



**GROUP 5**

# **PROJECT SOFTWARE ENGINEER**

**Nhóm O-Fine**

# Group Member

Student ID	Name
22127014	Nguyễn Kim Anh
22127092	Lê Bảo Giang
22127133	Đinh Vũ Huân
22127460	Quách Trần Quán Vinh
22127471	Ngô Hải Bằng





# WEB APPLICATION

## Electronics Defiant Store



01

**Problem Statement**

04

**Analysis and design**

02

**Project Management**

05

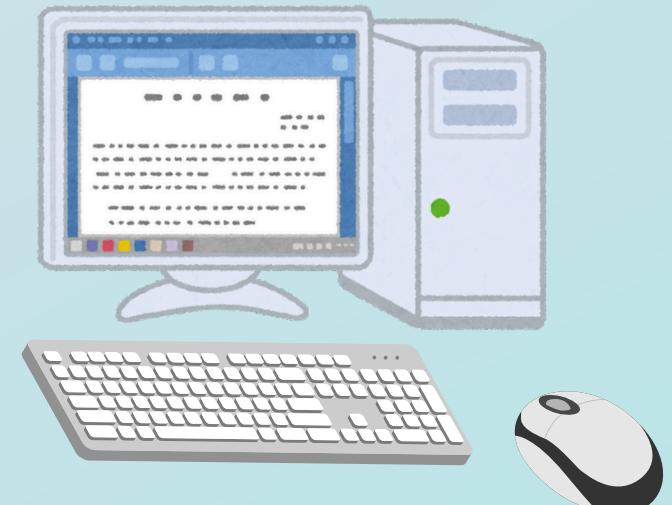
**Testing**

03

**Software Requirements**

06

**Demo**



# Problem Statement

**Objective:** Define the specific challenges in selling electronic products online.

## Welcome To The Store

Have a nice day!

Welcome to our online web store, your one-stop shop for video game purchases.

### Solution 1

Our Platform aims to address these challenges by providing a user-friendly e-commerce site specifically designed for selling electronics.

### Solution 2

Key features include comprehensive product comparison tools, verified reviews

### Solution 3

Strong after-sales support to build customer trust.

# Product Position Statement

For

Tech enthusiasts, gadget lovers, and everyday consumers who want reliable and affordable electronic products.

Who

Need a trustworthy and informative platform to purchase electronics online.

Unlike

Generic e-commerce platforms, our solution focuses on the specific needs of electronics shoppers, providing expert advice.

# Users/Market



## Tech Enthusiasts

Individuals who are always on the lookout for the latest gadgets and electronic innovations.



## Everyday Consumers

People who need reliable electronics for daily use, such as smartphones, laptops, and home appliances.



## Businesses

Small and medium enterprises (SMEs) that require electronics for their operations



# Project Management

Ensures the project stays on schedule and meets its objectives, coordinating across all members.



# Front-End Developers

Design and implement the user interface, ensuring it's intuitive, responsive, and optimized for mobile devices.



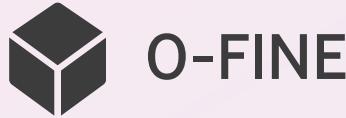
# Back-End Developers

Handle server-side logic, database management, and integration with third-party services like payment gateways and inventory systems.



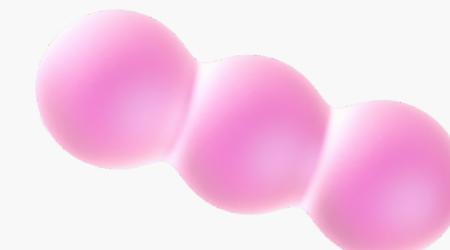
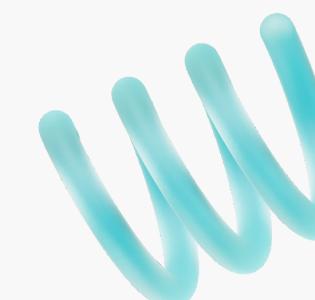
# UX/UI Designers

Create a visually appealing and user-friendly design that caters specifically to the electronics market.



# Tester

Ensure the software meets high-quality standards by checking its functionality and performance against requirements.

