

URBAN EDGE

**A Report of Mini Project work submitted in partial
Fulfillment of the requirements for the Degree of**

BACHELOR OF COMPUTER APPLICATIONS

Submitted By:

P Krishnadev

Reg.No: 220021083832

Neswin George

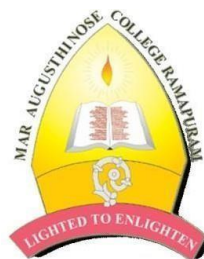
Reg.No:220021083827

Joyal Thomas

Reg.No:2200210838

15

Under the guidance of Ms. Alphonsa Jose, ME
(Assistant Professor, Department of Computer Science)



Department of Computer Science

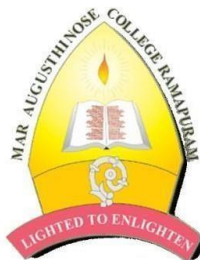
**MAR AUGUSTHINOSE COLLEGE,
RAMAPURAM
(Affiliated to Mahatma Gandhi University)**

Ramapuram - 686576

2022-2025

**MAR AUGUSTHINOSE COLLEGE
RAMAPURAM**

(Affiliated to Mahatma Gandhi University)



CERTIFICATE

This is to certify that this Project work entitled “**URBAN EDGE**” is a bonafide report of the mini project done by **P KRISHNADEV (Reg No: 220021083832)**, **NESWIN GEORGE (Reg No:220021083827)** and **JOYAL THOMAS (Reg No:220021083815)** during the academic year 2024- 2025 for the partial fulfillment for the award of the Degree of Bachelor of Computer Applications from Mahatma Gandhi University, Kottayam.

Mr. Prakash Joseph MCA

Ms. Alphonsa Jose ME

Head of the Department

Assistant Professor

Viva – voice Examination conducted onat
Mar Augusthinose College, Ramapuram

Internal Examiner

External Examiner

DECLARATION

We , **P Krishnadev, Neswin George and Joyal Thomas** , hereby declare that the project work Entitled “**URBAN EDGE** ” is a record of bonafide project carried out by us under the supervision and guidance of **Ms. Alphonsa Jose ME** , Assistant Professor , Department of Computer Science , Mar Augusthinose College , Ramapuram. We also declare that it has not been previously submitted for the award of any Degree, Diploma or similar titles by anyUniversity or similar other institutions.

**P KRISHNADEV
NESWIN GEORGE
JOYAL THOMAS**

Place : Ramapuram

Date :

ACKNOWLEDGEMENT

Firstly, We thank **God Almighty**, whose blessings were always with us and helped us to complete this project work successfully.

We acknowledge our deepest gratitude to **Dr. Joy Jacob MA, Mphil, PhD** principal for permitting us to do this project.

We express our sincere and profound thanks to **Mr. Prakash Joseph MCA**, Head of the Department of computer science, for giving us an opportunity to do this project.

We express our greatest gratitude to **Ms Alphonsa Jose ME**, Assistant professor, Mar Augusthinose College, Ramapuram for her valuable guidance and encouragement for completing this project.

Words are boundless to express our gratitude to all the faculty and members of the Computer Science Department for their timely help and support.

I wish to extend our grateful thanks to **Mr. Ditto Joseph** Computer Programmer and System Administrator for his technical support, valuable advice and guidance throughout the project.

We wish to extend our grateful thanks to management and staff, of **Mar Augusthinose College, Ramapuram**, for their valuable support to carry out this project work.

Finally, We express our deep appreciation to all our friends and family members for the moral support and encouragement they have given to complete this project successfully.

**P KRISHNADEV
NESWINGEORGE
JOYAL THOMAS**

ABSTRACT

The constant evolution of digital technologies has reshaped the way customers interact with businesses particularly in the realm of e-commerce. The primary goal of e-commerce websites is to sell goods and services online. This project builds a user-friendly e-commerce website for URBAN EDGE, a brand specializing in high quality t-shirts. HTML, CSS, JavaScript, PHP and MySQL power the website enabling customers to browse select and purchase URBAN EDGE t-shirts online. Customers can explore the complete URBAN EDGE collection. Product pages will showcase detailed information including high quality images and descriptions. Users can seamlessly add desired t-shirts to their shopping cart and finalize the purchase. A dedicated admin page empowers URBAN EDGE to manage their online store. The project offers a practical learning experience by implementing essential web development technologies to create functional e-commerce website.

