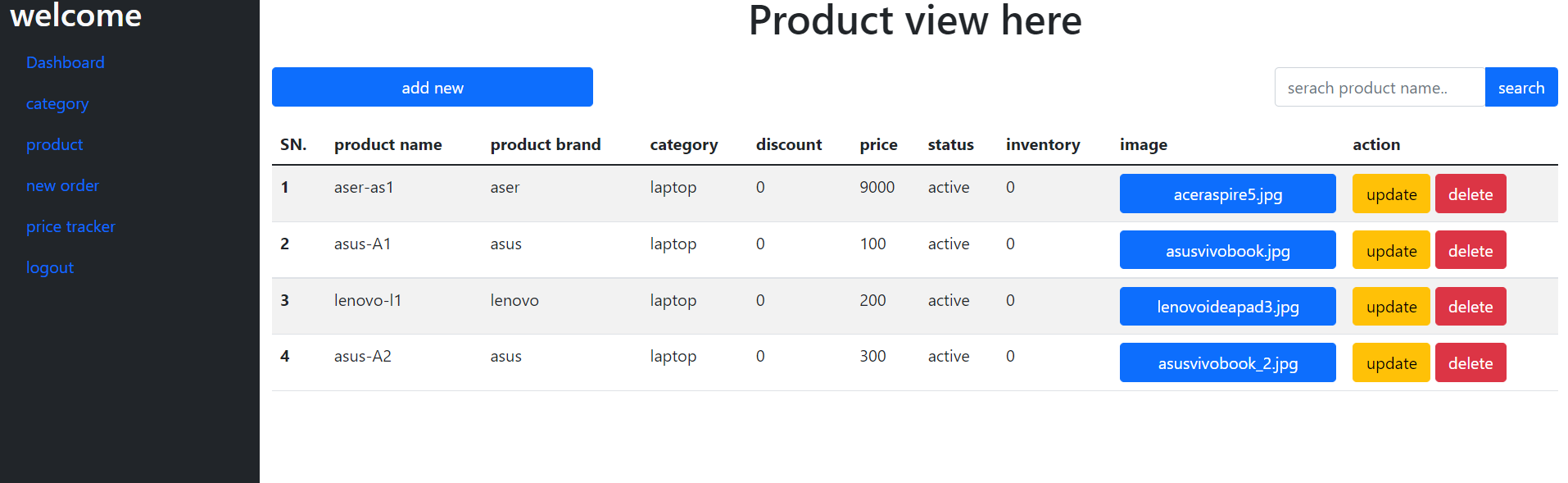


Figure ‑ : Product view page

##### Insert Product

To insert the data of new product, HTML form is built which takes all required information as shown in Figure 2‑5. The drop down menu is used to insert the category of a product which contains the name of only that category having active status. When the form is successfully submitted it saves the data on database and return product view page with only one product, that is recently entered data. It will throw the exception if the image name is already acquired by any other product, to prevent from file being replaced during submission.