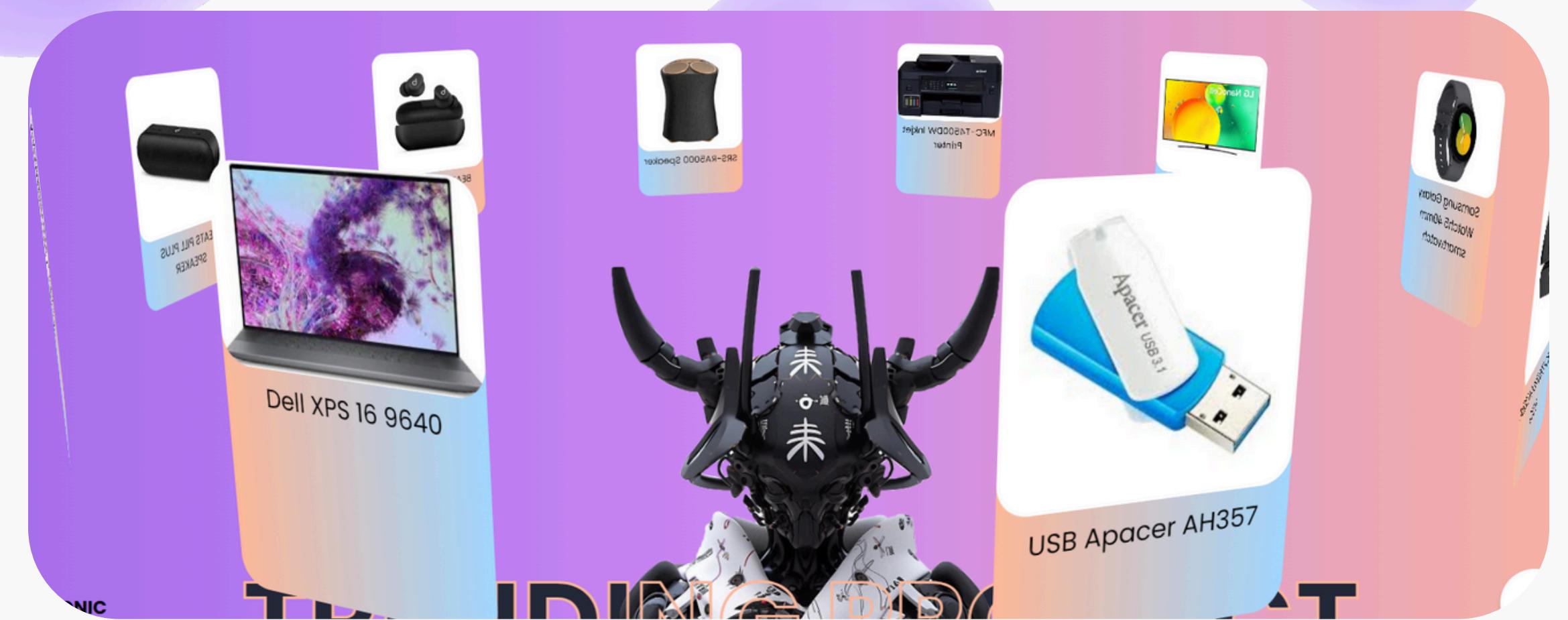


Hải Bằng	Kim Anh	Quán Vinh	Bảo Giang	VŨ HUÂN
<ul style="list-style-type: none"><li>• Project Management</li><li>• Front-End Developer</li><li>• Back-End Developer</li><li>• UI/UX Designer</li></ul>	<ul style="list-style-type: none"><li>• Tester</li><li>• Back-End Developer</li></ul>	<ul style="list-style-type: none"><li>• Back-End Developer</li><li>• Tester</li><li>• Front-End Developer</li></ul>	<ul style="list-style-type: none"><li>• Tester</li><li>• Back-End Developer</li></ul>	<ul style="list-style-type: none"><li>• Tester</li><li>• Back-End Developer</li></ul>



O-FINE

# Software Requirement



## Use-case model

- Customers: Can search, compare products, add to cart, and pay online.
- Administrator: Manage products, orders, and check sales.

## Non-Functional Requirements

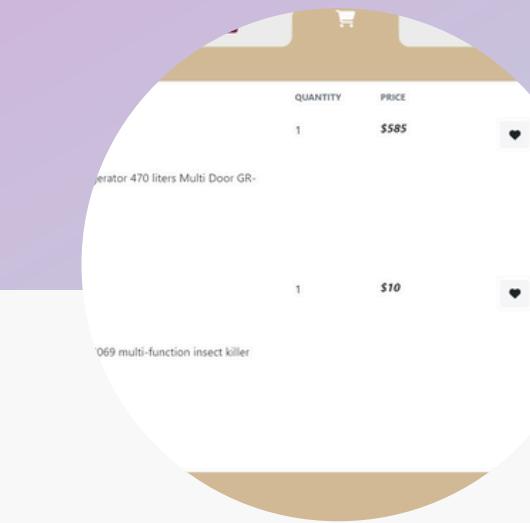
- Security Features: Protect customer information and online transactions.
- High Performance: The system is capable of handling multiple simultaneous visits.
- Scalability: Supports expansion and adding new features in the future.