TABLE OF CONTENTS

1. Introduction	8
1.1 Software Introduction	9
1.2 Background and Motivation	9
2. System Study	12
2.1 System Study	13
2.2 Existing System	14
2.3 Proposed System	14
3. System Analysis	15
3.1 Introduction	16
3.2 Stakeholders In The Project	16
3.3 Feasibility study	17
3.4 Software Development Lifecycle Model	18
3.5 Hardware & Software Requirements	19
4. System Design	23
4.1 Input Design	24
4.2 Output Design	26
4.3 Database Design	27
4.4 Tables	30
4.5 Data Flow Diagram	34
4.6 Sitemap Diagram	39

5. System Testing	40
5.1 Testing Procedures	41
5.2 Testing Methodologies	41
6. Implementation and Maintenance	44
7. Future Scope Of The Project	47
8. Conclusion	49
9. Bibliography	51
10. Screenshots	53

INTRODUCTION

1 INTRODUCTION

1.1 SOFTWARE INTRODUCTION

The URBAN EDGE T-SHIRTS Online Shopping Platform is designed to make buying T-shirts easy and enjoyable for men, women, and children. It combines a wide selection of T-shirts with convenient shopping features to ensure a smooth experience for all customers.

This platform allows customers to browse, select, and purchase T-shirts from a variety of styles and brands. Users can easily access product details, including high-quality images, descriptions, and prices.

Key features include:

- **Easy Browsing:** Customers can quickly find T-shirts by navigating through categories or using the search function to discover specific styles.
- **User-Friendly Interface:** The website is designed to be intuitive and easy to navigate, making the shopping experience enjoyable for everyone.
- **Secure Payments:** Customers can shop with confidence, knowing that their payment information is protected.
- **Customer Reviews:** Shoppers can read reviews from other customers to help them make informed choices.

URBAN EDGE T-SHIRTS aims to provide a central platform for all T-shirt needs, making it simple to find the perfect T-shirt for every occasion.

BACKGROUND AND MOTIVATION

The Online T-Shirt Shopping System aims to create an efficient and user-friendly platform for purchasing T-shirts specifically designed for men and women. As the popularity of online shopping continues to grow, providing a streamlined and enjoyable shopping experience is essential for both customers and retailers. This project is designed to enhance the T-shirt buying experience by offering customers a variety of styles, sizes, and designs tailored to different preferences. By optimizing features such as inventory management, order tracking, and customer service, the system ensures a smooth experience for buyers and supports sellers in managing their products effectively.

BACKGROUND

- **Market Demand:**

T-shirts are a staple clothing item across various demographics, with demand year-round, especially for trending designs and seasonal collections. An online shopping platform can reach a broad audience, helping customers find their ideal T-shirts from a variety of brands, styles, and sizes tailored for men and women.

- **Efficiency:**

Traditional retail shopping can be time-consuming and often limits variety. An online store centralizes a large selection of T-shirts, allowing customers to browse, compare, and purchase at their convenience, enhancing the efficiency and ease of the shopping process.

- **Accessibility:**

An online platform increases accessibility by enabling customers to shop for T-shirts from anywhere, regardless of their physical location. This makes the shopping process more inclusive and convenient for a wider customer base.

- **User Experience:**

A well-designed website improves user experience by providing search filters, detailed product descriptions, customer reviews, and size guides, making it easier and more enjoyable for users to find T-shirts that match their style and fit preferences.

- **Data Management:**

An online system streamlines product listings, customer inquiries, and order histories, reducing the administrative workload for the business while providing customers with a seamless shopping experience.

- **Safety:**

A reputable online store includes secure payment options, verified reviews, and hassle-free return policies, helping customers feel confident and safe about their purchases.

MOTIVATIONS

- **Convenience:**
An online store makes it easy to browse, compare, and purchase T-shirts in one place, saving time and reducing hassle for customers.
- **Transparency:**
Clear product information, including photos, descriptions, and prices, helps customers make informed choices and avoid surprises.
- **Innovation:**
Using technology like virtual try-ons, detailed product search, and secure online payments creates a more modern and engaging shopping experience.
- **Scalability:**
A well-designed website can easily grow, adding new products, features, or expanding to serve more regions, making it a scalable business model.

The aim of the URBAN EDGE T-SHIRTS project is to create a user-friendly platform that makes it easy to browse, search, and purchase T-shirts for all ages. By offering a wide range of styles, the website will provide detailed product descriptions, high-quality images, and essential fit and style information. It aims to enhance the shopping experience with secure payments, user reviews, and personalized recommendations, making the shopping process convenient and enjoyable. Through technology, the project seeks to simplify purchasing for customers and boost sales for the business.

SYSTEM STUDY

2 SYSTEM STUDY

2.1 SYSTEM STUDY

System study is in detailed analysis of various operations and their relationship within and outside the system. It is the first step in developing and managing systems.

o Fact Finding Techniques:

The specific methods analysts use for collecting data about requirements are called fact-finding techniques. These include the interview, questionnaire, record inspection and observation.

o Record Review:

Many kinds of records and reports can provide analysis with valuable information about organizations and operations. In record reviews, analysts examine information that has been recorded about the system and users. Records include written policy manuals, regulations and standard operating procedures used by most organization as a guide for managers and employees.

o Observation:

Observation allows analysts to gain information they can't obtain by any other fact-finding method. Through observation, analysts can obtain first-hand information about how activities are carried out. This method is most useful when analysts need to actually observe. How documents are handled, how processes are carried out and whether specified steps actually followed.

o Interview:

Analysts use interviews to collect information from individuals or from groups. The respondents are generally current users of the existing system or potential users of the proposed system. In some instances, the respondents may be managers or employees who provide data for the proposed system or who will be affected by it. It is important to remember that respondents and analysts converse during an interview, the respondents and analysts with opportunities for gathering information from respondents who have been chosen for their knowledge of the system under study. This method is frequently the best source of study.

