Build & Buy: Custom PC Builder and Online Store for Components

1.0. INTRODUCTION

1.1 Purpose

The Build & Buy platform aims to offer users an easy way to build custom PCs by selecting compatible components, purchasing them, and tracking their orders. It also provides tools for users to compare products, check performance, and ensure everything works together smoothly. The platform aims to provide a complete solution for anyone looking to build or buy PC components online.

1.2 Project Proposal

The Build & Buy platform will include several key modules: a PC Builder for customizing PCs with compatible parts, an Inventory Management system for updating and tracking components, and an E-commerce section for ordering and securing payments. Additional features include component comparison, performance checks, and customer services like reviews, polls, and coupon generation. This project will create a seamless experience for users to build, buy, and manage their PC components.

2.0. OVERALL DESCRIPTION

2.1 System Environment
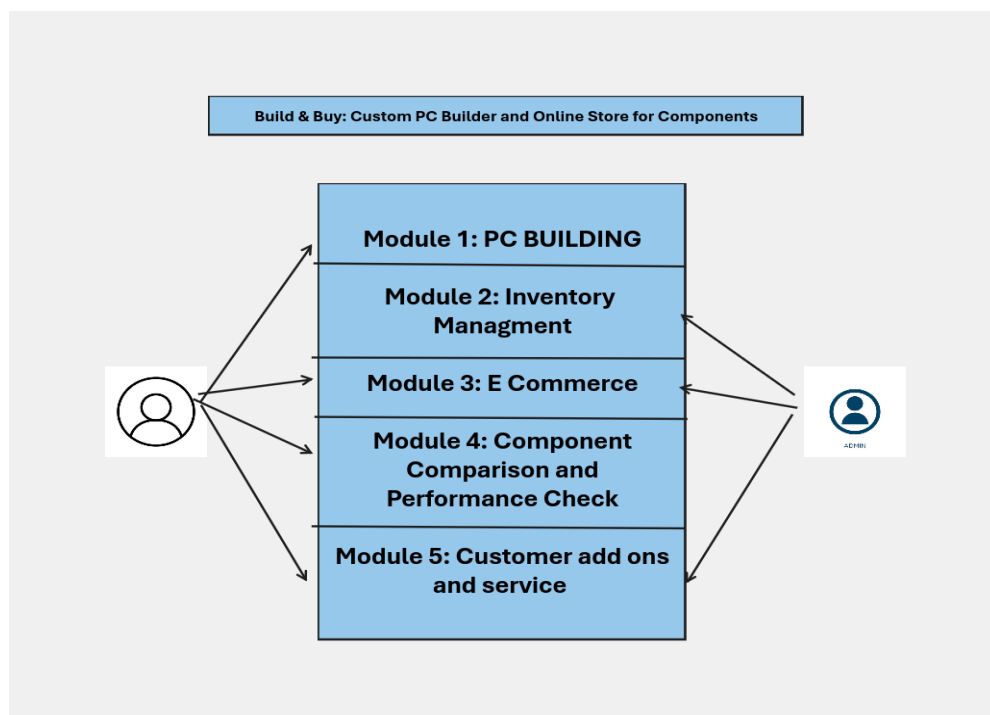
2.2 Functional and Nonfunctional Requirements

2.2.1 MODULE 1: PC BUILDER

This module allows users to customize and assemble their PC by selecting various parts.
1. Component Selection
2. Quantity and Pricing
3. Printing and Screenshot
4. Compatibility Check (For Intel CPU all the Intel-compatible products will be shown)

2.2.2 MODULE 2: Inventory Management

Manages all PC components available in the warehouse and updates them.
1. Adding new components
2. Updating components
3. Stock summary (Giving a summary of the current products)
4. Category management for different types of products.

2.2.3 MODULE 3: E commerce

Purchasing different types of pc components
1. Ordering components
2. Shopping cart
3. Order Tracking
4. Secure payment/ order history

2.2.4 MODULE 4: Customer Add-Ons and Service

Providing a customer community
1. Customer can post their experience as a review
2. Participate in polls
3. Request for a component / Filing a complaint / Request for a component
4. Coupon Generation