# Customer Use Cases



## Product Discovery

Users can browse or search for electronic products, filter by category, brand, or specifications.



## Shopping Cart Management

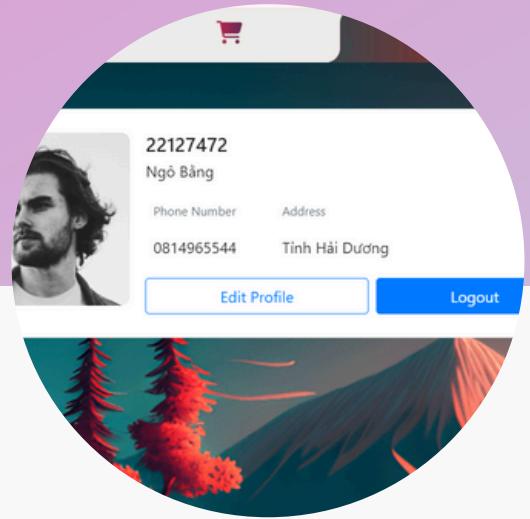
Users can add, remove, and adjust product quantities, with recommendations for complementary products.



## Detailed Product Pages

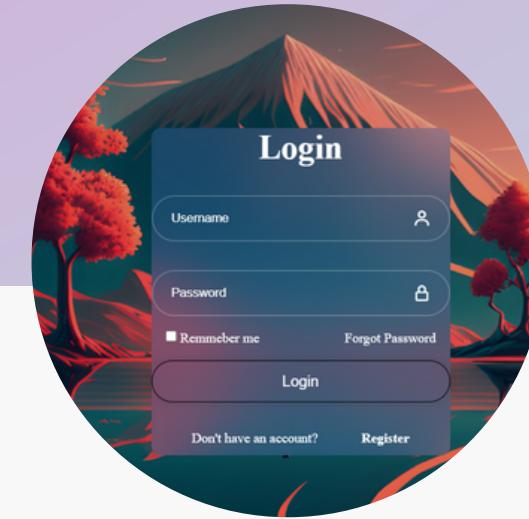
Each product page includes high-quality images, detailed specifications, user manuals, and customer reviews.

# Customer Use Cases



## Profile

The user navigates to their profile page. The user can view and edit personal information (e.g., name, email, address, phone number).



## Login

The user can access the login page. The user enters their email/username and password.



## View other

The user adds items to a wishlist for future consideration. Admins can manage user accounts, products, and orders.

# Non-Functional Requirements

**Usability:** The platform must be accessible to all users, including those with limited technical expertise, with a clear and straightforward user interface.