2.2 EXISTING SYSTEM

A company named Denim republic, specializing in T-shirts, faces several challenges with traditional in-store shopping. Customers must visit physical stores, where they often struggle with limited stock availability, making it difficult to find the exact style, color, or size they want. This results in wasted time navigating crowded aisles, only to discover that their preferred T-shirt is out of stock or unavailable in the desired style.

Denim republic also experiences difficulties with effective inventory management, as stock levels are not always accurately reflected, leading to customer frustration and missed sales opportunities. Additionally, shoppers are limited by store hours and often have to visit multiple locations to compare prices or styles, creating a cumbersome and inefficient shopping process.

By creating an online platform, Denim republic could address these issues, offering customers a seamless shopping experience with access to a broad selection of T-shirts, real-time stock information, and the convenience to browse and purchase anytime, anywhere. This transition would streamline stock management, improve customer satisfaction, and optimize the shopping experience.

2.3 PROPOSED SYSTEM

The proposed URBAN EDGE Online Shopping Platform is designed to enhance the shopping experience by overcoming the limitations of traditional retail methods. Built with HTML, CSS, and JavaScript for the front end, and PHP with MySQL for the backend, this interactive platform offers a smooth and engaging user experience while effectively reducing operational costs. Key features include easy product browsing, allowing customers to explore a wide variety of T-shirts for men, and women. Secure payment processing ensures that customers can shop with confidence, knowing their financial information is protected. User account management enables customers to track their orders, save preferences, and receive personalized recommendations. Additionally, administrators can efficiently manage product listings and monitor sales reports, enhancing overall operational efficiency. With robust user authentication measures in place, the platform safeguards personal information, maintaining customer trust and confidentiality. By leveraging modern web technologies, URBAN EDGE aims to provide a centralized online shopping experience that makes finding and purchasing T-shirts convenient, efficient, and enjoyable for everyone.

SYSTEM ANALYSIS

3 SYSTEM ANALYSIS

3.1 INTRODUCTION

Software Engineering is the analysis, design, construction, verification and management of technical or social entities. To engineer software accurately, a software engineering process must be defined. System analysis is a detailed study of the various operations performed by the system and their relationship within and module of the system. It is a structured method for solving the problems related to the development of a new system. The detailed investigation of the present system is the focal point of system analysis. This phase involves the study of present system and identification of system objectives. Information has to be collected from all people who are affected by or who use the system. During analysis, data are collected on the variable files, decision point and transactions handled by the present system. The main aim of system is to provide the efficient and user-friendly automation. So, the system analysis process should be performed with extreme precision, so that inaccurate picture of existing system, its disadvantages and the requirements of the new system can be obtained.

System analysis involves gathering the necessary information and using the structured tool for analysis. This includes the studying existing system and its drawback, designing a new system and conducting cost benefit analysis. System analysis is a problem-solving activity that requires intensive communication between the system users and system developers. The system is studied to the minute detail and analyzed. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through various phases of processing of inputs.

3.2 STAKEHOLDERS IN THE PROJECT

3.2.1 CUSTOMER

Customers are the primary users and key stakeholders of the system. The system is designed to make it easy for customers to buy T-shirts online. It offers a variety of features to enhance the shopping experience. Customers can quickly register on the platform, browse through a wide selection of T-shirts, and choose the design that matches their style and preferences. Once a selection is made, customers can place orders online and make payments securely.

3.2.2 ADMIN

The admin single-handedly oversees the system, and there is only one admin within the system. Just like any system, issues can arise, and it is the admin's responsibility to handle those problems, such as managing disputes or ensuring that customers adhere to the system's guidelines. The admin monitors the activity of each customer and has the authority to add, update, and delete products. Additionally, the admin manages orders, including approving or rejecting them and verifying the payment status.

3.3 FEASIBILITY STUDY

Feasibility analysis is a system proposal according to its workability, impact on the organization, ability to meet client and user needs and efficient use of resources. The key considerations that are involved in the feasibility analysis are

1. Technical Feasibility
2. Economic Feasibility
3. Operational Feasibility

TECHNICAL FEASIBILITY

Technical feasibility concentrates on the existing computer system (hardware, software etc.) and to what extent it can support the proposed software. The hardware and software requirements are industry standards. Here no extra expenditure is expected. This system is technically feasible. The considerations that are normally associated with technical feasibility include:

- **Development risk** can the system element be designed so that the necessary function and performance is achieved within the constraints uncovered during the analysis.
- **Resource availability** are competent team available to develop the system element in question, are other necessary resources (hardware & software) available to build the system.
- **Technology** has the relevant technology progressed to a state that will support the system.