2.2.5 MODULE 5: Component Comparison and Performance Check

Customer can choose the best product for him/er
1. Customer Can Compare and select the same product between different companies, prices, features
2. Using a bottleneck calculator, Customers can check if their PC has a bottleneck.
3. Customers can see their selected product's total watt consumption
4. Benchmark Comparison / Recommended Upgrade
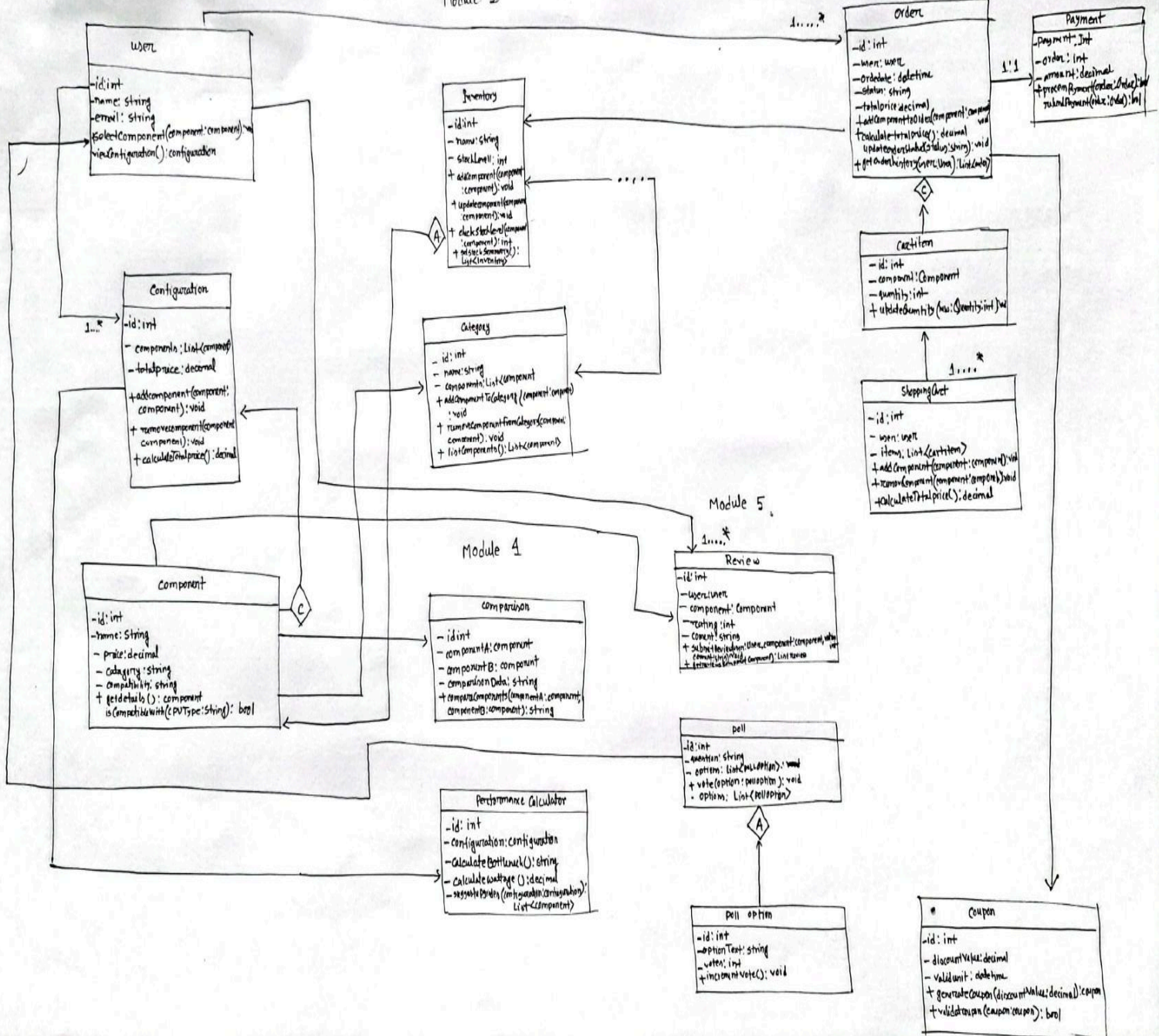
2.3 Class Diagram

Public → '+'
Private → '-'

Module 1

User
- id: int
- name: string
- email: string
+ selectComponent(component: component)
+ viewConfiguration(): configuration

Configuration
- id: int
- components: List<component>
- totalprice: decimal
+ addcomponent(component: component): void
+ removecomponent(component: component): void
+ calculateTotalprice(): decimal

Component
- id: int
- name: String
- price: decimal
- category: string
- compatibility: string
+ getdetails(): component
+ isCompatibleWith(cPUType: String): bool

Module 2

Inventory
- id: int
- name: string
- stockLevel: int
+ addComponent(component: component): void
+ updateComponent(component: component): void
+ checkStockLevel(component: component): int
+ getStockSummary(): List<Inventory>

Category
- id: int
- name: string
- components: List<component>
+ addComponentToCategory(category, component: component): void
+ removeComponentFromCategory(component: component): void
+ listComponents(): List<component>

Module 3

Order
- id: int
- user: user
- orderDate: datetime
- status: string
- totalprice: decimal
+ addComponentToOrder(component: component): void
+ calculateTotalprice(): decimal
+ updateOrderStatus(status: string): void
+ getOrderHistory(user: user): List<order>

Payment
- payment: int
- order: int
- amount: decimal
+ processPayment(order: order): void
+ refundPayment(ordex: order): bool

CartItem
- id: int
- component: Component
- quantity: int
+ updateQuantity(new: Quantity: int): void

ShoppingCart
- id: int
- user: user
- items: List<cartitem>
+ addComponent(component: component): void
+ removeComponent(component: component): void
+ calculateTotalPrice(): decimal

Module 4

Comparison
- id: int
- componentA: component
- componentB: component
- comparisonData: string
+ compareComponents(componentA: component, componentB: component): string

Performance Calculator
- id: int
- configuration: configuration
+ calculateBottleneck(): string
+ calculateWattage(): decimal
+ suggestUpgrades(configuration: configuration): List<component>

Module 5

Review
- id: int
- user: user
- component: Component
- rating: int
- content: string
+ submitReview(user: user, component: component, value: content): void
+ getReviews(component: component): List<Review>

Poll
- id: int
- question: string
- options: List<polloption>: void
+ vote(option: polloption): void
- options: List<PollOption>

Poll option
- id: int
- optionText: string
- votes: int
+ incrementVote(): void

Coupon
- id: int
- discountValue: decimal
- validUntil: datetime
+ generateCoupon(discountValue: decimal): coupon
+ validateCoupon(coupon: coupon): bool

2.4 Tools And Technologies

Database: MySQL

Frontend: HTML, CSS, JavaScript

Backend: Laravel 11

2.5 Compatibility

WILL BE ADDED LATER

3.0. IMPLEMENTATIONS
3.1 Sprint 2

Fig 3.1.1Main page as an anonymous user for custom pc building (purchasing in the module)

Fig 3.1.2 Selecting a particular type of component from the database of different components