**Scalability:** The platform should easily accommodate an increasing number of users, products, and transactions.

**Security:** Robust encryption must protect user data and payment information.



O-FINE

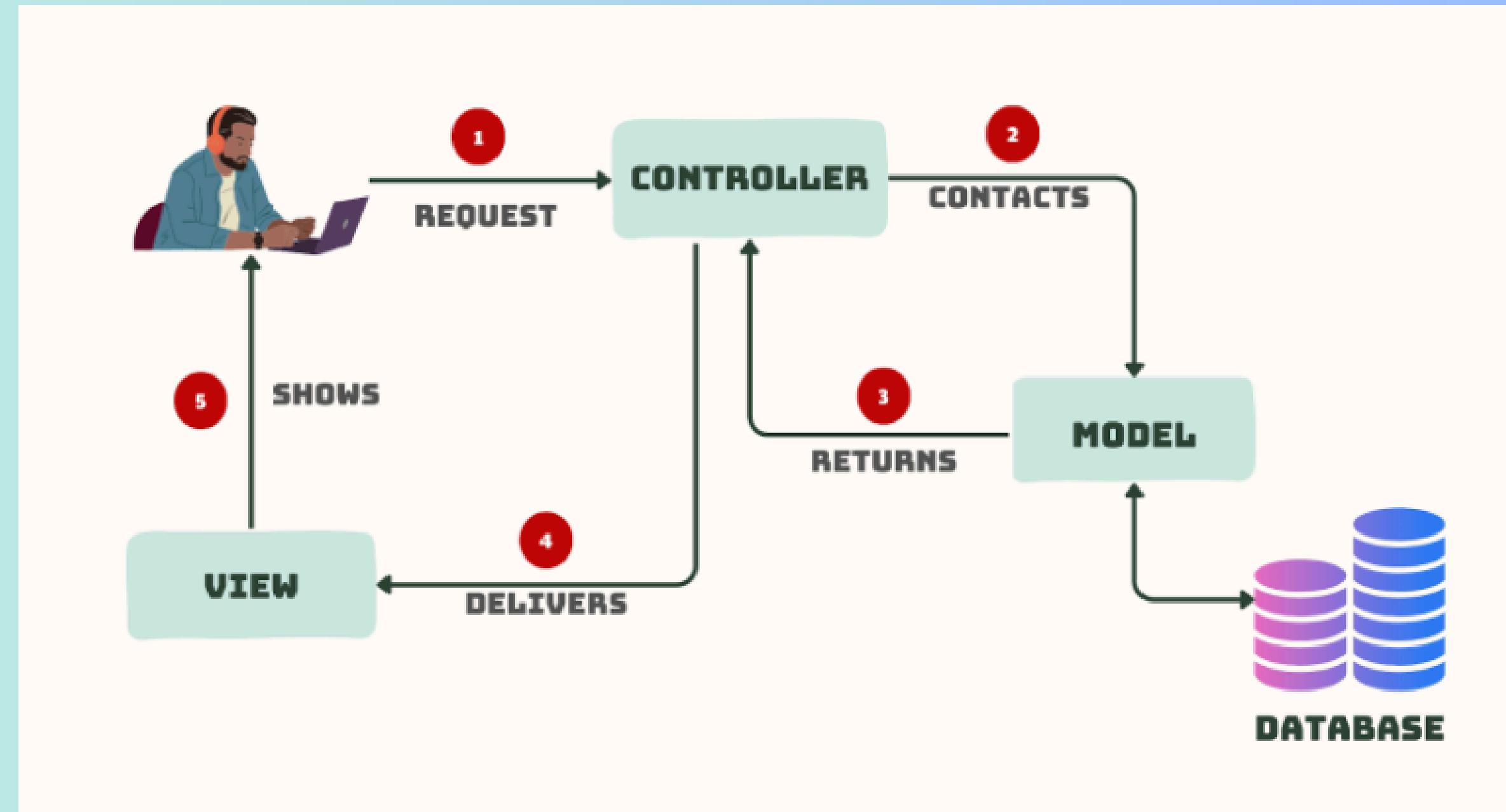
**Analysis &  
Design**

**Architecture**

**Technology**

# Architecture Patterns

## MVC Model



# Technology

## Frontend

### User Interface ( HTML, CSS, and JavaScript )

The UI is the visual part of the application that users see and interact with, including buttons, forms, text, images, and the overall layout.

### User Experience ( Sketch)

UX focuses on how users feel and experience the application, including usability, ease of navigation, and the efficiency of functions.

### APIs and Data Fetching (Tools like 'Axios')

The front end often needs to fetch data from the back end through APIs

# Technology

## Backend

### Server ( Node.js )

The server is the environment where the back end code runs. It processes incoming requests from clients (the front end), executes business logic, and sends responses back to the client.

### Application Logic (Language: JavaScript, Frameworks : Express)

This is the core functionality of the application, where the rules and procedures that define how the application operates are implemented

### Database ( MongoDB )

The database stores, retrieves, and manages all the data that the application needs to function.