ECONOMIC FEASIBILITY

Economic analysis is the most frequently used method for evaluating effectiveness of the proposed system. More commonly known as cost-benefit analysis. This procedure determines the benefits and savings that are expected from the proposed system and compared with the cost of the existing system. As this system works as a Computer based system, reduces a lot of manual effort and thus manpower cost. It also introduces faith and goodwill that can be measured as an intangible benefit. As they are generated from the Computer based system it reduces cost and time and is naturally error prone as compared to manual typewriters.

OPERATIONAL FEASIBILITY

In operational feasibility, the entire application is checked whether the system will be used if it is developed and implemented. Also it is checked whether there will be resistance from users that may undermine the possible application benefits. There is no barrier for implementing the system. The system also helps to access the information immediately as required. Thus the system is found operational feasible.

3.4 SOFTWARE DEVELOPMENT LIFECYCLE MODEL

One of the basic notions of the software development process is SDLC models which stand for Software Development Life Cycle models. SDLC – is a continuous process, which starts from the moment, when it's made a decision to launch the project, and it ends at the moment of its full remove from the exploitation. Software development lifecycle (SDLC) is a framework that defines the steps involved in the development of software. It covers the detailed plan for building, deploying and maintaining the software. SDLC defines the complete cycle of development i.e. all the tasks involved in gathering a requirement for the maintenance of a Product.

Some of the common SDLC models are Waterfall Model, V-Shaped Model, Prototype Model, Spiral Model, Iterative Incremental Model, Big Bang Model, Agile Model. We used Iterative Enhancement Model for our Project.

Iterative Enhancement Model

The iterative enhancement model is a software development methodology that focuses on incrementally improving and expanding a system through successive iterations. Each iteration begins with a simple, working version of the software and gradually adds new features or refines existing ones based on user feedback and

called the core or prototype), and with each subsequent iteration, new functionalities are integrated. This allows for continuous user involvement and product evolution as more knowledge is gained about the system's needs.

ADVANTAGES

- Involves users in every step.
- Reduces risk by identifying problems early.
- Flexible and adapts to changes.

3.5 HARDWARE AND SOFTWARE REQUIREMENTS

3.5.1 SOFTWARE SPECIFICATION

This project is built upon the latest technology software.

Front end	:	HTML, CSS, JavaScript
Development tool	:	PHP
Database	:	MySQL
Web server	:	WAMP server
Operating System	:	Windows 10

3.5.1.1 PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. As of January 2013, PHP was installed on more than 240 million websites (39% of those sampled) and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1994, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, a recursive acronym code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page - for example, PHP code can generate a web pages.

HTML code, an image, or some other data. PHP has also evolved to include a command-line interface (CLI) capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge

3.5.1.2 MySQL

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open source MySQL project to create Maria DB. High availability: Ensure business continuity with the highest levels of system availability through technologies that protect data against costly human errors and minimize disaster recovery downtime. Support can be obtained from the official manual. Free support additionally is available in different IRC channels and forums. Oracle offers paid support via its MySQL Enterprise products. They differ in the scope of services, including MariaDB and Percona. MySQL has received positive reviews, and reviewers noticed it “performs extremely well in the average case” and that the “developer interfaces are there, and the documentation (not to mention feedback in the real world via web sites and the like) is very good”. It has also been tested to be a “fast, stable and true multi-user, multi-threaded sql database server”.

3.5.1.3 WAMP SERVER

WAMP Server is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your databases. WAMP Server refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, Open SSL for SSL support, MySQL database and PHP programming language. WAMP Server is a Web development platform on Windows that allows you to create dynamic Web applications with Apache2, PHP, MySQL and MariaDB. WampServer automatically installs everything you need to intuitively develop Web applications.

3.5.1.4 VISUAL STUDIO

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including C, C#, C++, FORTRAN, Go, Java, JavaScript, Node.js, Python, Rust, and Julia. It is based on the Electron framework, which is used to develop Node.js web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (code named "Monaco") used in Azure DevOps.

Out of the box, Visual Studio Code includes basic support for most common programming languages. This basic support includes syntax highlighting, bracket matching, code folding, and configurable snippets. Visual Studio Code also ships with Intelligent for JavaScript, Type Script, JSON, CSS, and HTML. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace. Instead of a project system, it allows users to open one or more directories, which can then be saved in work spaces for future reuse. This allows it to operate as a language- agnostic code editor for any language. It supports many programming languages and a setoff features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interfacebut can be accessed via the command palette.