Fig 3.1.3 Component Database Table

3.1.1 Implementation of MODULE 1 (PC BUILDER)
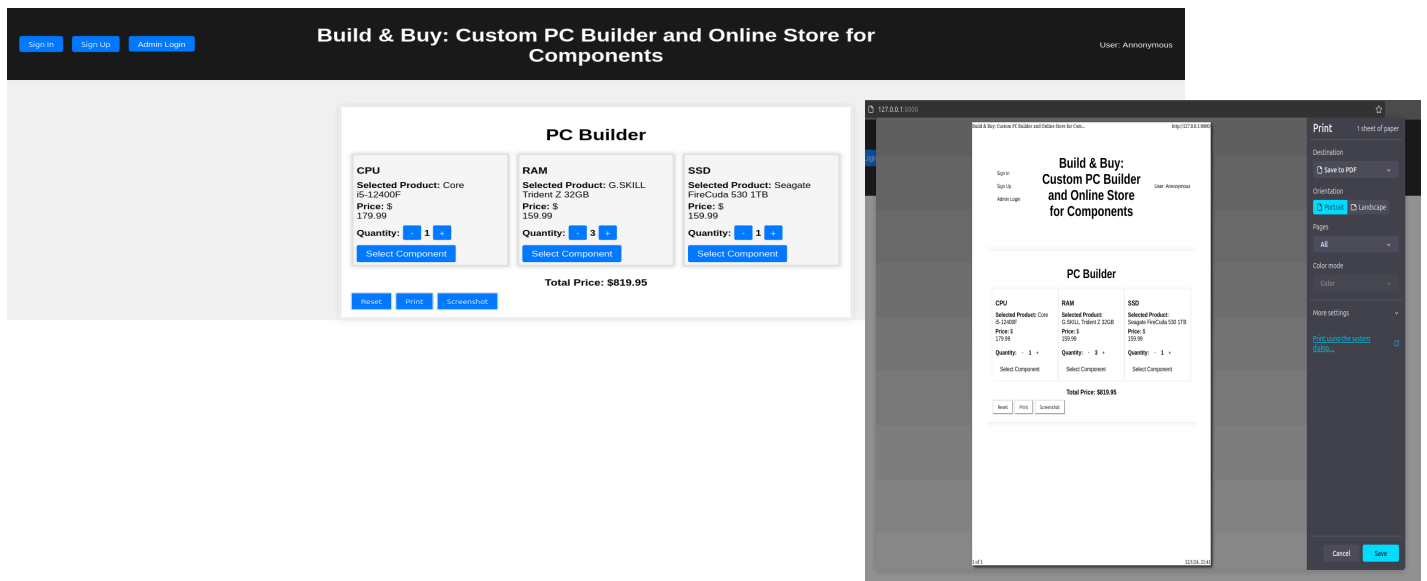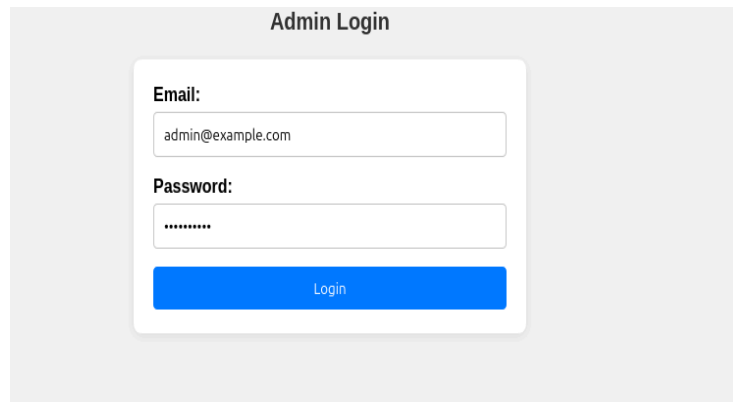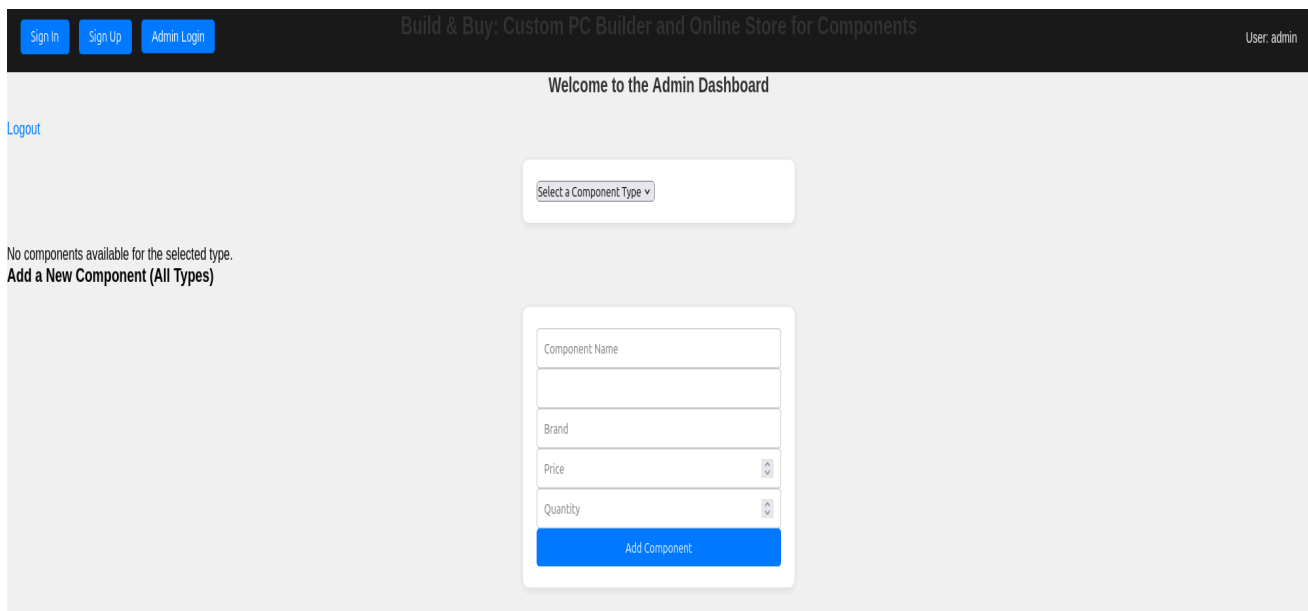
Fig 3.1.4 Taking screenshots and printed copies Selected components with different quantities and total price