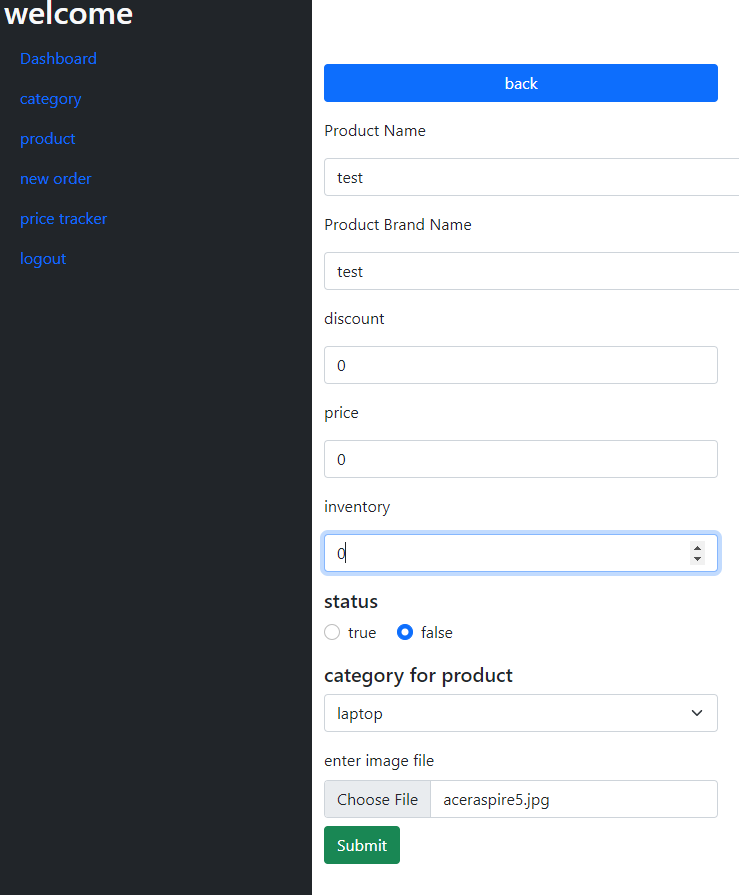


Figure ‑ : product insertion form

##### Update Product

To update a product admin should follow the link on action column as shown in Figure 2‑4 which redirect to update view that contain filled up html form (Figure 2‑6) with all previous data from database including existing image. Once the admin replace the data with new one and submit, the previous data will be replaced with new one in database and redirect to the view product page with only recently updated data.

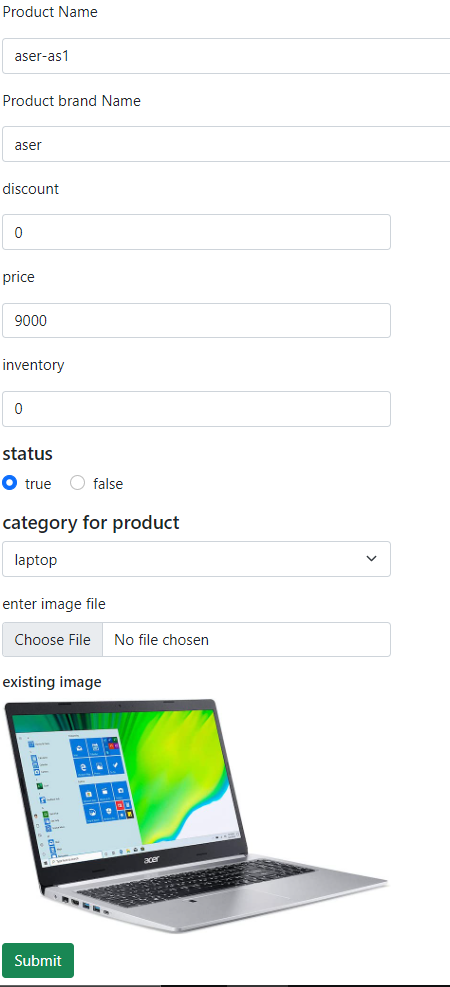


Figure ‑: product update view

##### Delete Product

When delete button of action column of product view (Figure 2‑7) is pressed, it triggers confirm box of JavaScript when ok button is pressed it delete the product data from the database but if cancel button is pressed it will not delete the product data.