3.5.2 HARDWARE REQUIREMENTS

The selection of hardware configuring is a very task related to the software development, particularly inefficient RAM may affect adversely on the speed and corresponding on the efficiency of the entire system. The processor should be powerful to handle all the operations. The hard disk should have the sufficient to solve the database and the application.

Hardware used for development:

CPU	:	Intel i5 Processor
Memory	:	4 GB
Cache	:	6 MB
Hard Disk	:	1 TB
Monitor	:	15.6" Monitor
Keyboard	:	Any
Mouse	:	Any

Minimum Hardware Required for Implementation:

CPU	:	Pentium IV Processor
Memory	:	256MB Above
Cache	:	512 KB Above
Monitor	:	Any
Keyboard	:	Any
Mouse	:	Any

SYSTEM DESIGN

4 SYSTEM DESIGN

System design involves the detailed design of the system selected in the system study phase.

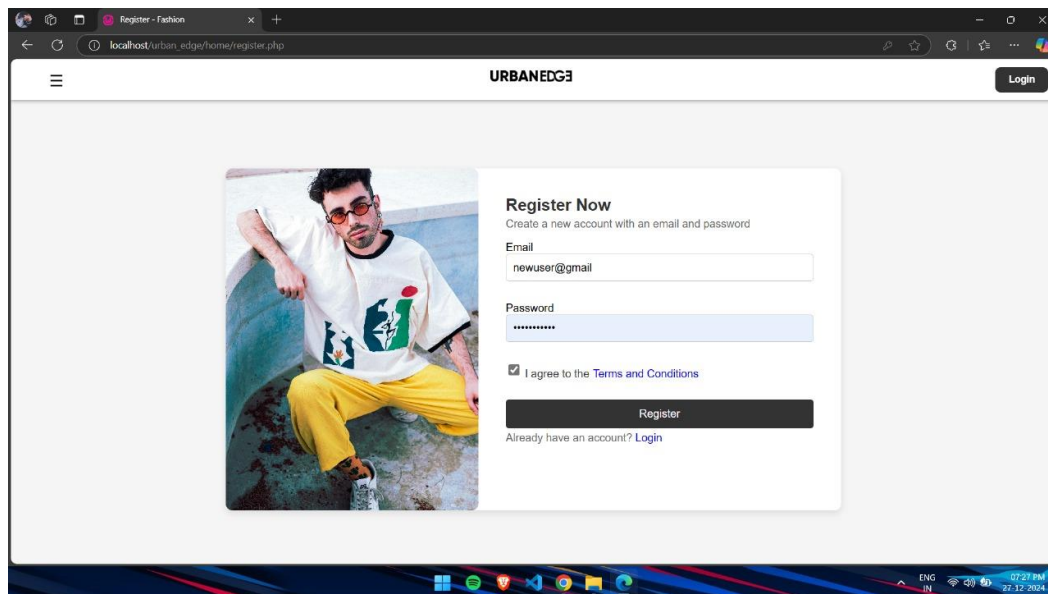
4.1 INPUT DESIGN

The input design is the process of converting the user- oriented inputs into the computer based format. The data is fed into the system using simple inactive forms. The forms have been supplied with messages so that the user can enter data without facing any difficulty. The data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into system. The goal of designing input data is to make the automation as easy and free from errors as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right messages and help for the user at right are also considered for development for this project. Input Design is a part of the overall design. The input methods can be broadly classified into batch and online. Internal controls must be established for monitoring the number of inputs and for ensuring that the data are valid. The basic steps involved in input design are:

- Review input requirements.
- Decide how the input data flow will be implemented.
- Decide the source document.
- Prototype on line input screens.
- Design the input screens.

The quality of the system input determines the quality of the system output. Input specifications describe the manner in which data enter the system for processing. Input design features can ensure the reliability of the system and produce results from accurate data. The input design also determines whether the user can interact efficiently with the system.

These is a sample input forms:

A screenshot of a web browser displaying the registration page for 'URBANEDGE'. The browser's address bar shows 'localhost/urban_edge/home/register.php'. The page features a navigation bar with a hamburger menu icon on the left, the 'URBANEDGE' logo in the center, and a 'Login' button on the right. The main content area is divided into two sections. On the left is a large image of a man with dark hair and sunglasses, wearing a white t-shirt with a graphic and bright yellow pants, sitting on a ledge. On the right is a white registration form titled 'Register Now' with the subtitle 'Create a new account with an email and password'. The form contains an 'Email' field with the placeholder 'newuser@gmail.com', a 'Password' field with masked characters, a checkbox labeled 'I agree to the Terms and Conditions', a dark grey 'Register' button, and a link 'Already have an account? Login'.