3.1.1 Implementation of MODULE 2 (Inventory Management)

Admin Login

Email:

admin@example.com

Password:

••••••••••

Login

Fig 3.1.5 Login to the admin panel for Inventory management

Sign In Sign Up Admin Login   Build & Buy: Custom PC Builder and Online Store for Components   User: admin

Welcome to the Admin Dashboard

Logout

Select a Component Type ∨

No components available for the selected type.
Add a New Component (All Types)

Component Name

Brand

Price

Quantity

Add Component

Fig 3.1.6 Admin Dashboard for inventory management with category-wise inventory management
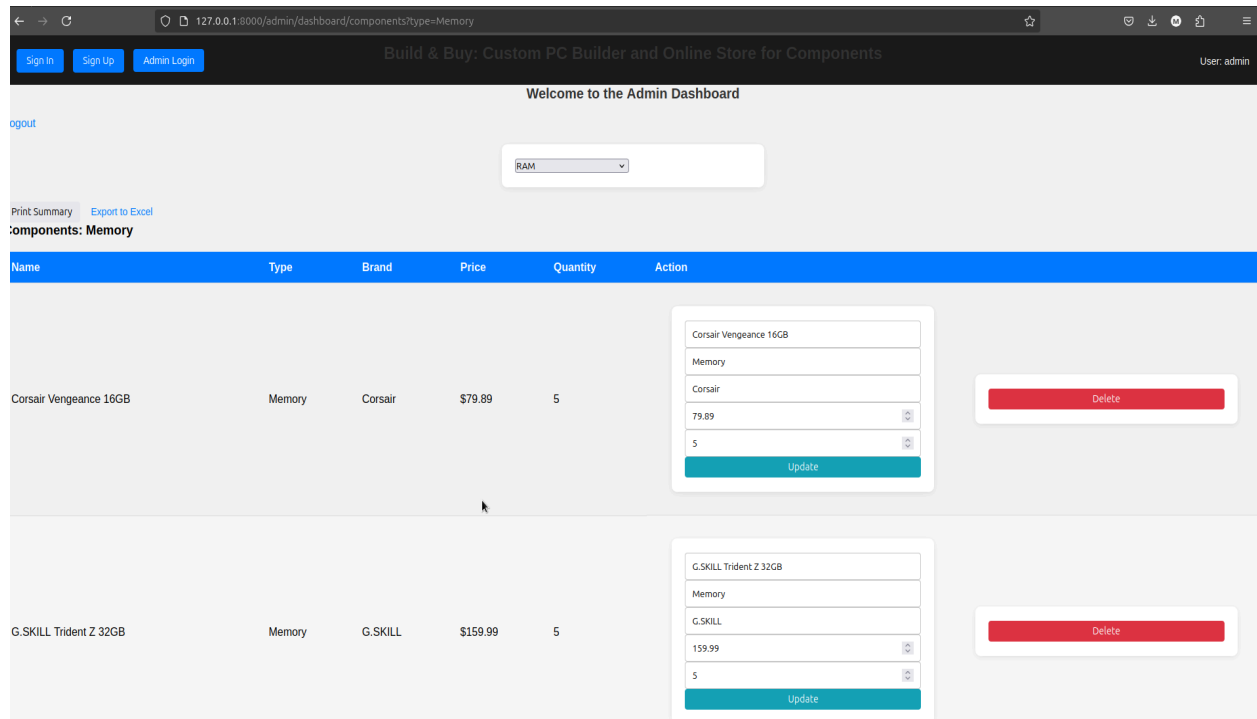
Fig 3.1.7 Admin can update(any column), and delete particular rows with the selected type of component for better category-wise inventory management
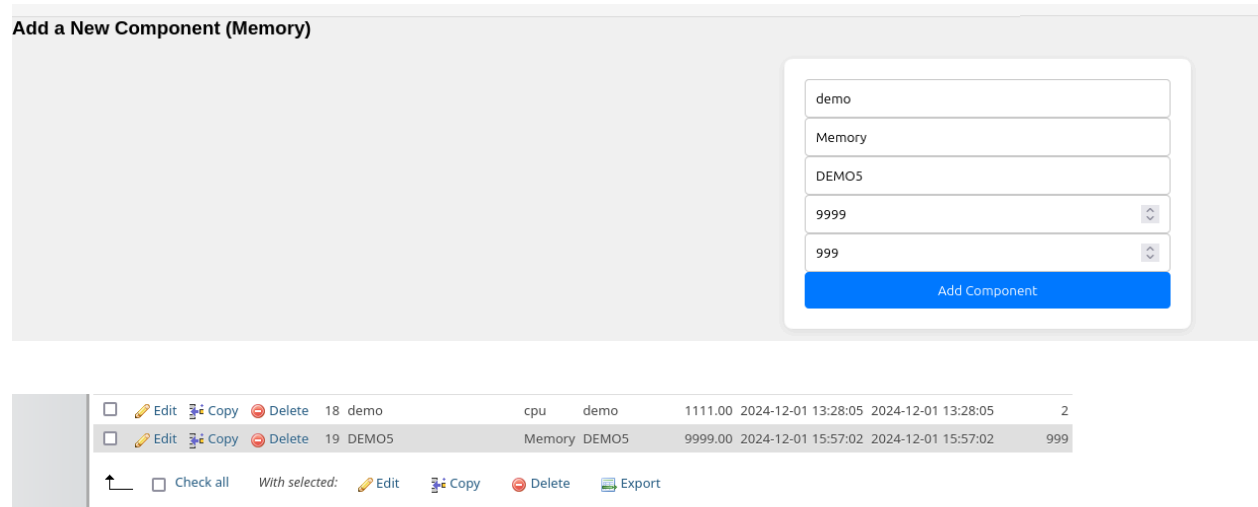

