“Electronica: E-Commerce Platform for Electronics”

A Major Project Report

Submitted in partial fulfilment of the requirements for the

Award of degree of Bachelor of Computer Applications

2019-2022

Submitted by:

Saransh Chawla

0191BCA046

Guided by:

Mr Ajay Kumar

BHARATI VIDYAPEETH

(DEEMED TO BE UNIVERSITY)

INSTITUTE OF MANAGEMENT & RESEARCH, NEW DELHI

A-4, Paschim Vihar, New Rohtak Road, New Delhi-110033

2022

STUDENT UNDERTAKING

This is to certify that the project titled **Electronica: E-Commerce Platform for Electronics** submitted to **Bharati Vidyapeeth (Deemed to be University), Pune** in partial fulfilment of the requirement for the award of the degree of **Bachelor of Computer Applications** is an original work carried out under the guidance of **Mr Ajay Kumar.** The matter embodied in this project is a genuine work done by me and has been submitted neither to this University nor to any other University for the fulfilment of the requirement of the course of study.

Signature of student

Name of the student : Saransh Chawla

ERPID : 0191BCA046

PRN No : 1920100582

CERTIFICATE FROM INTERNAL GUIDE

This is to certify that the Project titled **Electronica: E-Commerce Platform for Electronics** is an academic work done by **Saransh Chawla** submitted in the partial fulfillment of the requirement for the award of the Degree of **Bachelor of Computer Applications** from **Bharati Vidyapeeth (Deemed to be University), Pune** under my guidance.

To the best of my knowledge and belief the data & information presented by him/her in the project has not been submitted earlier whether to this University or to any other University / Institute for the fulfillment of the requirement of any course of study.

Name of Internal Guide: Mr. Ajay Kumar

CERTIFICATE FROM THE DIRECTOR

This is to certify that the Project titled "**Electronica: E-Commerce Platform for Electronics**" is an academic work done by **Saransh Chawla** submitted in partial fulfillment of the requirement for the award of the Degree of Bachelor of Computer Applications from **Bharati Vidyapeeth (Deemed to be University), Pune.**

The authenticity of the project work will be examined by the viva examiner which includes data verification, checking duplicity of information etc. and it may be rejected due to non-fulfillment of quality standards set by the institute.

Dr. Yamini Aggarwal

Director-Incharge

PREFACE

The term “e-commerce” simply means the sale of goods or services on the internet. In its most basic form, e-commerce involves electronically transferring funds and data between 2 or more parties. This form of business has evolved quite a bit since its beginnings in the electronic data interchange of the 1960s and the inception of online shopping in the 1990s.

Electronica allows people to buy and sell physical Electronics goods, services, and digital products over the internet rather than at a brick-and-mortar location. Through our website, a business can process orders, accept payments, manage shipping and logistics.

Varous modules included in our website is:

1. Full featured shopping cart
2. Product reviews and ratings
3. Top products carousel
4. Product pagination
5. Product search feature
6. User profile with orders
7. Admin product management
8. Admin user management
9. Admin Order details page
10. Mark orders as delivered option
11. Checkout process (shipping, payment method, etc)
12. Stripe payment integration
13. Database seeder (products & users)

ACKNOWLEDGEMENT

I am using this opportunity to express my gratitude to everyone who supported me throughout the course of this project. I am thankful for their aspiring guidance, invaluably constructive criticism and friendly advice during the project work.

I am sincerely grateful to them for sharing their truthful and illuminating views on a number of issues related to the project.

I express my warm thanks to our director **Dr. Yamini Aggarwal** for all these resources provided by her and such a wonderful to learn and enhance our knowledge.

I would also like to give a special thanks to my project guide **Mr. Ajay Kumar** for his/her support and precious guidance at each step throughout the project.

INDEX

CHAPTER 1	**INTRODUCTION**	**1-7**
1.1	Introduction about project	2
1.2	Present state of the existing System	3
1.3	Need of Computerization of System	4
1.4	Problem of the existing System	5
1.5	Proposed Software	6
1.6	Importance of the Project	7
CHAPTER 2	**SYSTEM ANALYSIS**	**8-12**
2.1	Analysis methodology / gathering techniques	9
2.2	Feasibility Study	10-11
2.3	Choice of Platforms 2.3.1 Software Used 2.3.2 Hardware Used	12
CHAPTER 3	**SYSTEM DESIGN**	**13-20**
3.1	Process Model used	14-15
3.2	Entity Relationship Diagram (ERD)	16
3.3	Functional Design (DFD)	17-18
3.4	Interface Design	19
3.5	Output Design	20
CHAPTER 4	**TESTING AND IMPLEMENTATION**	**21-30**
4.1	Testing Methodology (Types)	22-23
4.2	Testing methodology applied (why)	24
4.3	Test Cases	25-26
4.4	Characterstics of Good test cases	27
4.5	Gap Analysis / Rework	28-29
4.6	Hardware and software Requirement	30
CHAPTER 5	**CONCLUSION AND REFERENCES**	**31-35**
5.1	Conclusion	32
5.2	Limitation of the System	33
5.3	Future Scope for Modification	34
5.4	References / Bibliography	35
CHAPTER 6	**ANNEXURES**	**36-131**
A-1	Menu Flow Diagram	37
A-2	Sample Input	38-41
A-3	Sample Output	42-43
A-4	Program Code	44-145
A-5	Mentee Feedback	146

CHAPTER 1	INTRODUCTION	1-7
1.1	Introduction about project	2
1.2	Present state of the existing System	3
1.3	Need of Computerization of System	4
1.4	Problem of the existing System	5
1.5	Proposed Software	6
1.6	Importance of the Project	7

1.1 INTRODUCTION TO THE PROJECT

E-commerce is purchasing and selling activities happening over the Internet. The different types of commerce include B2B, B2C, C2C (consumer to consumer) and C2B (consumer to business). An e-commerce platform is the software application where both parties, the seller and the consumer, come and play their role. Essentially, a consumer should be able to use an e-commerce platform to discover products, shop around using a basket and then check out.

Going by the above definition of an E-Commerce Platform we can say that the aim of our project is to create a scalable E-Commerce Solution where a seller can manage and sell his inventory of electronics.

The webapp features a fully functioning cart, checkout steps, payment integration with Stripe gateway and an admin panel for adding, deleting and updating users and products. Making it one of the most comprehensive solutions available to the buyers as well as sellers.

1.2 PRESENT STATE OF THE EXISTING SOFTWARE

Though e-commerce offers many advantages to customers, business, society and nation, there are still some areas of concern that need to be addressed. The following are some of the limitations or disadvantages of e-commerce.

a. **Security:** The biggest drawback of e-commerce is the issue of security. People fear to provide personal and financial information, even though several improvements have been made in relation to data encryption. Certain websites do not have capabilities to conduct authentic transactions. Fear of providing credit card information and risk of identity limit the growth of e-commerce.

b. **Lack of privacy:** Many websites do not have high encryption for secure online transaction or to protect online identity. Some websites illegally collect statistics on consumers without their permission. Lack of privacy discourages people to use internet for conducting commercial transactions.

c. **Tax issue:** Sales tax is another bigger issue when the buyer and seller are situated in different locations. Computation of sales tax poses problems when the buyer and seller are in different states. Another factor is that physical stores will lose business if web purchases are free from tax.

d. **Technical limitations:** Some protocol is not standardized around the world. Certain software used by vendor to show electronic images may not be a common one. It may not be possible to browse through a particular page due to lack of standardized software. Insufficient telecommunication bandwidth may also pose technical problems.

1.3 NEED OF IMPROVED SYSTEM

1. **Site navigation**: is everything. If users can find what they want in the shortest time possible, they are more likely to buy something. They are also more likely to return to your site when they want something else. Our Website features easy to navigate pages and breadcrumbs during checkout, so there is no confusion to the user.

2. **Site speed:** Since most websites are built once and not updated or maintained, the latency increases and speed decreases meaning the ultimate shopping experience falls for the end user. Our website is scalable amd maintainable in every possible way implying that the future scope of the website can be increased easily.

3. **Checkout process:** This needs to be as simple as possible. If your checkout process involves multiple steps, simply let your customer know. This will decrease the chances of them leaving your site frustrated before purchasing. Our website has breadcrumbs added in the checkout process which tell the user which information to enter on a particular screen.

4. **Seeking Reviews:** is one of the most important aspect of an E-Commerce Platform. Reviews help users make informed decisions. Therefre our website boasts a full functioning star based review system which highlights a product's likability on the home and single product page.

1.4 PROBLEM OF EXISTING SYSTEM

After the Spanish flu, France followed Britain into industrial revolution. Similarly after Covid, there is a structural shift in the world economy and businesses are transitioning to online solutions for better reach, access and sales. Even though an impressive number of business have transitioned online, a lot of them are still left to break the barrier of going online. While a traditional mode of business is not bad, there are several limitations that make E-Commerce a lucrative option.

a. **Overcome Geographical Limitations**: If you have a physical store, you are limited by the geographical area that you can service. With an e-commerce website, the whole world is your playground. Additionally, the advent of m-commerce, i.e., e-commerce on mobile devices, has dissolved every remaining limitation of geography.

b. **Gain New Customers with Search Engine Visibility:** Physical retail is driven by branding and relationships. In addition to these two drivers, online retail is also driven by traffic from search engines. It is not unusual for customers to follow a link in search engine results and land on an e-commerce website that they have never heard of. This additional source of traffic can be the tipping point for some e-commerce businesses.

c. **Locate the Product Quicker:** It is no longer about pushing a shopping cart to the correct aisle or scouting for the desired product. On an e-commerce website, customers can click through intuitive navigation or use a search box to narrow down their product search immediately. Some websites remember customer preferences and shopping lists to facilitate repeat purchase.

d. **Provide Abundant Information:** There are limitations to the amount of information that can be displayed in a physical store. It is difficult to equip employees to respond to customers who require information across product lines. E-commerce websites can make additional information easily available to customers. Most of this information is provided by vendors and does not cost anything to create or maintain.