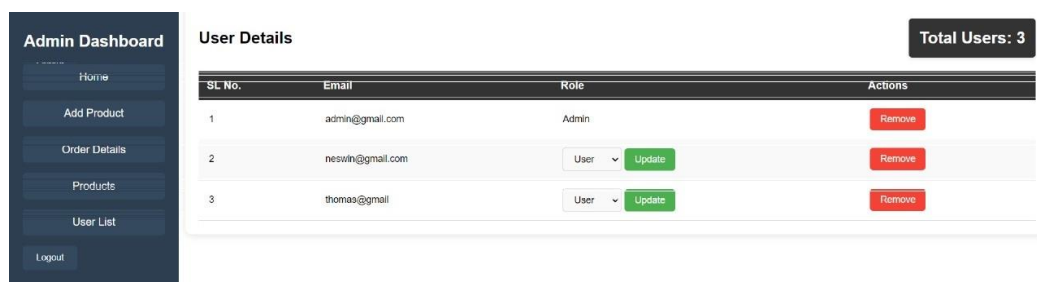
This input form is for creating a profile for new customers. It includes text fields for entering Email, Password. When a user login successfully, they will be redirected to the Products Page. This page will display a wide selection of t-shirts available for purchase, allowing customers to easily browse through different styles and categories. The user will have the option to view detailed information about each product, including images, descriptions, and prices.

The customer registration form is crucial for the project as it enables users to register before logging in. This process helps verify their credentials. Each customer must fill out all fields in the form to register successfully. Each field has a clear label indicating what information to enter, and placeholders guide users on the expected input. The form is neatly arranged, with labels and input boxes aligned for a clean appearance. Once completed, users can submit their information to be stored in the database. It's important that all entered data is accurate according to the specified field types.

4.2 OUTPUT DESIGN

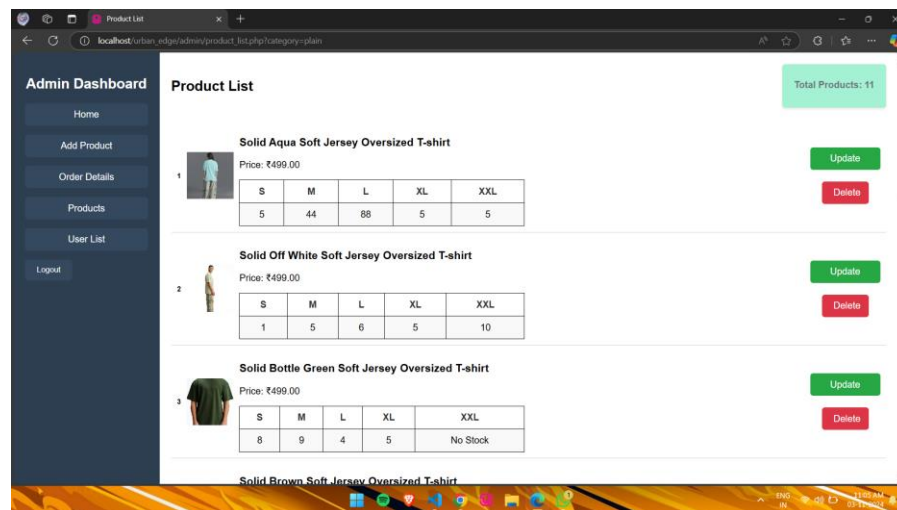
A quality output is one, which meets the requirements of end user and presents the information clearly. In any system result of processing are communicated to the user and to the other system through outputs. In the output design it is determined how the information is to be displayed for immediate need. It is the most important and direct source information is to the user. Efficient and intelligent output design improves the system's relationships with the user and helps in decision -making. The objective of the output design is to convey the information of all the past activities, current status and to emphasis important events. The output generally refers to the results and information that is generated from the system. Outputs from computers are required primarily to communicate the results of processing to the users. Output also provides a means of storage by copying the results for later reference in consultation. There is a chance that some of the end users will not actually operate the input data or information through workstations, but will see the output from the system.

In our project, outputs are generated as user management tables to display information about registered users and their details and managing products by updating and deleting product listing efficiently.



Admin Dashboard			
User Details			Total Users: 3
SL No.	Email	Role	Actions
1	admin@gmail.com	Admin	Remove
2	neswin@gmail.com	User Update	Remove
3	thomas@gmail	User Update	Remove

In our project, outputs are generated as user details tables to display information about registered users and their details. These tables provide insights into user data and allow administrators to easily view users ensuring efficient handling of user information and enhancing the overall user experience. This approach enables better tracking of user activity and helps maintain an organized database of all registered users.



The Product Management feature allows administrators to efficiently manage the inventory of t-shirts on the URBAN EDGE platform. This system enables admin to add new products, update existing product details such as descriptions, prices, and images, and delete products that are no longer available. With a user-friendly interface, administrators can ensure that the product listings are accurate and up-to-date, enhancing the overall shopping experience for customers.

4.3 DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the users. The general theme behind a database is to integrate all information. Database design is recognized as a standard of management information system and is available virtually for every computer system. In database design several specific objectives are considered:

- Ease of learning and use
- Controlled redundancy
- Data independence
- Accuracy and integrity

A database is an integrated collection of data and provides centralized access to the data. Usually, the centralized data managing the software is called RDBMS. The main significant difference between RDBMS and other DBMS is the separation of data as seen by the program and data has in direct access to stores device. This is the difference between logical and physical data.