# Technology

## Backend

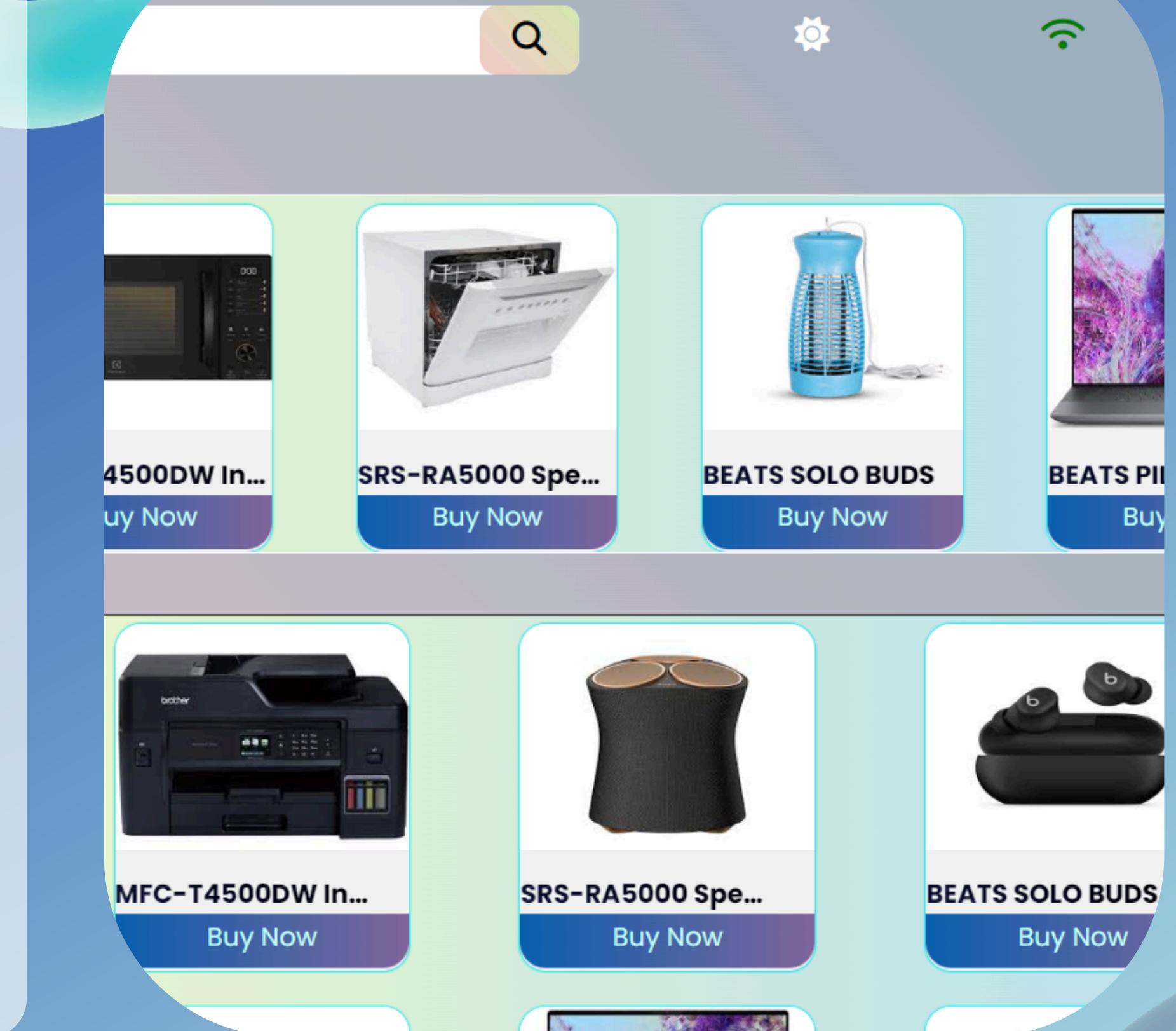
### API ( Express.js (Node.js) )

APIs define how the front end communicates with the back end. They expose endpoints that allow the front end to send requests (like fetching or submitting data) and receive responses.

### Authentication and Authorization ( JWT (JSON Web Tokens))

These mechanisms ensure that users are who they say they are (authentication) and control what resources a user can access (authorization)

# Software Testing



# What is Software Testing?

Software testing is the process of evaluating and verifying that a software application or system performs as expected. It involves executing the software in a controlled environment to identify any defects, errors, or gaps in requirements.

Software testing is crucial for ensuring the quality and reliability of a product. By identifying and fixing issues early, testing helps prevent costly errors, enhances user satisfaction, and ensures that the software meets its specified requirements.

# Software Testing



## Test Environment

A test environment is a setup of software, hardware, network configurations, and other components required to execute test cases



## Test Plan

A detailed document outlining the strategy, scope, objectives, resources, schedule, and activities to be carried out during the software testing process



## Test Results

Test results are the outcomes of executing test cases in a test environment

# Functional Testing

Objective: Ensure all features of the website work as expected.

Example: Verify that the product search function returns accurate results when users search for specific electronics items.

## Add To Cart

Objective: Ensure users can add products to their cart.

Example: Verify that when a user clicks “Add to Cart” for a product, the item is correctly added to the cart and the cart icon updates with the correct item count.

## Product Listing

Objective: Ensure products are displayed correctly on the store page.

Example: Verify that all products are listed with accurate details such as name, price, image, and availability. Check that filters and sorting options (e.g., by price, popularity) work correctly.

## Wishlist Management

Objective: Allow users to add and manage products in their wishlist.

Example: Verify that users can add products to their wishlist and that these items are saved across sessions. Test the functionality to remove items from the wishlist and ensure the list updates correctly.

THE

END