1.5 PROPOSED SOFTWARE

1. The proposed website has multiple features like password encryption through package **bcrypt.js** and login token facility through **Json Web Token package** to validate and persist the login user for a set period of time. Above mentioned packages along with an **encrypted MongoDB cluster** should address some of the most commen security concerns related to a website.

2. To prvent man-in-the-middle attacks our website is hosted on **heroku (free cloud hosting provider)** which uses reverse proxy to force the **proto-header (internet information packages)** of all connections to the website to be redirected via HTTPS, removing any threats to privacy.

3. Taxes can be a headache for any business. Therefore our website uses a relatively new but famous payment gatewat called **stripe,** which handles all the taxes, shipping charges and bank integration automatically for the business owner, so that he can concentrate on more pressing matters.

4. There is nothing worse than a potential customer visiting your website and him not being able to access it properly. Our website is built using Javascript frameworks namely **React.js and Node.js**. These two technologies are forward and backward compatible with most browsers, even if those browsers might be running on a low end machine, giving business owner the peace of mind that their website is always ready to server customers.

5. Lastly, our website uses a component library called the **React-Bootstrap** which ensures responsive design meaning that our website looks good on all kinds of screens, whether big or small.

1.6 IMPORTANCE OF THE SOFTWARE

The importance of the “Electronica” is that it:

• stores computerised data, therefore future retrieval of data becomes easy and fast.

• minimizes the chance of errors.

• Keeps the prices standard, therefore a fine quality treatment is often possible.

• reduces malpractices within the organization.

• doesn’t require file management.

CHAPTER 2	**SYSTEM ANALYSIS**	**8-12**
2.1	Analysis methodology / gathering techniques	9
2.2	Feasibility Study	10-11
2.3	Choice of Platform 2.3.1 Software Used 2.3.2 Hardware Used	12

2.1 ANALYSIS METHODOLOGY /REQUIREMENT GATHERING TECHNIQUES

Requirements gathering, also known as requirements elicitation, is the practice of defining software requirements. This process involves a variety of activities, such as acquiring business requirements from relevant stakeholders to better understand user needs, codifying the data in the form of user stories and ensuring that all parties are on the same page regarding the project being built. Requirements gathering is often overlooked, especially when timelines are tight and budgets are stretched thin. However, disregarding the requirements gathering process can jeopardize expectations and leave room for error. Thus, understanding how to gather requirements for a software project is critical to success for both the vendor and the organization receiving the product.

Some requirement gathering techniques that we used in our software are:

One on one interviews

Interviews are strong medium to collect requirements. Organization may conduct several types of interviews such as:

- Structured (closed) interviews, where every single information to gather is decided in advance, they follow pattern and matter of discussion firmly.
- Non-structured (open) interviews, where information to gather is not decided in advance, more flexible and less biased.
- Oral interviews
- Written interviews
- One-to-one interviews which are held between two persons across the table.
- Group interviews which are held between groups of participants. They help to uncover any missing requirement as numerous people are involved.

Surveys

Organization may conduct surveys among various stakeholders by querying about their expectation and requirements from the upcoming system

Questionnaires

A document with pre-defined set of objective questions and respective options is handed over to all stakeholders to answer, which are collected and compiled.

A shortcoming of this technique is, if an option for some issue is not mentioned in the questionnaire, the issue might be left unattended.

Task analysis

Team of engineers and developers may analyze the operation for which the new system is required. If the client already has some software to perform certain operation, it is studied and requirements of proposed system are collected.

Domain Analysis

Every software falls into some domain category. The expert people in the domain can be a great help to analyze general and specific requirements.

Brainstorming

An informal debate is held among various stakeholders and all their inputs are recorded for further requirements analysis.

Prototyping

Prototyping is building user interface without adding detail functionality for user to interpret the features of intended software product. It helps giving better idea of requirements. If there is no software installed at client's end for developer's reference and the client is not aware of its own requirements, the developer creates a prototype based on initially mentioned requirements. The prototype is shown to the client and the feedback is noted. The client feedback serves as an input for requirement gathering.

2.2 FEASIBILITY STUDY

Feasibility Study in Software Engineering is a study to evaluate feasibility of proposed project or system. Feasibility study is one of stage among important four stages of Software Project Management Process. As name suggests feasibility study is the feasibility analysis or it is a measure of the software product in terms of how much beneficial product development will be for the organization in a practical point of view. Feasibility study is carried out based on many purposes to analyze whether software product will be right in terms of development, implantation, contribution of project to the organization etc. Various types of feasibility study:

Economic feasibility:

In Economic Feasibility study cost and benefit of the project is analyzed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analyzed whether project will be beneficial in terms of finance for organization or not.

Technical feasibility:

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team.

Legal feasibility:

In Legal Feasibility study project is analyzed in legality point of view. This includes analyzing barriers of legal implementation of project, data protection acts or social media laws, project certificate, license, copyright etc.

Operational feasibility:

In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment. Along with this other operational scopes are determining usability of product, Determining suggested solution by software development team is acceptable or not etc.

2.3 CHOICE OF PLATFORM

2.3.1 Hardware used

- Processor : Intel® Core™ i7-10750H CPU @ 2.60GHz GHz 2.50 GHz (10^{th} Gen)
- Clock speed : 1000MHz
- System bus : 64 bits
- RAM : 16 GB of RAM
- SSD/HDD : 256GB/1 TB
- System Type : ACER LAPTOP
- System model : ACER PREDATOR HELIOS 300

2.3.2 Tech Stack

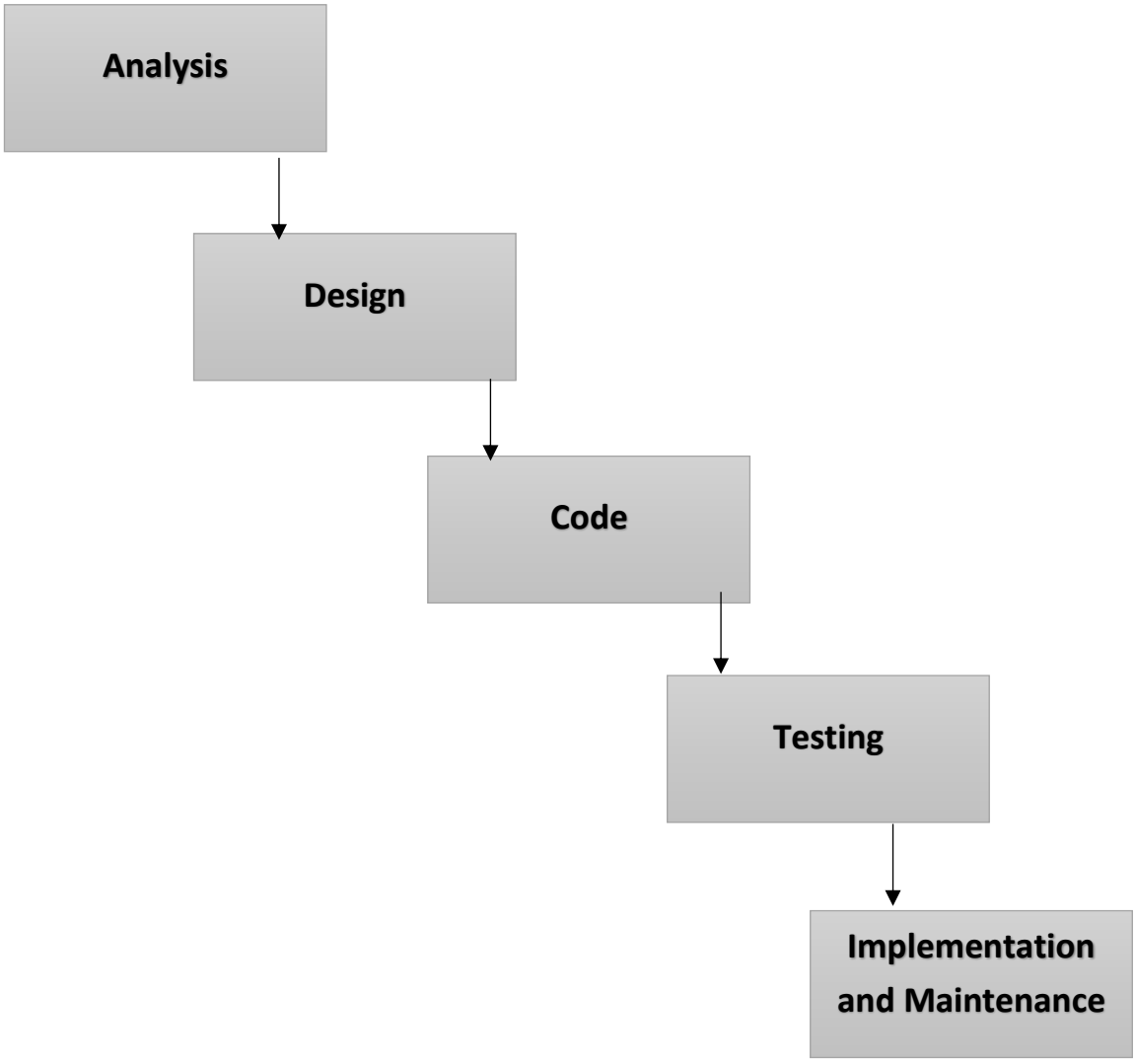
- Front-End : React.js
- Back-End : Node.js + Express.js
- Database : MongoDB (NoSQL Database)

CHAPTER 3	**SYSTEM DESIGN**	**13-20**
3.1	Process Model used	14-15
3.2	Entity Relationship Diagram	16
3.3	Functional Design (DFD)	17-18
3.4	Interface Design	19
3.5	Output Design	20

3.1 PROCESS MODEL USED

Our web-app is developed using the Waterfall Model primarily known for its clearly defined stages and well understood milestones. The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete.