Fig 3.1.8 Admin can add a particular row with the selected type of component for better category management

Fig 3.1.9 Admin for better category management can export the table in Excel and PDF format

3.2 Sprint 3
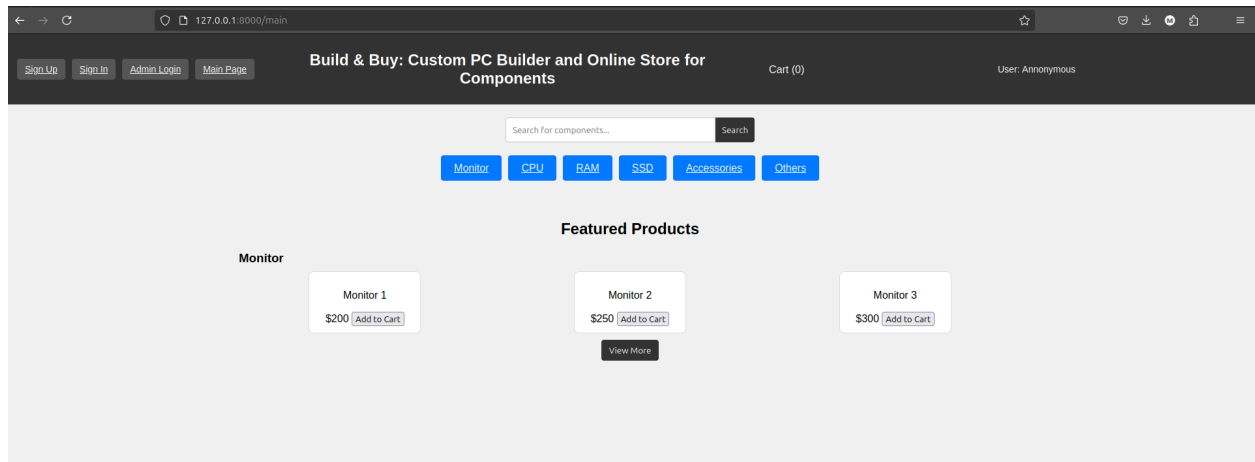## Implementation of MODULE 3 (E-commerce)

Fig 3.2.1 Customer Frontend for ordering different components like featured products(more will be added)
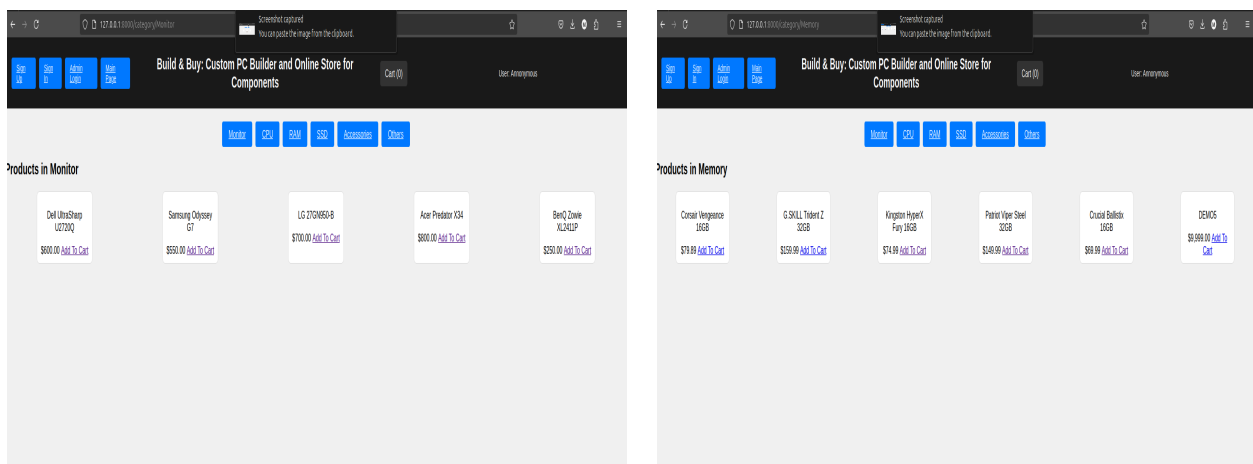

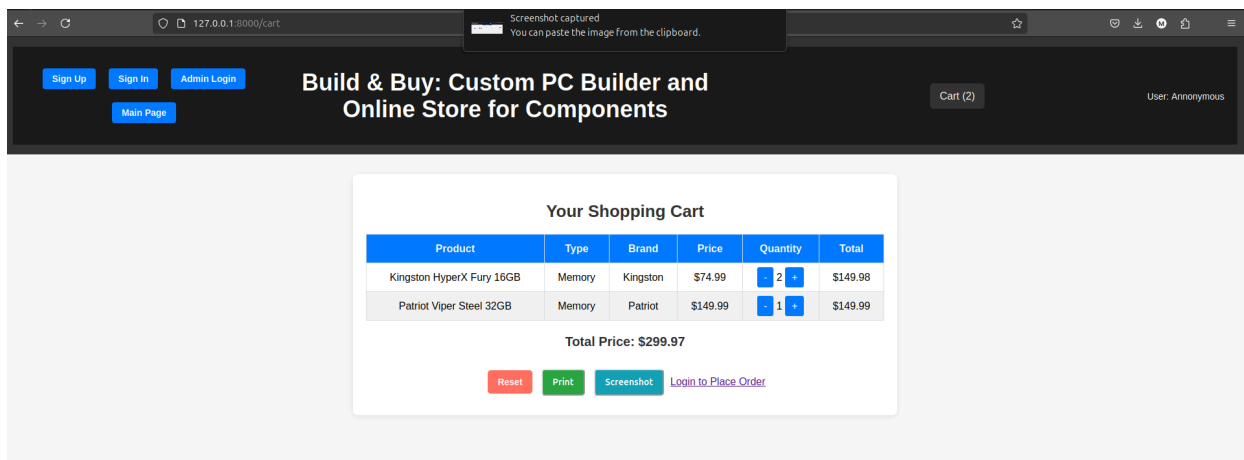Fig 3.2.2 Customer can also choose from type wise components and add to cart