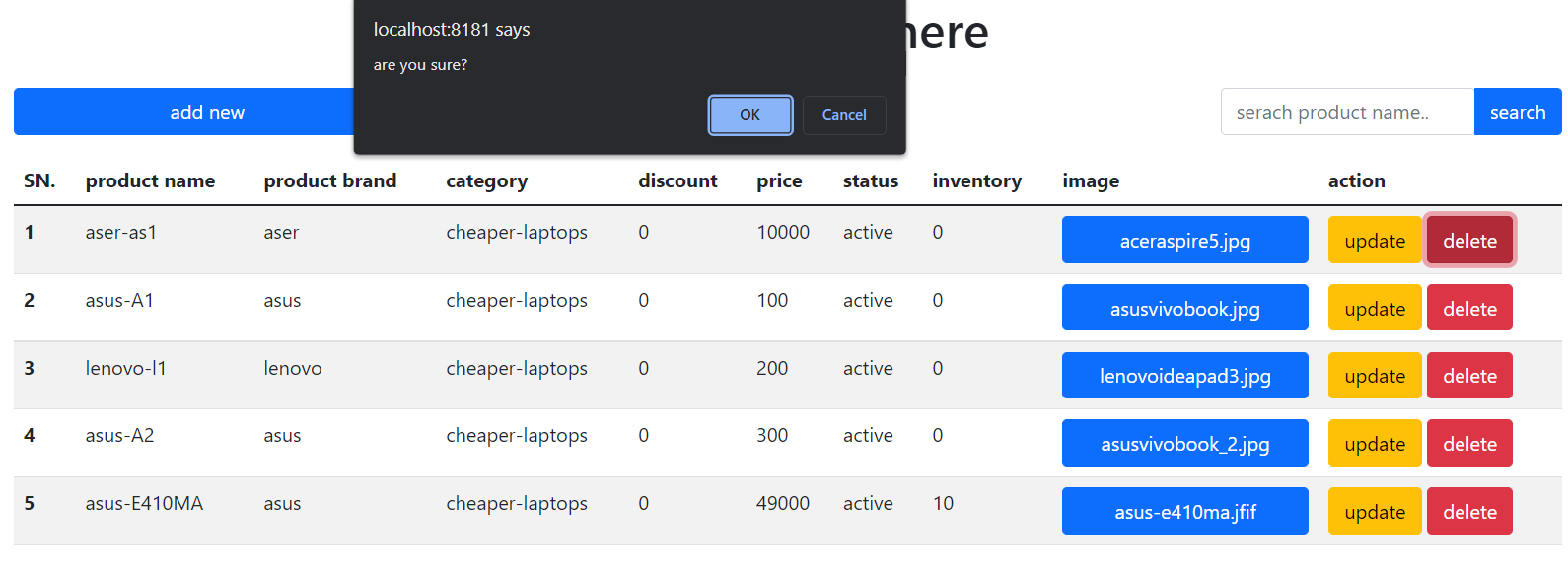


Figure ‑ : product delete view

#### Customer UI

The objective of customer UI is to create easy to use and understand user interface to provide good user interface. The home page (Figure 2‑8) contains card design that contains category link and respective link of best selling product of that category. The category link then takes to product page (Figure 2‑9) that contain all products associated with that category where customer can add the item to cart or further filter the product as their desire maximum price.