Structure of Waterfall Model diagram:



- **Requirements Gathering**: All possible requirements of the system to be developed are captured in this phase and based on that "Software Requirement Specification Document" is created.

- **Design**: Requirement specification from first phase are studied in this phase and system design is prepared. This system design helps in defining the overall system architecture

- **Coding:** With the inputs from system design, system is first developed in small programs called units. Each unit is tested for its functionality.

- **Integration &Testing:** All the units developed in the coding phase are integrated into a system. Post integration system is checked for any faults.

- **Implementation:** In this phase the system is deployed in customer environment or released into the market.

- **Maintenance:** If some issue comes up in the client environment, patches are released for the issue.

Every software developed is different and requires a suitable SDLC to be followed based on the internal and external factors.

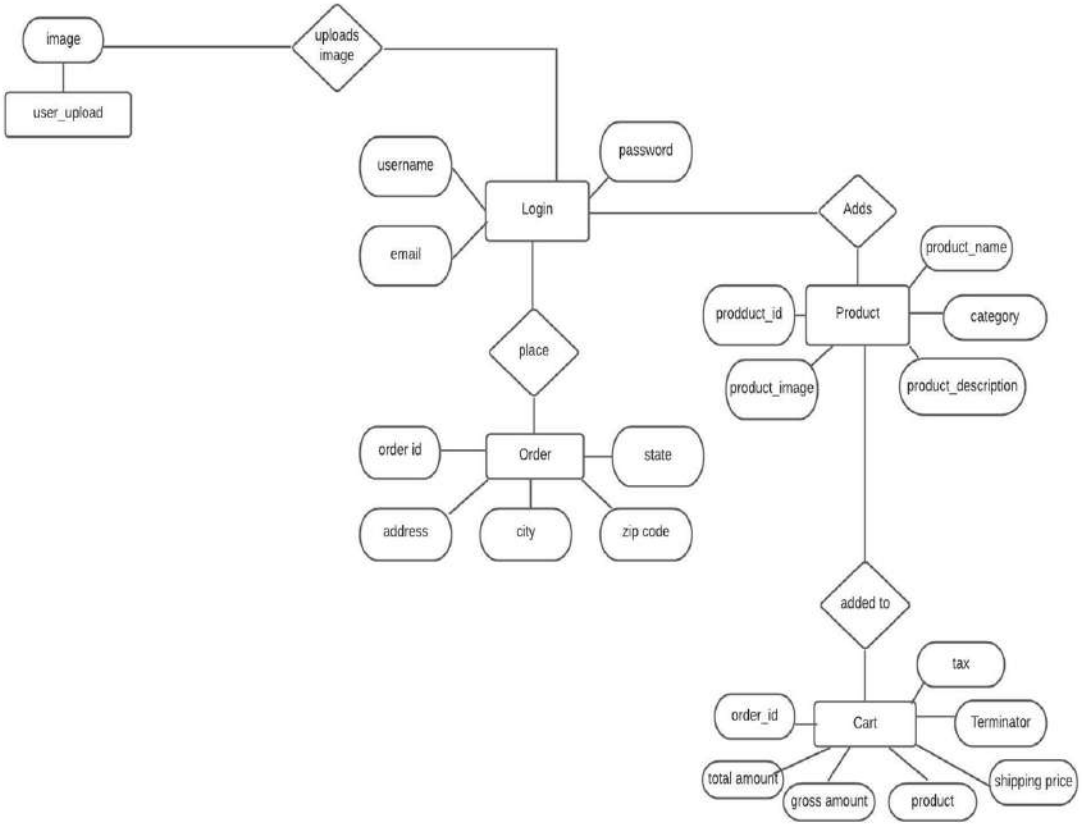
Situations where waterfall model is appropriate:

- Requirements are very well documented, clear and fixed.
- Product definition is stable.
- Technology is understood and is not dynamic.
- There are no ambiguous requirements.
- Ample resources with required expertise are available to support the product.
- The project is short.

Advantages of Waterfall Model

- Simple and easy to understand and use
- Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.
- Phases are processed and completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Clearly defined stages.

3.2 ENTITY RELATION DIAGRAM



3.3 FUNCTIONAL DESIGN (DFD)

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. They can be used to analyze an existing system or model a new one.

DFD components are:

1. Entity
2. Process
3. Data Flow
4. Output

DFD's mentioned in **Electronica** are

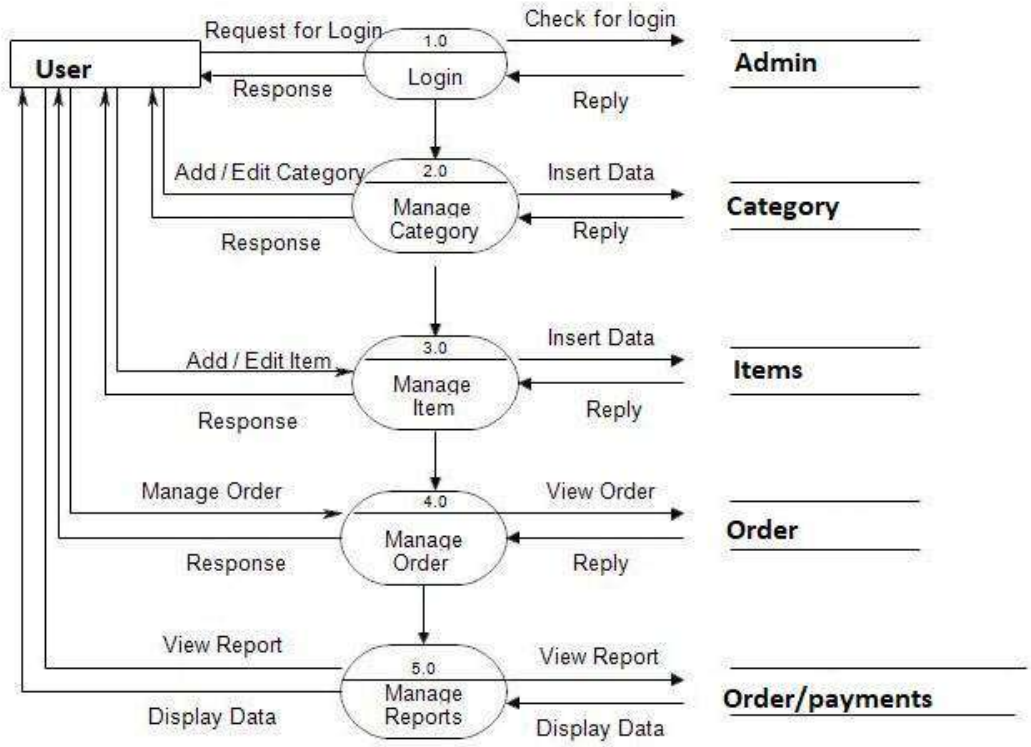
- **0 Level DFD**
- **1 Level DFD**

Level 0 DFD

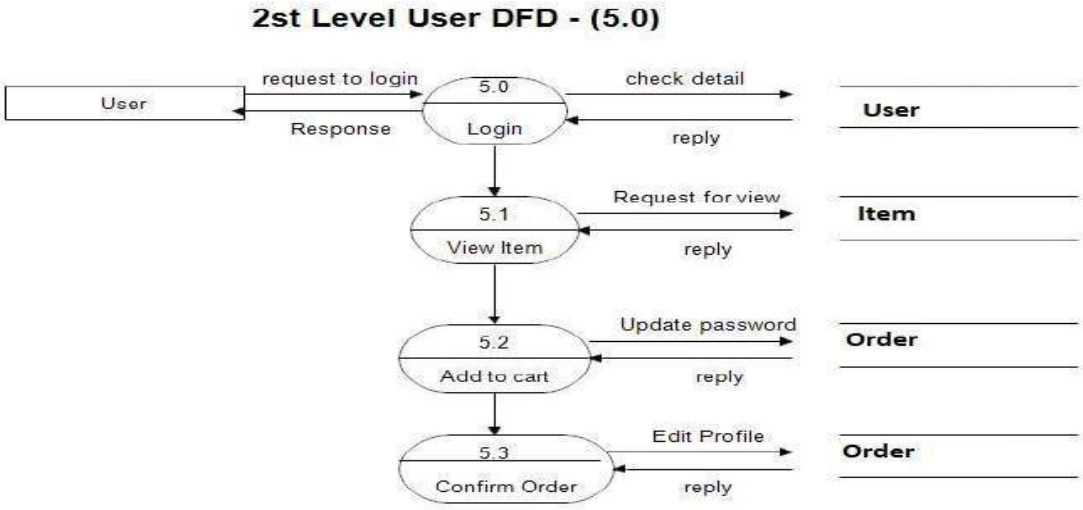
DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled. It's designed to be an at-a-glance view, showing the system as a single high-level process, with its relationship to external entities. It should be easily understood by a wide audience, including stakeholders, business analysts, data analysts and developers.