Fig 3.2.3 Customer cart, customer order and make payment(Dummy Payment Portal) after login

Make a Payment - Order #4

Total Price: $299.97

Card Number

411111111111111

PIN

••••

Make Payment

				order_id	user_id	total_price	status	created_at	updated_at	order_details
migrations										
orders										
users										
performance_schema										
phpmyadmin	☐	Edit	Copy	Delete	1	2	3700.00	processing	2024-12-07 14:39:55	2024-12-07 16:57:32	[{"name":"Dell UltraSharp U2720Q","quantity":"2",...
test	☐	Edit	Copy	Delete	3	1	974.99	pending	2024-12-07 17:17:39	2024-12-07 17:17:39	[{"name":"Kingston HyperX Fury 16GB","quantity":1,...
	☐	Edit	Copy	Delete	4	2	299.97	pending	2024-12-07 17:19:31	2024-12-07 17:19:31	[{"name":"Kingston HyperX Fury 16GB","quantity":"2...

Fig 3.2.4 Making payment through dummy payment portal with database

Make a Payment - Order #4

Total Price: $299.97

Card Number

Enter your card number

PIN

Enter your PIN

Make Payment

Payment Successful!

		order_id	user_id	total_price	status	created_at	updated_at	order_details
☐	Edit Copy Delete	1	2	3700.00	processing	2024-12-07 14:39:55	2024-12-07 16:57:32	[{"name":"Dell UltraSharp U2720Q","quantity":"2",...
☐	Edit Copy Delete	3	1	974.99	pending	2024-12-07 17:17:39	2024-12-07 17:17:39	[{"name":"Kingston HyperX Fury 16GB","quantity":1,...
☐	Edit Copy Delete	4	2	299.97	processing	2024-12-07 17:19:31	2024-12-07 17:20:48	[{"name":"Kingston HyperX Fury 16GB","quantity":"2...

Fig 3.2.5 After Making a payment through the dummy payment portal with the database

Fig 3.2.6 Customers dynamic dashboard for seeing orders, orders status, make payment