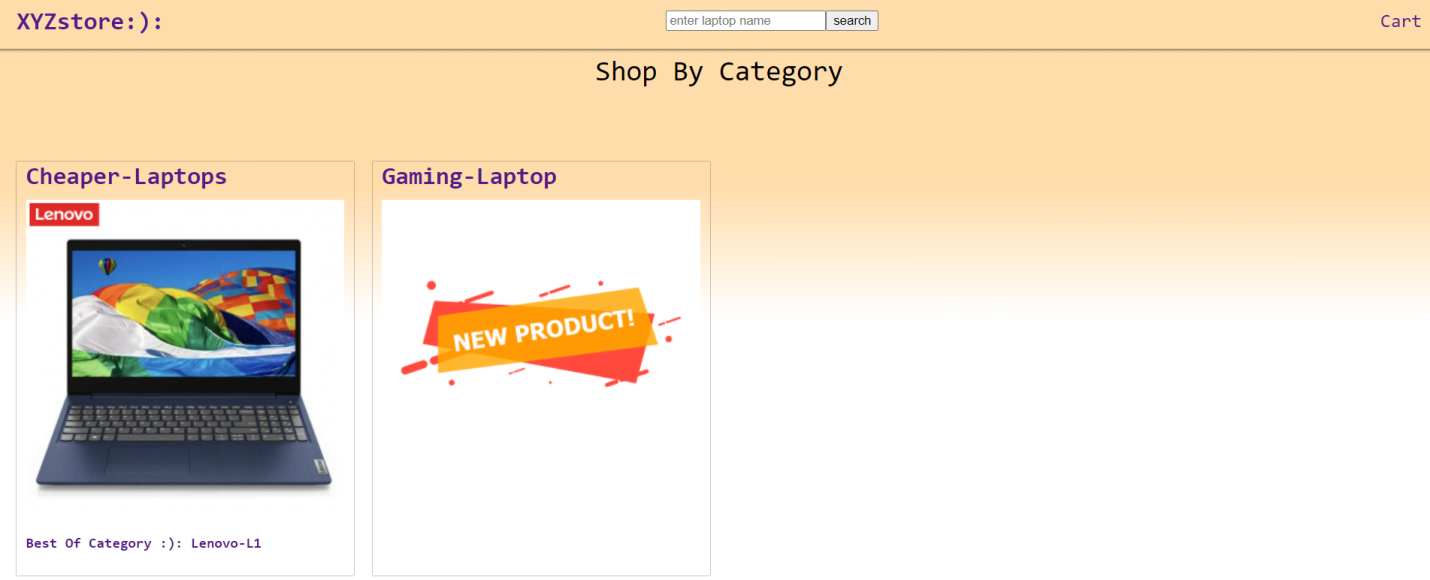


Figure ‑ : Customer home page

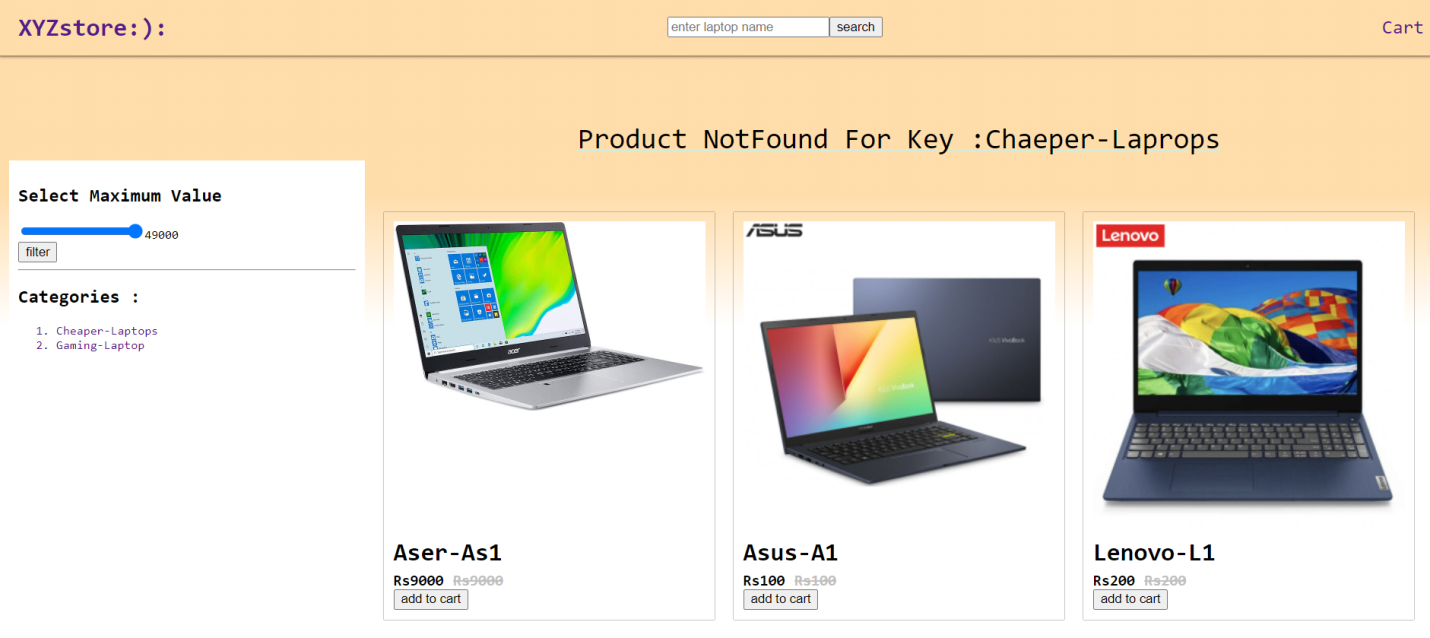


Figure ‑ : product view page

#### Product ordering system

The ordering process starts when customer began to add the product on cart which can be achieved by help of add to cart button as shown in Figure 2‑9. The cart data is stored as react-state which will not be saved if browser or tab is closed. The user can add multiple products at a time; remove from cart before placing order or increase the quantity of a product they added as shown in Figure 2‑10. The customer are then required to place email and number to contact during order delivery and name to identify before placing the order. After submit button is clicked the order will be stored in database and user cart will be cleared. Once the order is placed by the customer using customer UI, the customer data will be saved in database, the admin can view all the order as shown in Figure 2‑11.