Level 1 DFD

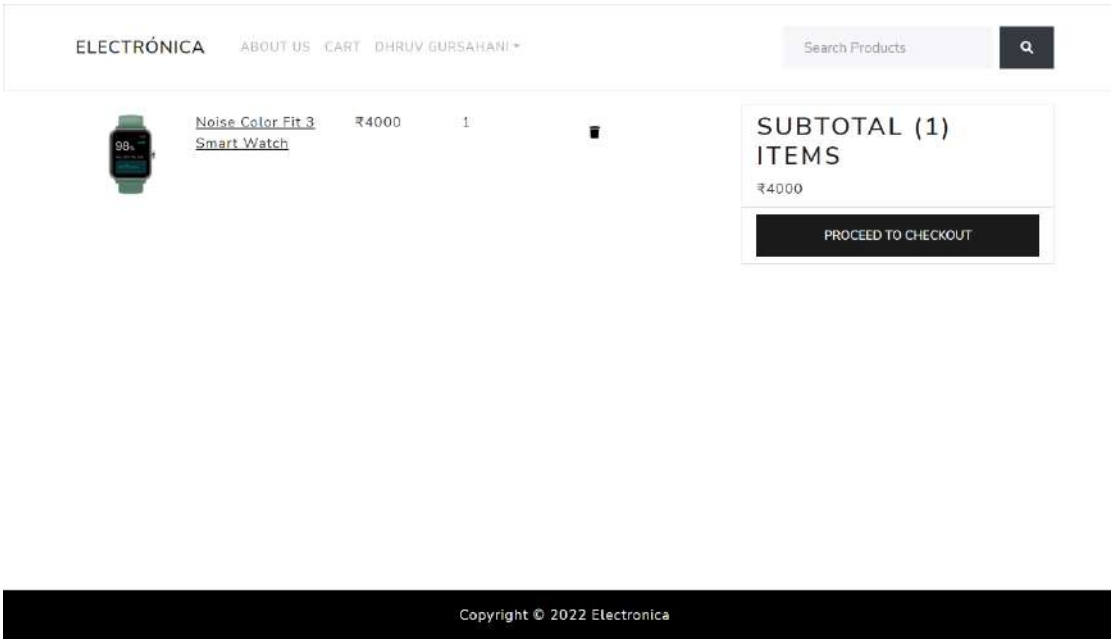
DFD Level 1 provides a more detailed breakout of pieces of the Context Level Diagram. You will highlight the main functions carried out by the system, as you break down the high-level process of the Context Diagram into its subprocesses.

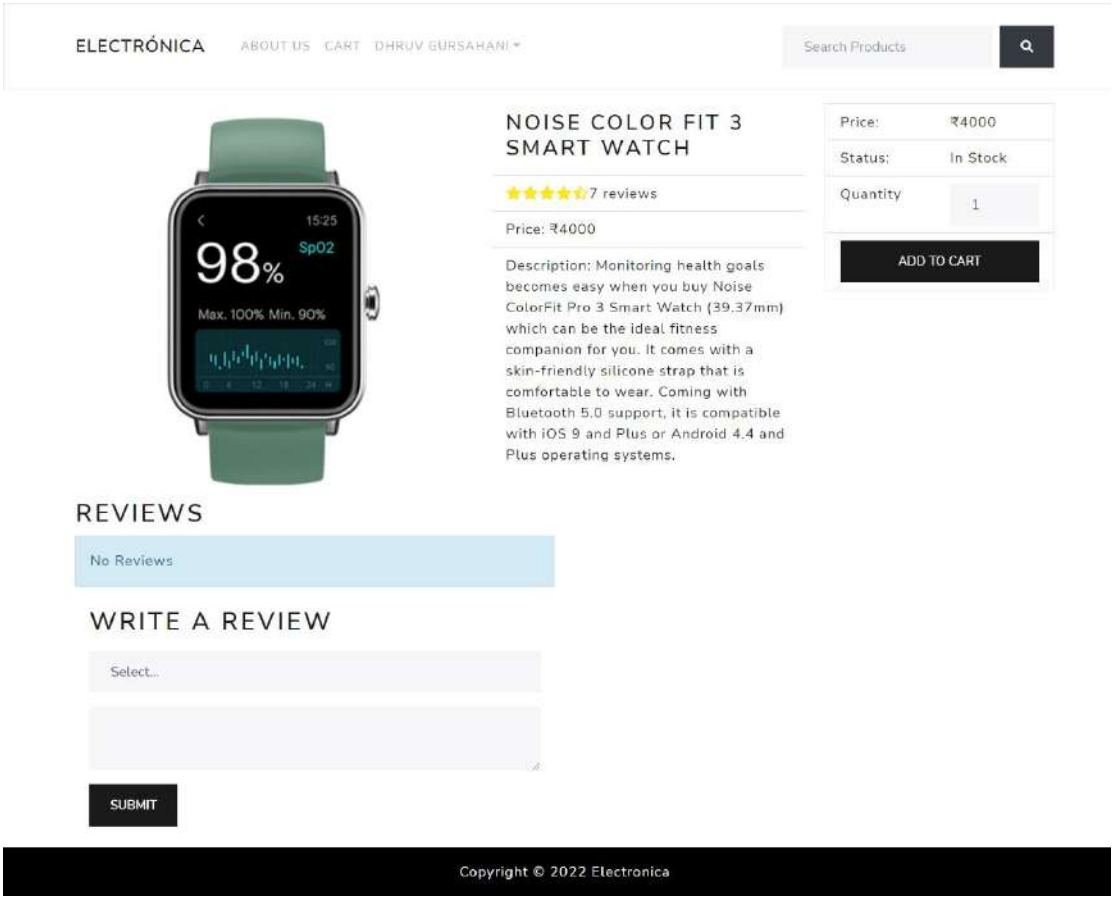


Level 2 DFD



3.4 INTERFACE/OUTPUT DESIGN





ELECTRÓNICA ABOUT US CART DHRUV GURSAHANI Search Products

LATEST PRODUCTS

Apple Watch 6
10 reviews
₹36000

Redmi 10 Prime (6GB RAM)
2 reviews
₹15000

Apple Earpods (In-Ear Headphones with Lightning Connector)
5 reviews
₹1500

Xiaomi Car Charger 18 Watts
5 reviews
₹800

Noise Color Fit 3 Smart Watch
7 reviews
₹4000

Canon 24.1MP DSLR Camera
3 reviews
₹56990

iPad 64GB
1 reviews
₹29000

Apple MacBook M1
10 reviews
₹100000

Airpods 3
3 reviews
₹18000

OPPO A53 5G (8GB RAM, 128GB ROM)
3 reviews
₹17990

Lenovo IdeaPad Slim 10th gen i5, Windows 11
4 reviews
₹53490

iPhone 12 256GB
5 reviews
₹50000

1 2

Copyright © 2022 Electronica

ELECTRÓNICA ABOUT US CART DHRUV GURSAHANI Search Products

ORDER SUMMARY

SHIPPING

Name: Dhruv Gursahani

Email: dhruv@example.com

ADDRESS: 1234 Paschim Vihar, NEW DELHI, 110063, , India

NOT DELIVERED

PAYMENT METHOD

METHOD: Stripe

NOT PAID

ORDER ITEMS

 Noise Color Fit 3 Smart Watch 1 x ₹4000 = ₹4000

SUB-TOTAL	₹4000
SHIPPING	₹100
TAX	₹720
TOTAL	₹4820

VISA 4242 04 / 24 242 42422

PAY

Copyright © 2022 Electronica

ELECTRÓNICA ABOUT US CART DHRUV GURSAHANI Search Products

USER PROFILE

Dhruv Gursahani

dhruv@example.com

Enter Password

Confirm Password

UPDATE

MY ORDERS

ID	DATE	TOTAL	PAID	DELIVERED	
61e8f24820b201ddc125acae	2022-01-20	₹4820	✕	✕	DETAILS

Copyright © 2022 Electronica

CHAPTER 4	**TESTING AND IMPLEMENTATION**	**21- 38**
4.1	Testing Methodology (Types)	22-23
4.2	Testing methodology applied (why)	24
4.3	Test Cases	25-26
4.4	Characteristics of a good Test Cases	27
4.5	Gap Analysis	28
4.6	Rework / Retest	29
4.7	Hardware and Software Requirement	30

4.1 TESTING METHODOLOGY

Unit testing:

It is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. Unit Testing is done during the development (coding phase) of an application by the developers. Unit Tests isolate a section of code and verify its correctness. A unit may be an individual function, method, procedure, module, or object.

In SDLC, STLC, V Model, Unit testing is first level of testing done before integration testing. Unit testing is a White Box testing technique that is usually performed by the developer. Though, in a practical world due to time crunch or reluctance of developers to tests, QA engineers also do unit testing.

Unit testing advantages:

- Developers looking to learn what functionality is provided by a unit and how to use it can look at the unit tests to gain a basic understanding of the unit API.
- Unit testing allows the programmer to refactor code at a later date, and make sure the module still works correctly (i.e. Regression testing). The procedure is to write test cases for all functions and methods so that whenever a change causes a fault, it can be quickly identified and fixed.
- Due to the modular nature of the unit testing, we can test parts of the project without waiting for others to be completed.