3.3 Sprint 4
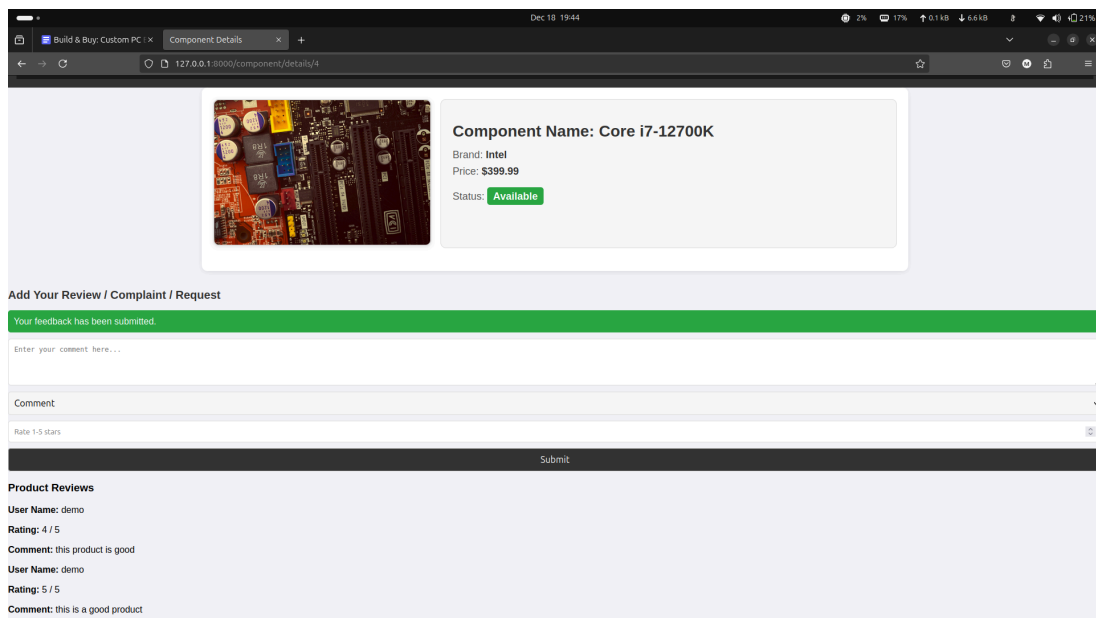## Implementation of MODULE 4 (Customer Add-Ons and Service)

Fig 3.3.1 Customers can review/ complain/ request for product and see others review
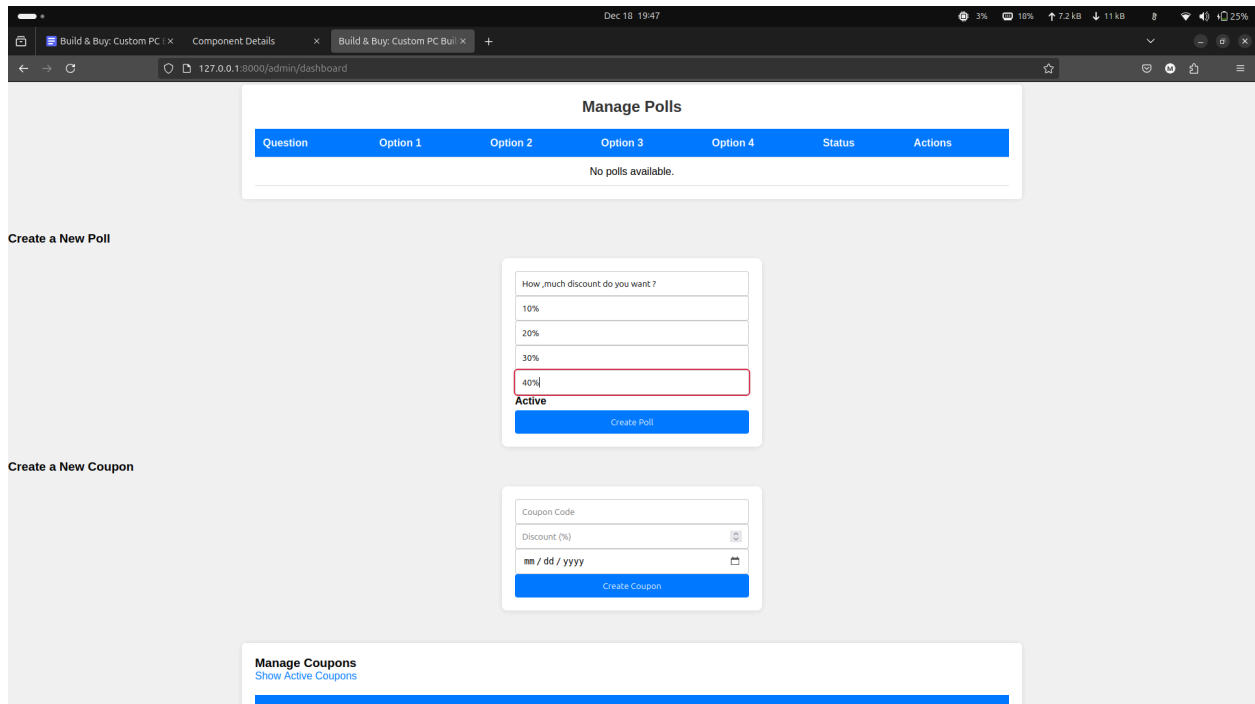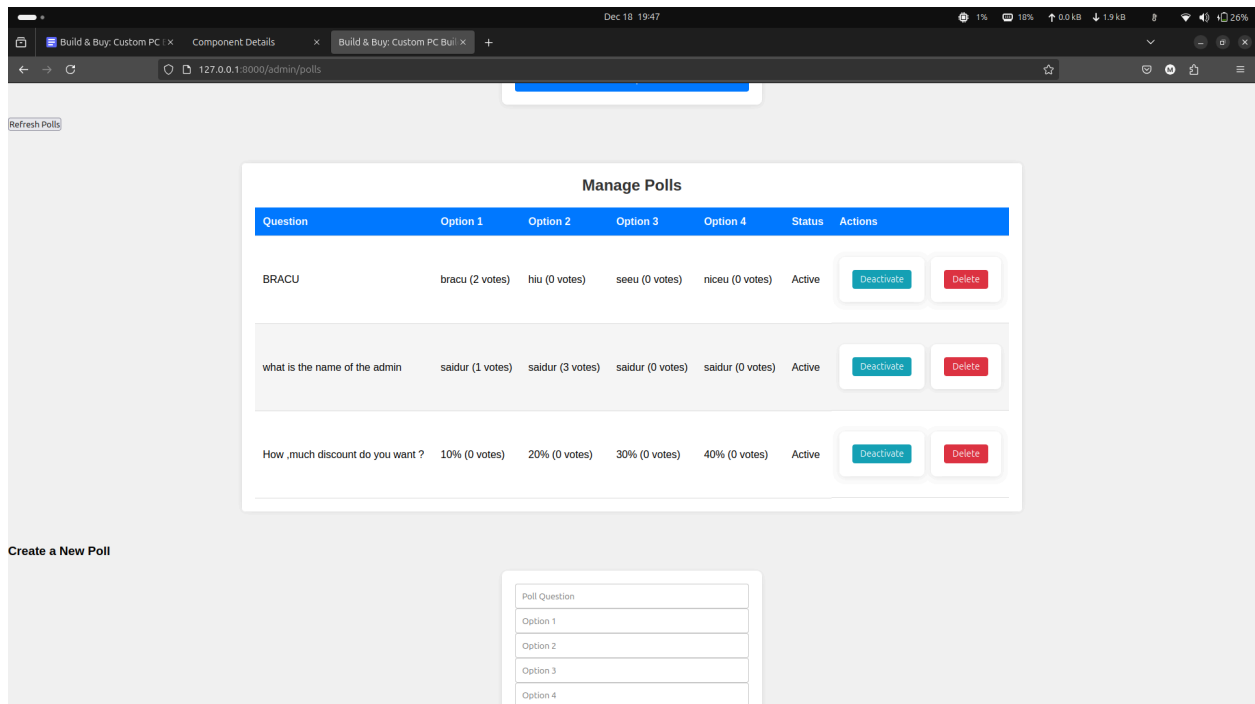
Fig 3.3.2 Admin can create polls