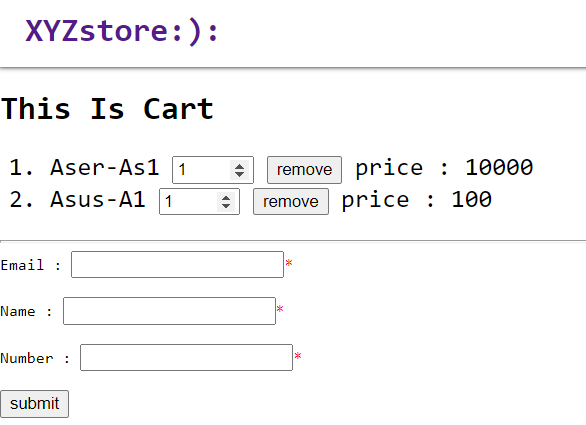


Figure ‑: Customer cart view

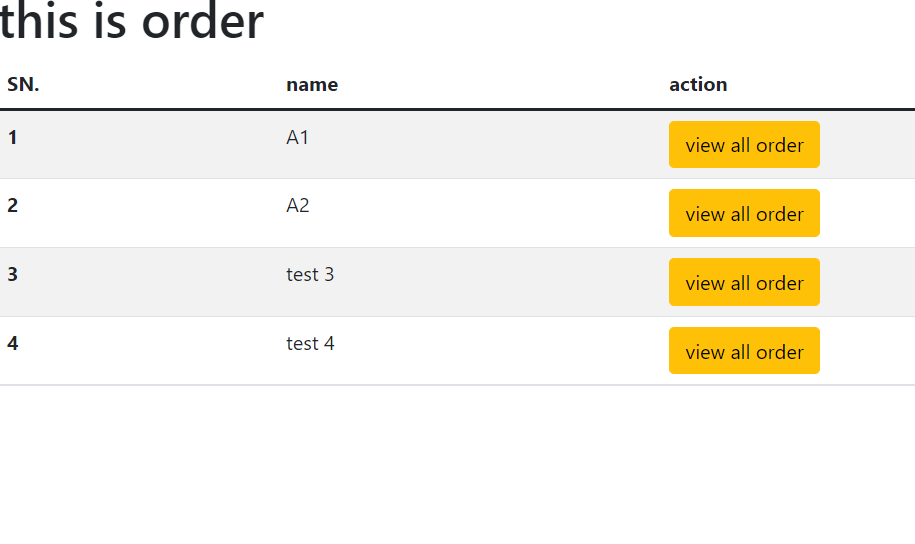


Figure ‑ : admin order view

#### Price tracking system

The objective or this system if to keep track of all the changes in price of a product at updated timestamp, calculate the timestamp when price or a product is highest, lowest, current price and all the price at updated timestamp and represented in line graph using [*google-linechart*](https://developers.google.com/chart/interactive/docs/gallery/linechart) as shown in Figure 2‑12. Whenever the price of product is updated, the updated price with timestamp is maintained on different table through which we can get information like seasonal affect on price of product.