Integration Testing:

INTEGRATION TESTING is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated. Integration Testing focuses on checking data communication amongst these modules.

Different types of integration testing are:

- Bottom-up Integration Testing is a strategy in which the lower level modules are tested first. These tested modules are then further used to facilitate the testing of higher level modules. The process continues until all modules at top level are tested. Once the lower level modules are tested and integrated, then the next level of modules are formed.

- Top down Integration Testing is a method in which integration testing takes place from top to bottom following the control flow of software system. The higher level modules are tested first and then lower level modules are tested and integrated in order to check the software functionality. Stubs are used for testing if some modules are not ready.

System Testing:

System testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system.

Two Category of Software Testing:

- Black Box Testing
- White Box Testing

User Acceptance Testing (UAT)

It is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done. The main Purpose of UAT is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is kind of black box testing where two or more end-users will be involved.

4.2 TESTING METHODOLOGY APPLIED

Unit testing:

Development of our project happened in way where we me and my partner divided the website into different modules namely the cart module, checkout module, login member module and so on. So during testing, applying unit testing was the only way forward as it would make no sense to integrate different modules (mentioned above) without testing their individual functionality.

In order to do Unit Testing, we wrote a section of code to test a specific function in website. We also isolated this component/login to test more rigorously which reveals unnecessary dependencies between component/login being tested and other units so the dependencies can be eliminated. Developers generally use UnitTest framework to develop automated test cases for unit testing.

Unit Testing is of two types

- Manual
- Automated

We applied the manual method where we entered various test cases over and over into the project.

Integration testing:

Once we knew that are individual modules worked fine we integrated all those modules into a single software and vigorously tested with help of "top-down integration method"

.

4.3 TEST CASES

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TU01	Check customer login with valid data	Goto the website Enter Email Enter Password	Username=john@example.com Password=b878gud3P3Y*5@	User should be able to login	As expected	Pass
TU02	Check customer login with invalid data	Goto the website Enter Email Enter Password	Username=john@example.com Password=2556	User should not be able to login	As expected	Pass
TU03	Check Admin login with valid data	Goto the website Enter Email Enter Password	Name = admin@example.com Password= !V4p6wCU****	Admin will be logged in and should be able to see admin panel	As Expected	Pass
TU04	Add New Product (Admin)	Login Navigate to Products Page Click on New Product Enter Details Click Update	Name = iPhone 13 Price = 50,000 Image Brand = Apple Category = Mobile Phone Description = New Product from Apple	New Product with the given information should be added to databses	As Expected	Pass
TU05	Delete New Product (Admin)	Login Navigate to Products Page	N.A.	The product should be removed	As Expected	Pass

		Click on Bin Icon next to the product		from the database		
TU06	Mark an Order as Delivered (Admin)	Login Navigate to Orders Page Select an Order Click on "Mark as Delivered" button	N.A.	User should be able to see his/her order as delivered	As Expected	Pass
TU07	Purchase a Product	Login Add a product to cart Proceed to checkout and enter address Make payment by Card	N.A.	Once the user click the "Pay" button the Order should be placed.	As Expected	Pass
TU08	Make another user admin (Admin)	Login Navigate to Users Page Select a User Check the isAdmin checkbox Click "Update" button	N.A.	The web app user should be made an admin. This should make the products and orders page visible to him/her	As Expected	Pass

4.4 CHARACTERISTICS OF A GOOD TEST CASE

- As far as possible, write test cases in such a way that you test only one thing at a time. Do not overlap or complicate test cases. Attempt to make your test cases ‘atomic’.
- Ensure that all positive scenarios and negative scenarios are covered.
- Language:
- Write in simple and easy-to-understand language.
- Use active voice instead of passive voice: Do this, do that.
- Use exact and consistent names (of forms, fields, etc).
- Characteristics of a good test case:
- Accurate: Exacts the purpose.
- Economical: No unnecessary steps or words.
- Traceable: Capable of being traced to requirements.
- Repeatable: Can be used to perform the test over and over.
- Reusable: Can be reused if necessary.

4.5 GAP ANALYSIS

Gap analysis is a way of comparing your current business performance to your long-term goals.

Also known as ‘needs analysis’ ‘need assessment’ and ‘needs-gap analysis’, gap analysis is a way to determine two key things. First, where you’re falling short of your goals. Second, what you need to do to get back on track.

Step one: identify your current state from the test case

The first step is knowing where you are at the present time. So, be clear as to what is being described and what is not. This will avoid scope creep and keep your analysis focused. Then comes collecting contextual information. That means collecting qualitative information, such as what are your team processes and methodologies.

Also, quantitative information is important, meaning anything that can be counted and measured. That includes everything you’re currently doing.

> **In context of the project**: From the test cases applied above and the result of testing we can confidently say that our software has reached our end goal and is working fine.

Step two: outline your desired future state

The point of a gap analysis is to figure out where you want to go and if you’re getting there. This is the desired state, future target or stretch goal. To get there you need to know about your current state and what a reasonable timeframe is to get from there to the goal you’ve set for yourself.

But first, you must mark that point in the future that you’re aiming for. Think about where that should be, what’s not happening that needs to happen, what could be happening that hasn’t before or has changed, and what needs to happen to get there.

> **In context of the project**: The desired future state of our software was to create an efficient and scalable E-Commerce platform. Which has been properly achieved

Step three: identify the gaps

You know where you are and where you want to go, the space between those two marks is the gap you must bridge to reach your target. This is when you want to figure out why there is a gap. To do that you need to be very specific about the gap. Also, dig deeper and determine why the gap happened. Ask yourself questions that are applicable to your business and answer them honestly**.**

> **In context of the project**: Given that our software is working at the level we wanted it to so for now there are no gaps

Step four: **work towards filling the gap**

You've done the due diligence, and now it's time to act. You know why there's a gap, so you must now devise a way to close it. To do this, you can follow the guidelines of basing all improvements on the information you discovered when you identified the gap. Also, consider the cost of implementation for each solution that you come up with; you might not have the capital or capacity to achieve it. Finally, figure out the date at which the gap will be closed. If you don't have a deadline, it will get overlooked or ignored.

> **In context of the project**: Since website is working up to the mark there are no gaps to work towards. But there is new functionality that we would like to add which is discussed in the subsequent chapters.

4.6 REWORK

As identified in the gap analysis there are no gaps or errors present for now in our software so no rework is required but errors might creep up later during usage so they would be fixed by updating the code.

4.7 HARDWARE AND SOFTWARE REQUIREMENTS

4.7.1 Hardware requirement

- Processor : Intel Core i3 Processor (OR ABOVE)
- RAM : 4GB RAM
- HDD : 20GB

4.7.2 Software requirement

- OS: Microsoft Windows 8.1 (OR ABOVE)
- Browser: Last 10 versions of Google Chrome / IE / Firefox

CHAPTER 5	**CONCLUSION AND REFERENCES**	**31-35**
5.1	Conclusion	32
5.2	Limitation of the System	33
5.3	Future Scope for Modification	34
5.4	References/ Bibliography	35

5.1 CONCLUSION

Our system provides different services like easy access to access customer details, efficient management, and clear productivity while being less time consuming. User friendly coding is adopted the objective of project is to provide a solution to the business owner which will make their work easy by removing the need to create, and track hand written files in order to manage their business

5.2 LIMITATION

- The product images are not compressed.
- Aggregated sales data cannot be represented through charts/graphs.
- Data once deleted cannot be recovered.
- It is not multilingual meaning people who do not know English cannot operate this software.
- Does not have analytics involved, meaning we cannot monitor consumer behaviour.
- For persistence, cart items are stored in local storage of browser which is un-encrypted.

5.3 FUTURE SCOPE OF MODIFICATION

- Integrate Google Analytics
- Search Engine Optimization.
- Include “you might also like” section in the web app

5.4 REFERENCES/BIBLIOGRAPHY

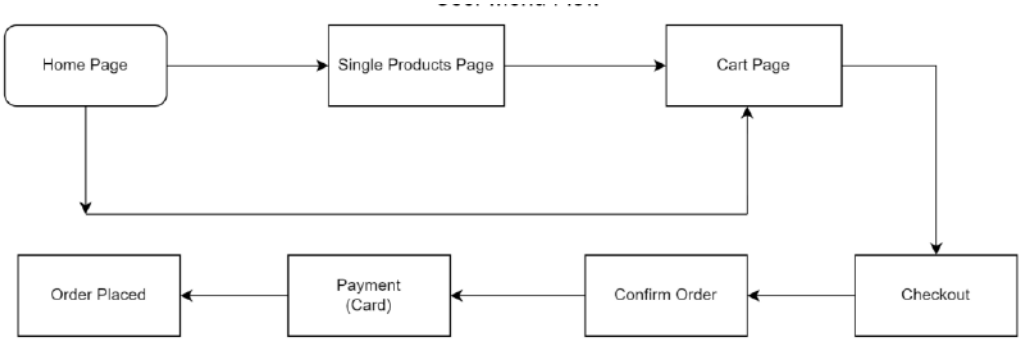
React - The Complete Guide (incl Hooks, React Router, Redux)