4.3.1 NORMALIZATION

Designing a database is complete task and the normalization theory is a useful aid in the design process. The process of normalization is concerned with transformation of conceptual schema into computer representation form. There will be need for most databases to grow by adding new attributes and new relations. The data will be used in new ways. Tuples will be added and deleted. Information stored may undergo updating also. New association may also be added. In such situations the performance of a database is entirely depend upon its design. A bad database design may lead to certain undesirable things like:

- Repetition of information
- Inability to represent certain information
- Loss of information

To minimize these anomalies, Normalization may be used. If the database is in a normalized form, the data can be growing without, in most cases, forcing the rewriting application programs. This is important because of the excessive and growing cost of maintaining an organization's application programs and its data from the disrupting effects of database growth. As the quality of application programs increases, the cost of maintaining the without normalization will rise to prohibitive levels. A normalized database can also encompass many related activities of an organization thereby minimizing the need for rewriting the applications of programs. Thus, normalization helps one attain a good database design and there by ensures continued efficiency of database.

Normalization theory is built around the concept of normal forms. A relation is said to be in normal form if it satisfies a certain specified set of constraints. For example, a relation is said to be in first normal form (1NF) if it satisfies the constraint that it contains atomic values only. Thus, every normalized relation is in 1NF. Numerous normal forms have been defined. Codd defined the first three normal forms.

All normalized relations are in 1NF, some 1NF relations are also in 2NF and some 2NF relations are also in 3NF. 2NF relations are more desirable than 1NF and 3NF are more desirable than 2NF. That is, the database designer should prefer 3NF than 1NF or 2NF. Normalization procedure states that a relation that is in some given normal form can be converted into a set of relations in a more desirable form. We

can define this procedure as the successive reduction of a given collection of relations to some more desirable form. This procedure is reversible. That is, it is always possible to take the output from the procedure and convert them back into input. In this process, no information is lost. So, it is also called “no loss decomposition”.

First Normal Form

A relation is in first normal form (1NF) if and all its attributes are based on single domain. The objective of normalizing a table is to remove its repeating groups and ensure that all entries of the resulting table have at most single value.

Second Normal Form

A table is said to be second Normal Form (2NF), when it is in 1NF and every attribute in record is functionally dependent upon the whole key, and not just a part of the key.

Third Normal Form

A table is in third Normal Form (3NF), when it is in 2NF and every non- key attribute is functionally dependent on just the primary key.

RULES FOR NORMALIZATION

Rule 1: There should be a one-to-one relationship between the instances of an entity and the rows of the table.

Rule 2: A field should have the same meaning in each row of the table.

Rule 3: Each table should represent at most one entity.

Rule 4: Multiple instances of an entity should be represented by multiple rows in a table.

Rule 5: Joins should be based only on primary and foreign-key equality.

Rule 6: Make sure keys are linked correctly.

4.4 TABLES

Table is a collection of complete details about a particular subject. These data are saved in rows and Columns. The data of each Row are different units. Hence, rows are called RECORDS and Columns of each row are called FIELDS. Data is stored in tables, which is available in the backend the items and data, which are entered in the input, form id directly stored in this table using linking of database. We can link more than one table to input forms. We can collect the details from the different tables to display on the output.

1. Table name: user

Field Name	Data Type	Constraint	Description
id	int	Primary key	Unique id
email	Varchar(30)	Not null	Email id
password	Varchar(30)	Not null	Hashed password
role	Enum	Default	User roll

2. Table name: order_details

Field Name	Data Type	Constraint	Description
order_detail_id	Int	Primary key	Unique id
order_id	Int	Foreign key	Order id
user_id	Int	Foreign key	User id
address_id	Int	Foreign key	Address id
total_amount	Decimal(10,2)	Not Null	Total amount
order_status	Varchar(20)	Not Null	Status of order
order_date	timestamp	Not Null	Ordered date

3. Table name: addresses

Field Name	Data Type	Constraint	Description
address_id	int	Primary key	Unique identifier
user_id	int	Foreign key	User id
address_name	Varchar (50)	Not null	Address label
address_line1	Varchar (50)	Not null	Primary address
address_line2	Varchar (50)	Not null	Optional
city	Varchar (50)	Null	User city
pincode	Varchar (20)	Not null	Pincode of user
phone_number	Varchar (15)	Not null	Phone
address_type	Enum	Not null	Specifies categories
street	Varchar (150)	Not Null	street

4. Table name: order_items

Field Name	Data Type	Constraint	Description
order_item_id	Int	Primary key	Unique id
order_detail_id	Int	Foreign key	Order detail id
product_id	Int	Foreign key	Product id
quantity	Int	Not null	Quantity
price	Decimal(10,2)	Not null	Total price
size	Varchar(50)	Null	size