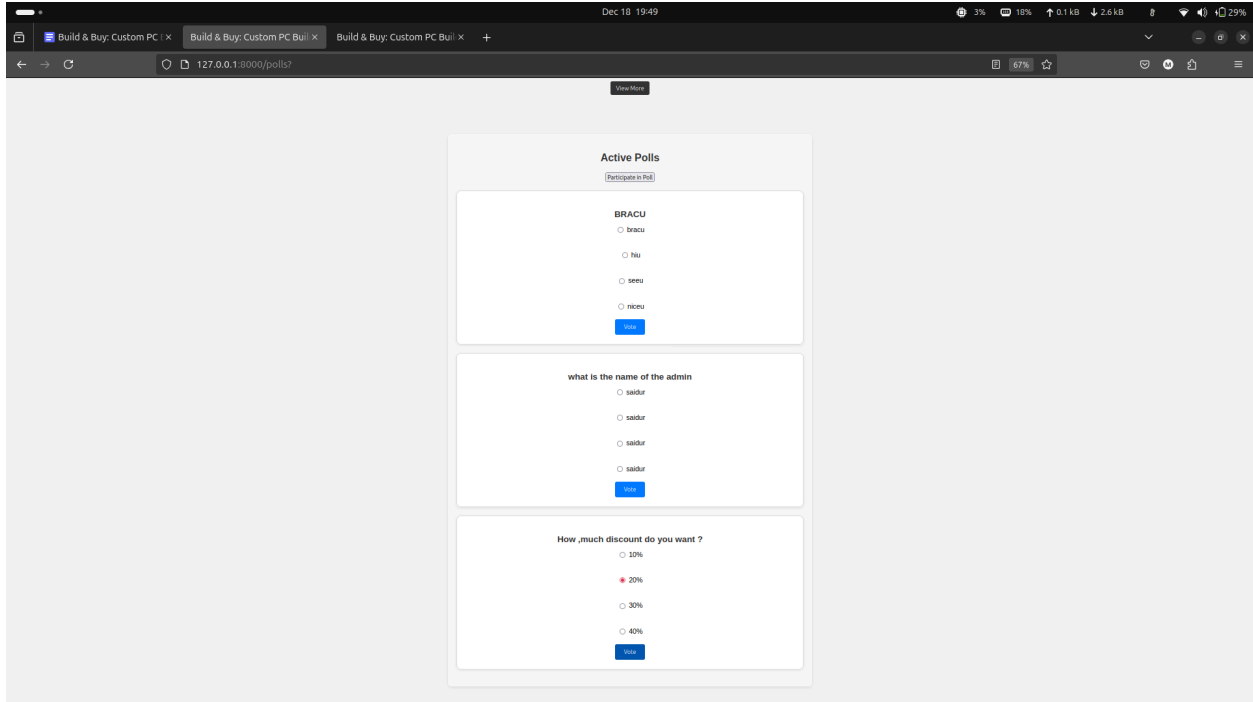Fig 3.3.3 Admin can manage already created polls
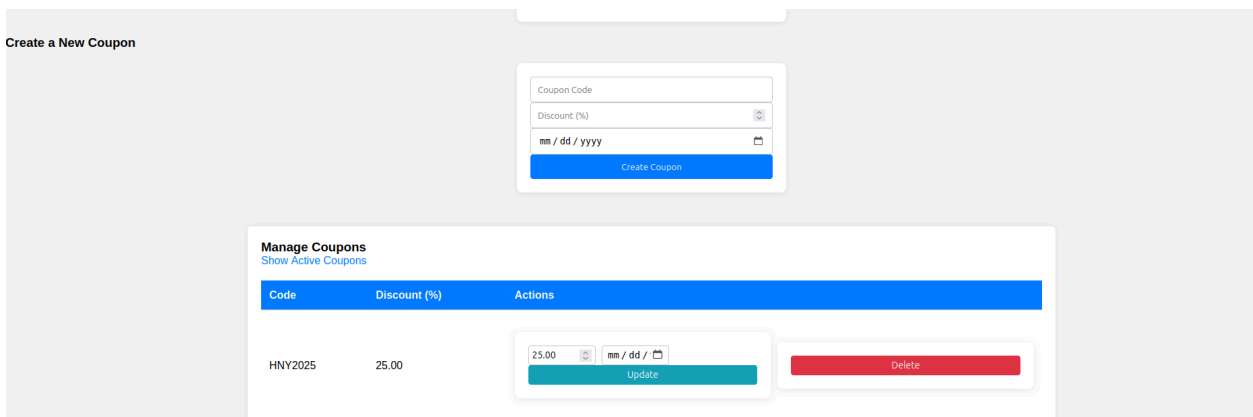
Fig 3.3.4 Logged-in users can participate in polls

Fig 3.3.5 Admin can generate and manage coupons

Fig 3.3.6 Customer before using coupon

Fig 3.3.7 Customer after using coupon

3.4 Sprint 5
Implementation of MODULE 5 (Component Comparison and Performance Check)

Fig 3.4.1 Product Comparison (The same type of product in different companies customer can compare products with prices and power consumption)

Fig 3.4.2-3 Bottleneck Calculator & Recommendation (Customer can predict how their chosen product will perform and get feedback on the components they have chosen)

Build & Buy: Custom PC Builder and Online Store for Components

Cart (0)

User: Annonymous

PC Builder

CPU

Selected Product: Core i5-12400F

Price: $199.00

Power Consumption: watt 50

Quantity: - 1 +

Select Component

RAM

Selected Product: Patriot Viper Steel 32GB

Price: $149.99

Power Consumption: watt 50

Quantity: - 1 +

Select Component

SSD

Selected Product: demo2

Price: $111.00

Power Consumption: watt 50

Quantity: - 1 +

Select Component

Total Power Consumption: 150 watt

Total Price: $459.99

Reset   Print   Screenshot   Bottleneck Calculator

Fig 3.4.4 Total power consumption (Customer can see the total power consumption of their chosen product)