The Complete 2022 Web Development Bootcamp

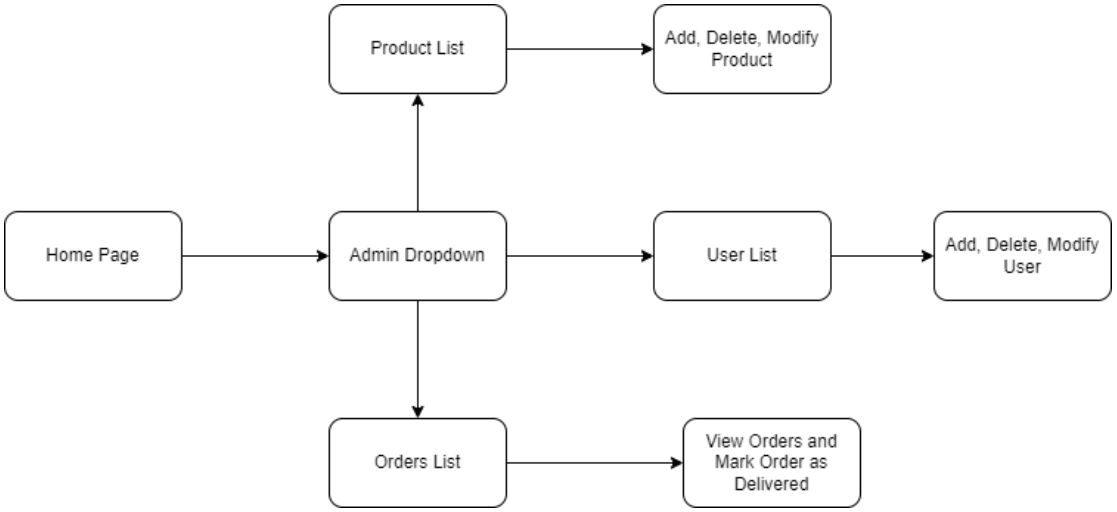
CHAPTER 6	**ANNEXURES**	**36-131**
A-1	Menu Flow Diagram	37
A-2	Sample Input	38-41
A-3	Sample Output	42-43
A-4	Program Code	44-130
A-5	Mentee Feedback	131