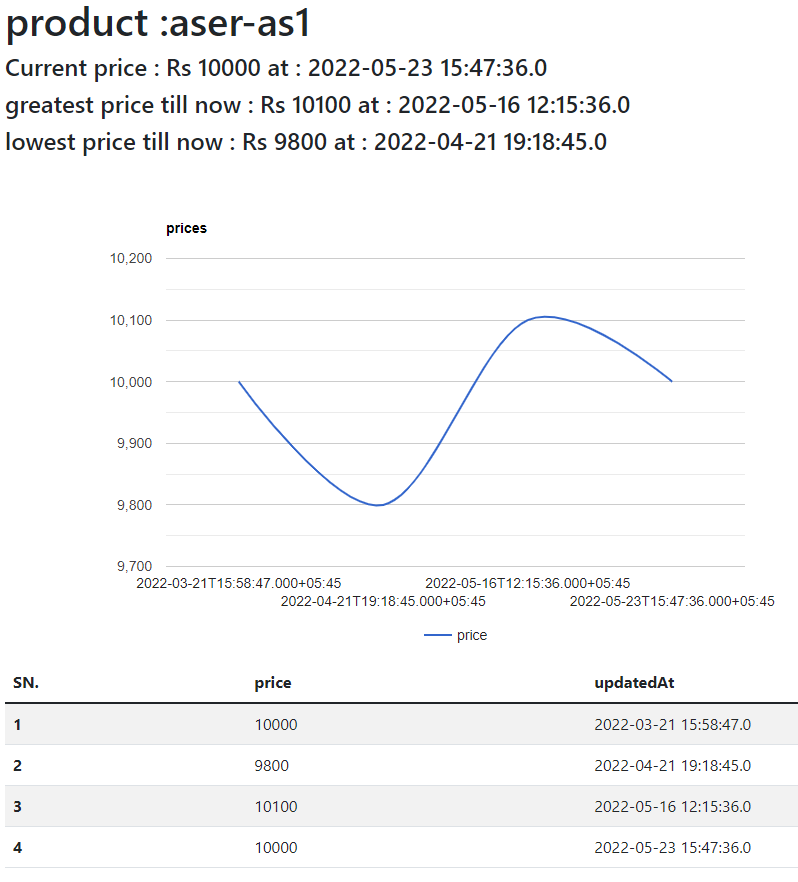


Figure ‑ : price tracking view

#### Product analysis

With availability of data about order of products on a database, it is now possible to determine the best selling product and bestselling brand of product. This helps the seller for inventory management, where they can add more inventories of best selling product and reduce the inventory of least selling product or exclude them if they don’t provide business value. The admin can easily access this information from admin dashboard. Best selling product is categorized by number of order placed (Figure 2‑13) on product or quantity sold (Figure 2‑14) and is represented in pre chart with help of [*google-piechart*](https://developers.google.com/chart/interactive/docs/gallery/piechart). Similar activities are performed for brand as well (Figure 2‑15).