A-1 MENU FLOW DIAGRAM

User's Menu Flow

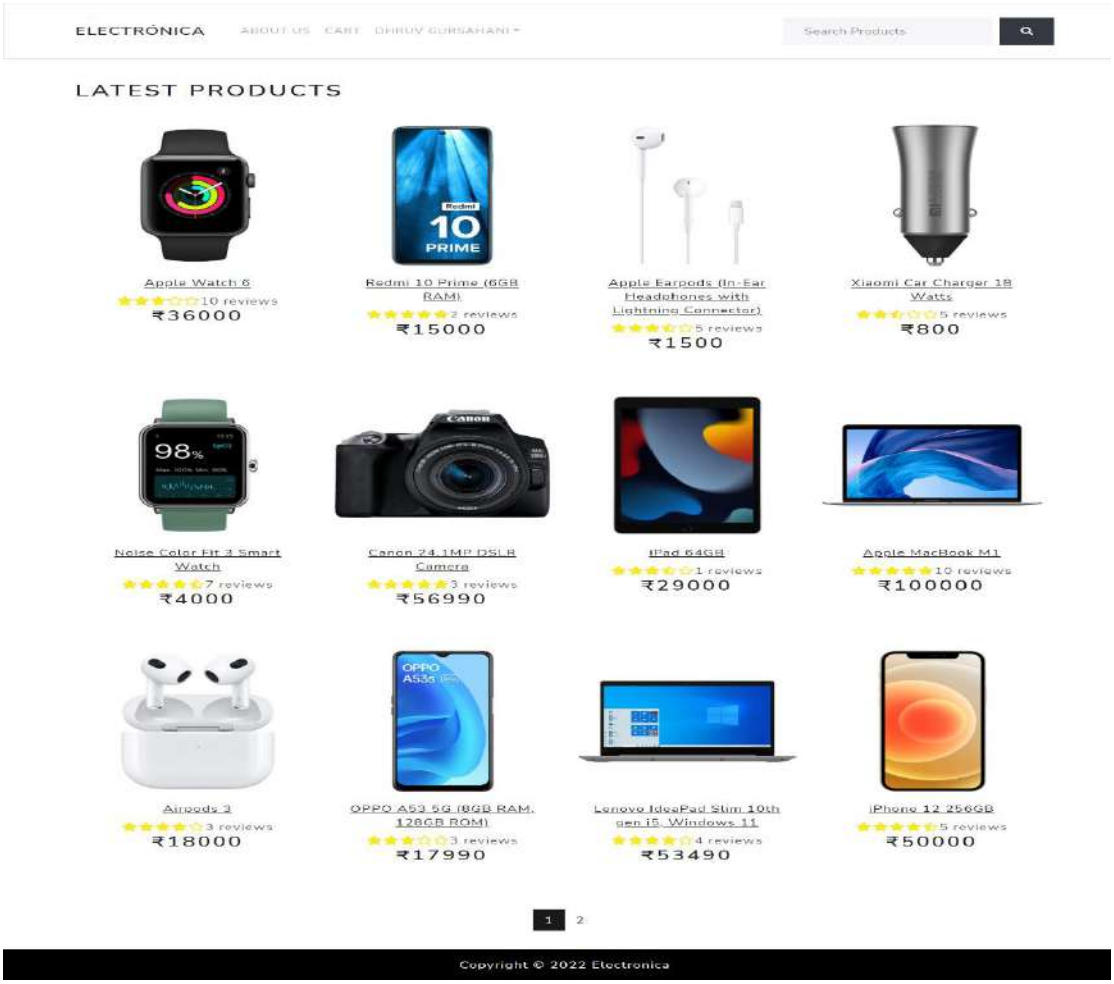


Admin's Menu Flow

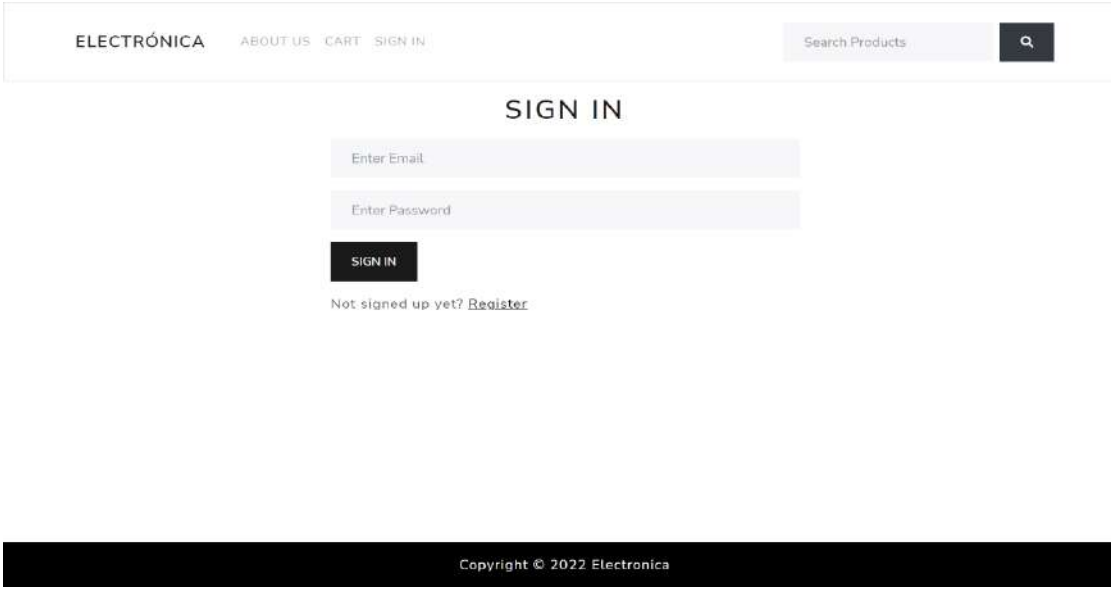


A-2 SAMPLE INPUT

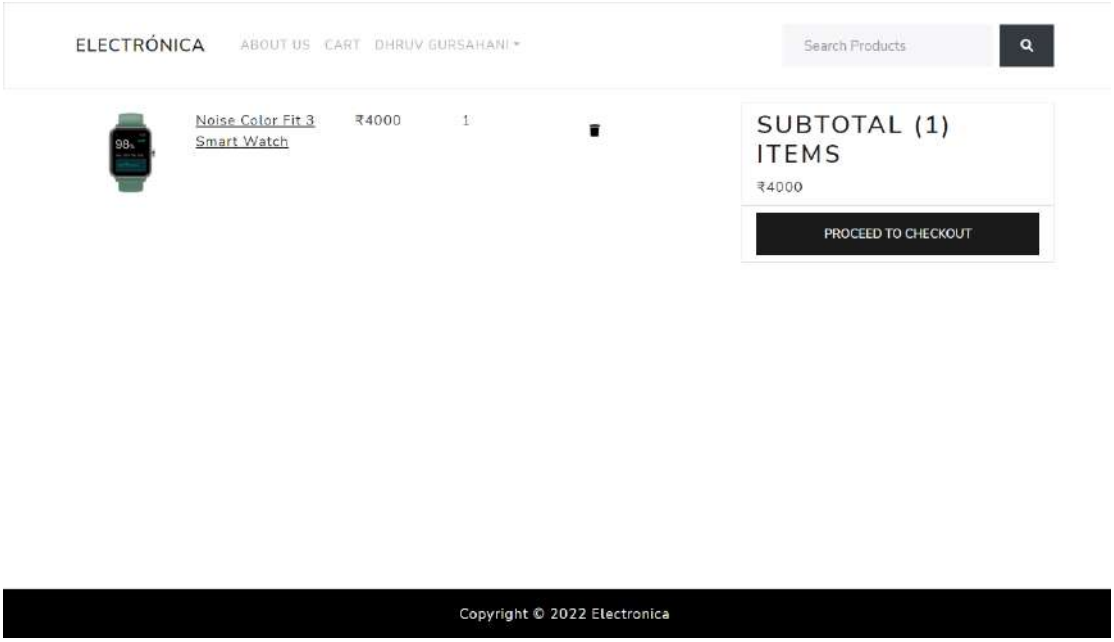
1. Home Page



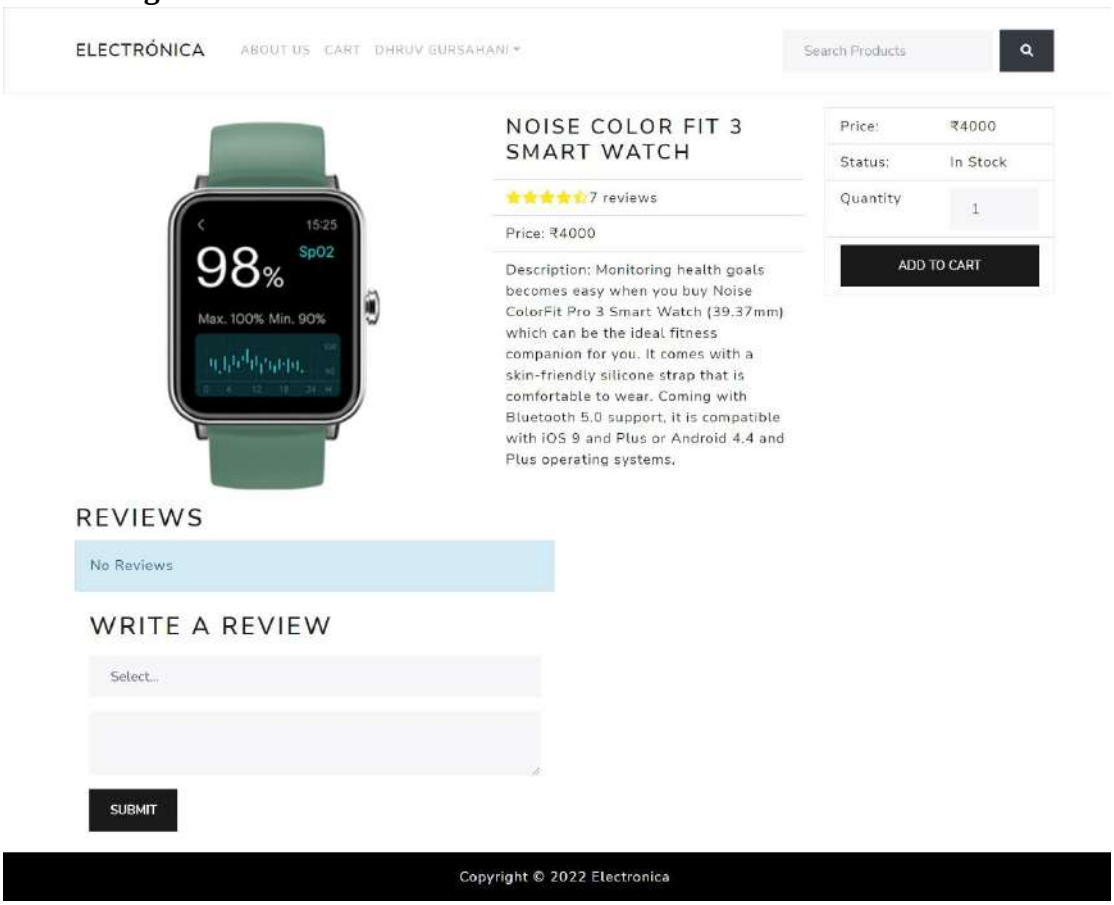
2. Login Screen



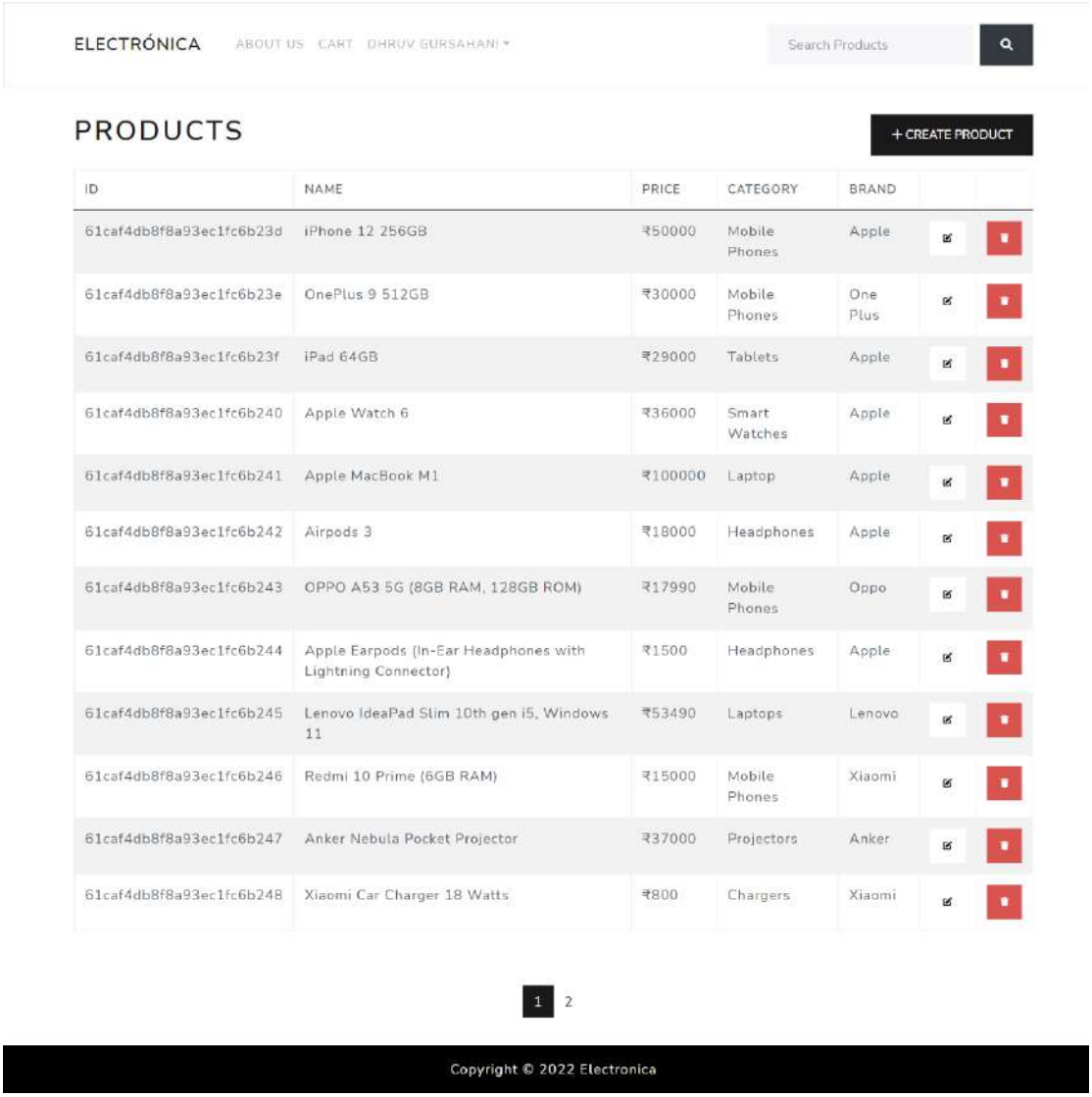
3. Cart Screen



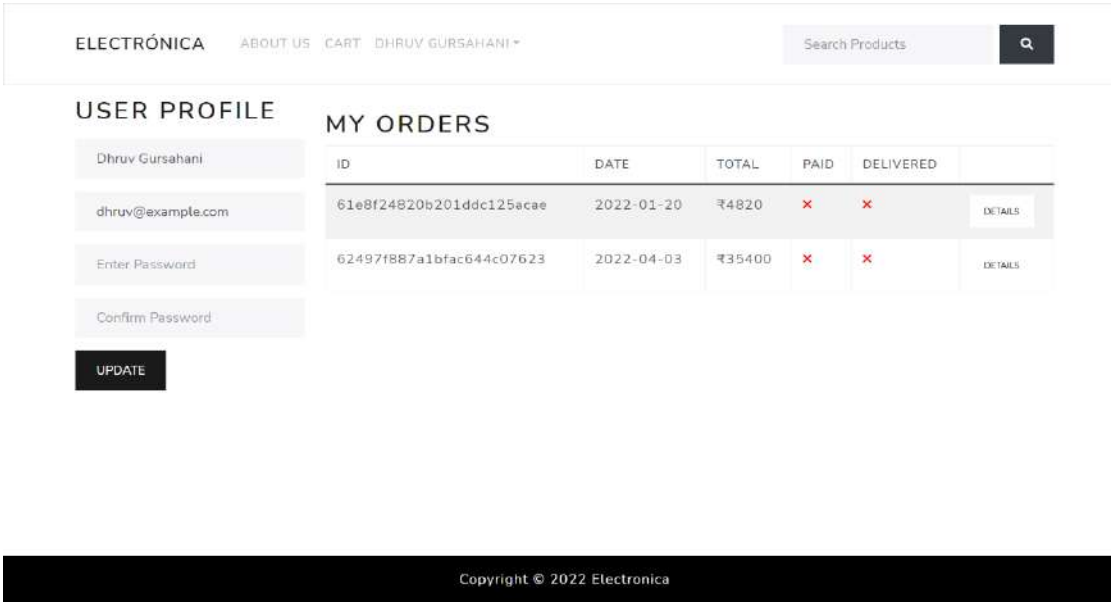
4. Single Product Screen



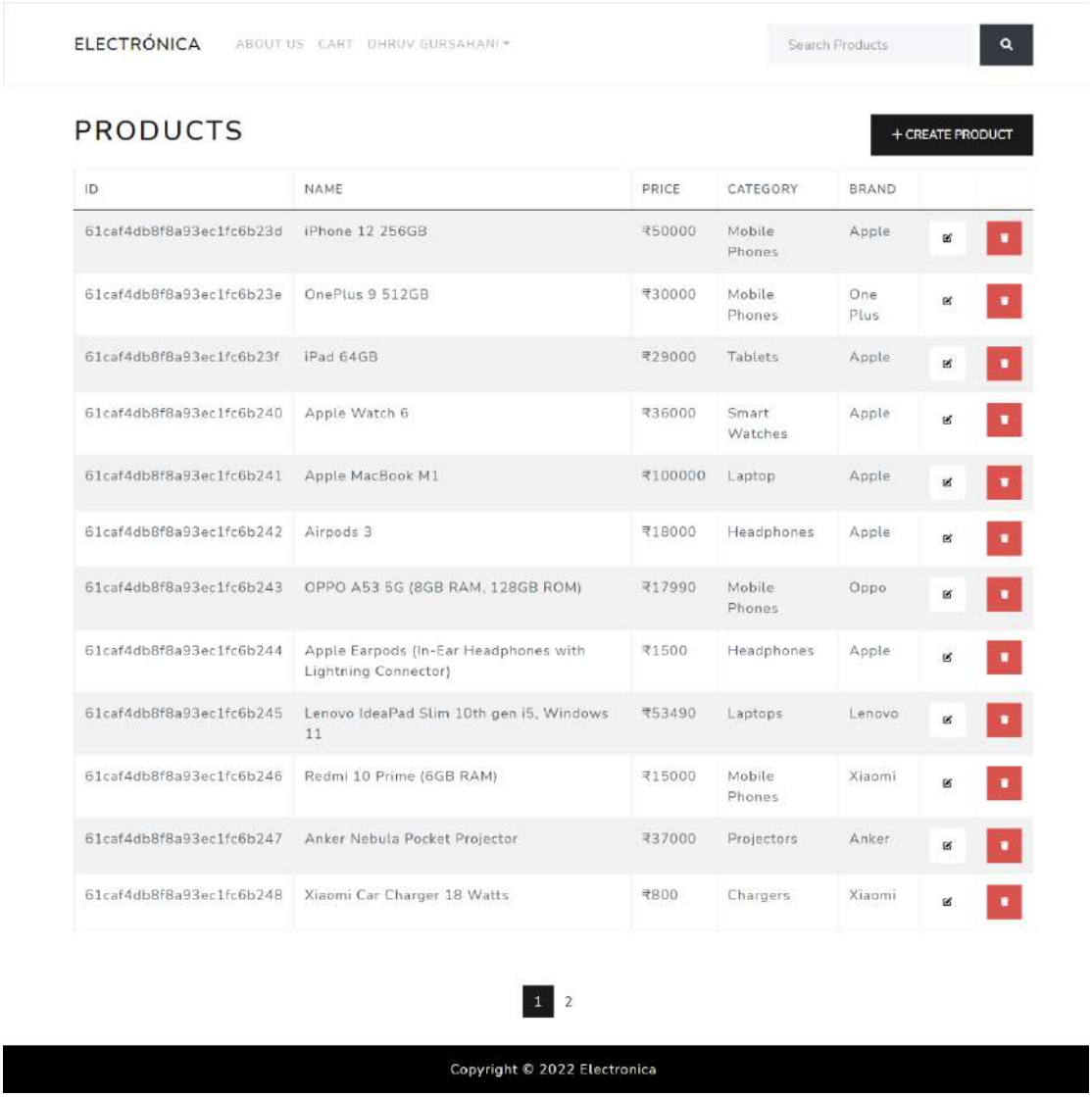
5. Products List Screen



6. Profile Screen

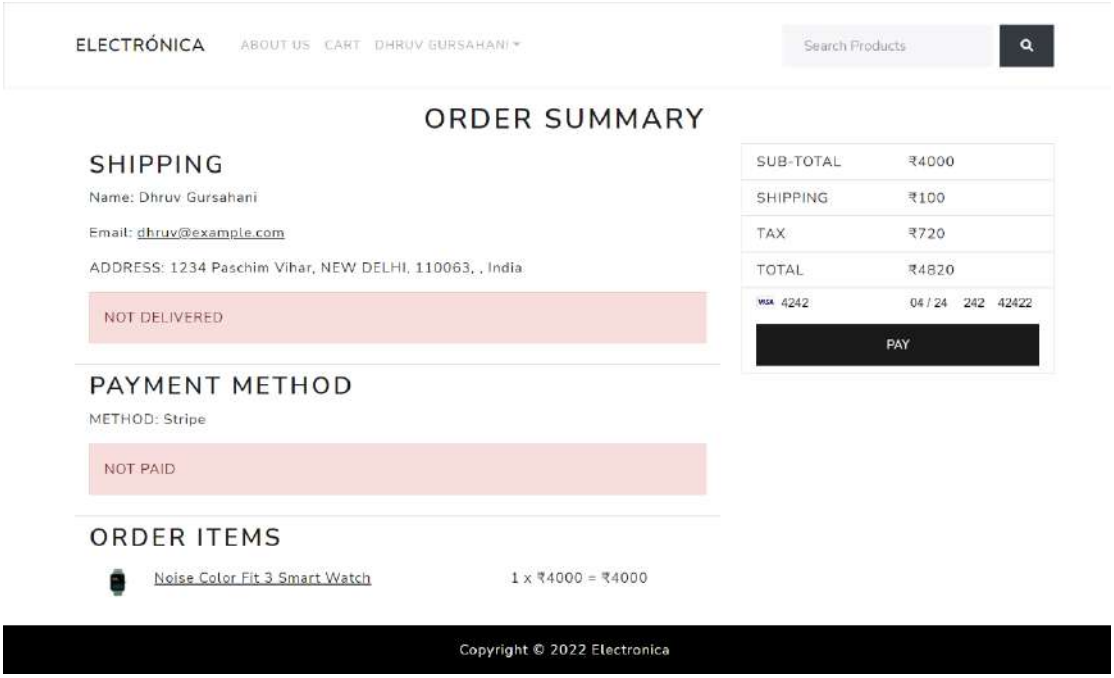


7. Order List Screen

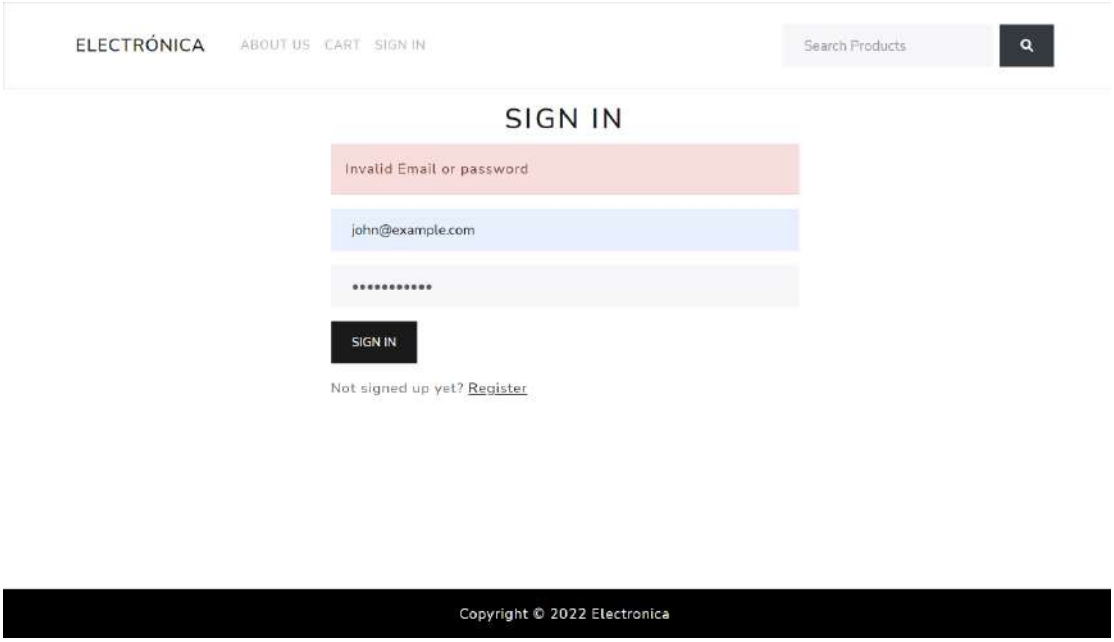


A-3 SAMPLE OUTPUT

1. Final Order Page



2. Invalid Login



A-4 PROGRAM CODE