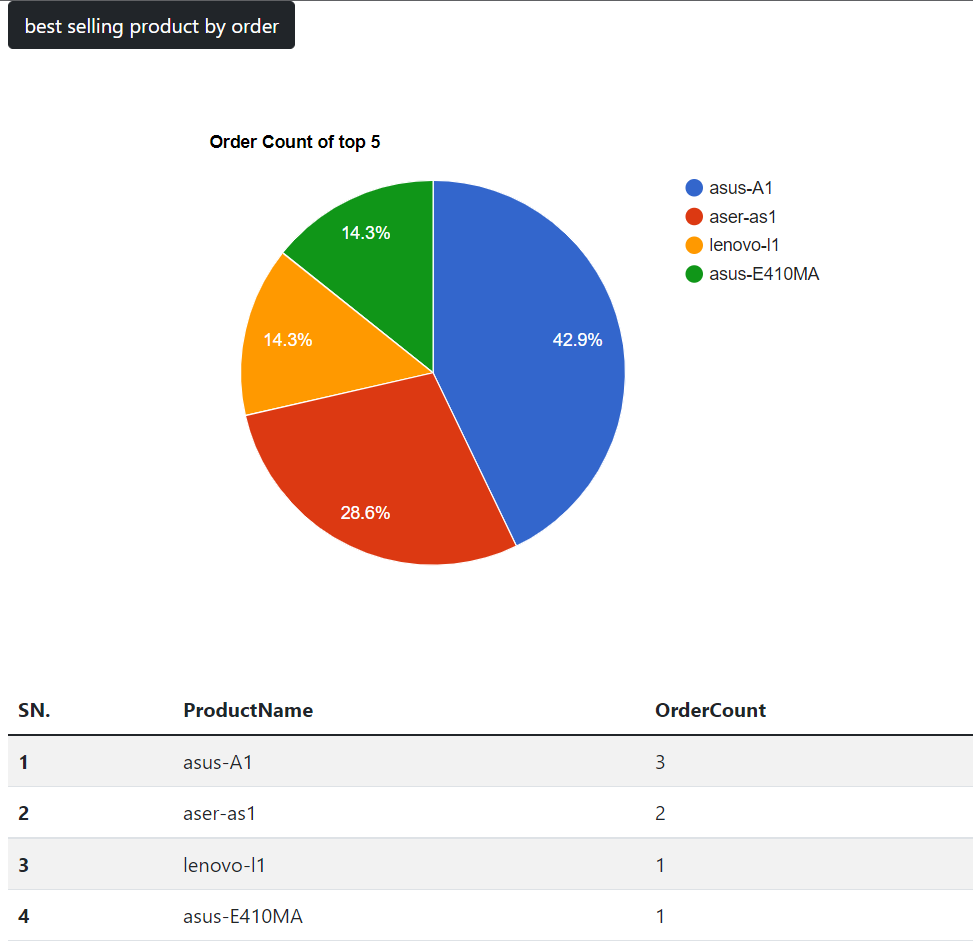


Figure ‑ : best product by order view

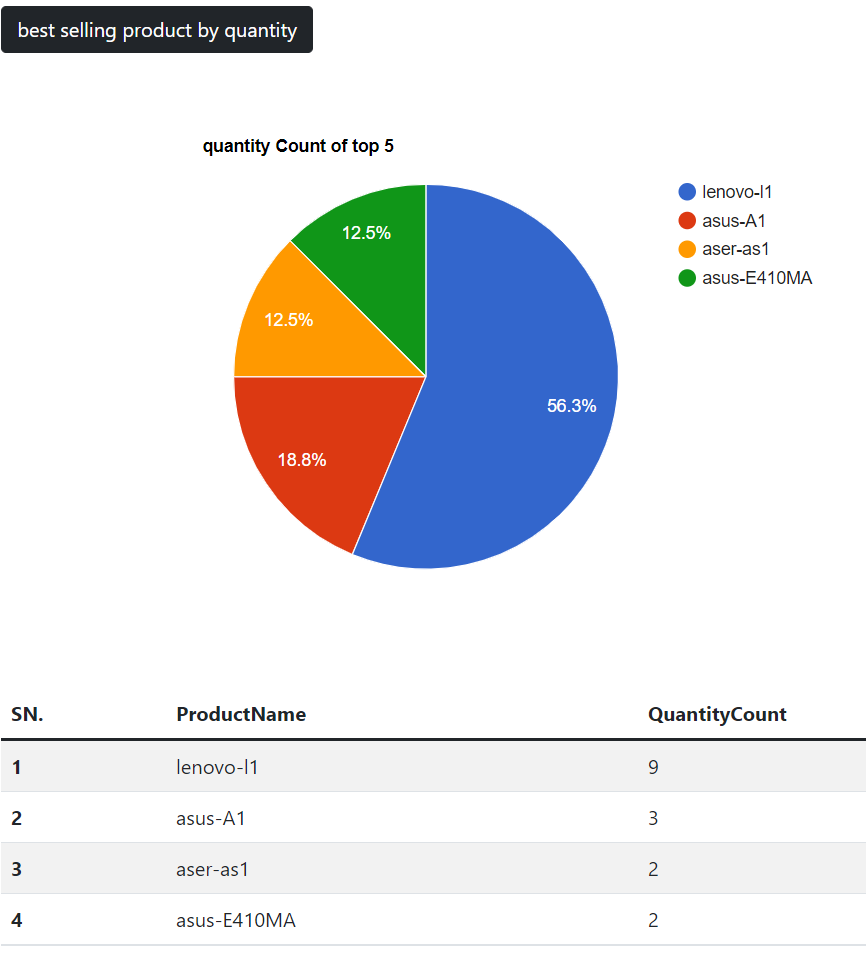


Figure ‑ : best product by quantity view

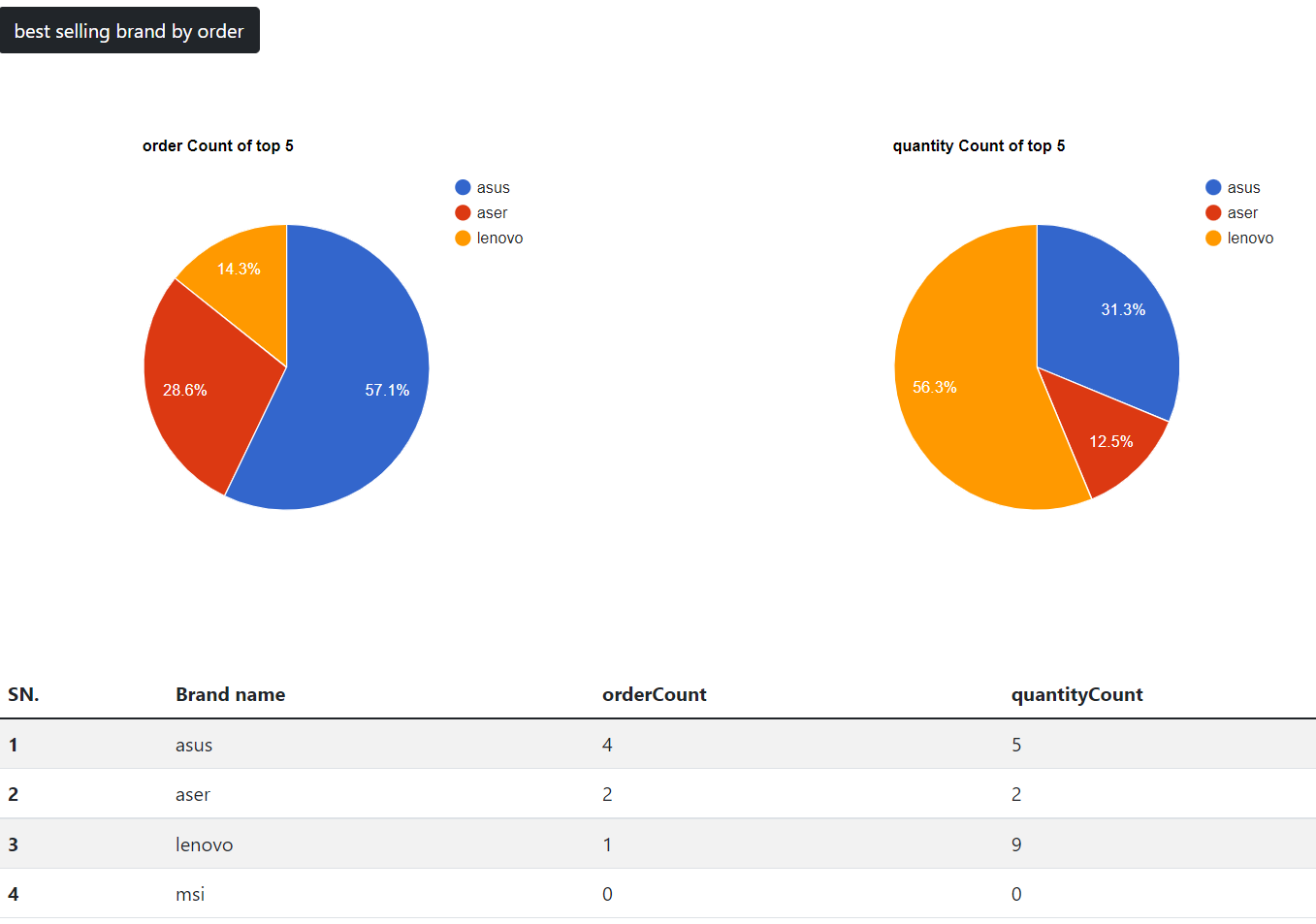


Figure ‑ : best brand view

# Conclusion

## Conclusion

As per the objectives, this application helps the organization to grow on an internet creating platform for seller to manage, and show their products along with generation of information like best product, best brand as well as the changes of price of a product.