5. Table name: products

Field Name	Data Type	Constraint	Description
product_id	int	Primary key	Unique id
title	Varchar(50)	Not Null	Product title
price	Decimal(10,2)	Not Null	Product price
description	Text	Not Null	Description
category_id	Int	Foreign Key	Category id
stock_s	Int	Default	Stock of size s
stock_m	Int	Default	Stock of size m
stock_l	Int	Default	Stock of size l
stock_xl	Int	Default	Stock of size xl
stock_xxl	Int	Default	Stock of size xxl
image_path	Varchar(255)	Not Null	Product image
created_at	timestamp	Default	Created at

6. Table name: payments

Field Name	Data Type	Constraint	Description
payment_id	Int	Primary key	Unique id
order_id	Int	Foreign key	Order id
payment_date	Datetime	Default	Date of payment
amount	Decimal(10,2)	Not null	Total amount
payment_type	Varchar(50)	Not null	Payment type

7. Table name: site_statistics

Field Name	Data Type	Constraint	Description
id	int	Primary key	Unique id
orders_placed	Int	Default	Total orders
total_customers	Int	Default	Total customers
total_revenue	Int	Default	Total revenue
last_revenue	Decimal(10,2)	Default	Last revenue
last_updated	timestamp	Default	Last update

8. Table name: categories

Field Name	Data Type	Constraint	Description
category_id	int	Primary key	Unique id
category_name	Varchar (50)	Not Null	Name of the category

9.Table name: product_images

Field Name	Data Type	Constraint	Description
image_id	Int	Primary key	Unique id
product_id	Int	Foreign key	Product id
image_path	Varchar(255)	Not Null	Path of image

10.Table name: product_sizes

Field Name	Data Type	Constraint	Description
product_size_id	Int	Primary key	Unique id
product_id	Int	Foreign key	Product id
size	Varchar(50)	Not Null	Size
size_s	Int	Default	Small size
size_m	Int	Default	Medium size
size_l	Int	Default	Large size
size_xl	Int	Default	Xl size
size_xxl	Int	Default	Xxl size

4.5 DATA FLOW DIAGRAM

Data Flow Diagram is a network that describes the flow of data and processes that change, or transform, data throughout the system. This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs. There are various symbols used in a DFD. Bubbles represent the processes. Named arrows indicate the data flow. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. Each component in a DFD

is labelled with a descriptive name. Process names are further identified with a number.

The Data Flow Diagram shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the input (source), outputs (destination), database (files) and procedures (data flow), all in a format that meet the user's requirements.

The main merit of DFD is that it can provide an overview of system requirements, what data a system would process, what transformations of data are done, what files are used, and where the results flow.

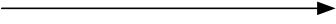
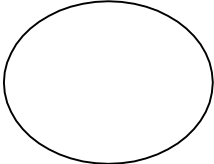


This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

Rules for constructing a Data Flow Diagram

- Arrows should not cross each other
- Squares, circles and files must bear names.
- Decomposed data flow squares and circles can have same time
- Choose meaningful names for data flow
- Draw all data flows around the outside of the diagram

4.5.1 Basic Data Flow Diagram Symbols

Table Name: Data Flow Diagram Symbols

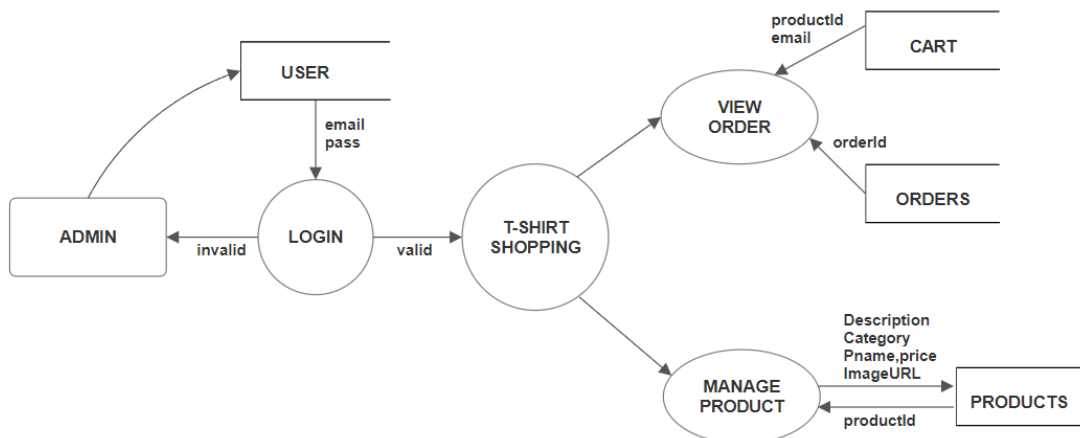
	<p>A data flow is a route, which enables packets of data to travel from one point to another. Data may flow from a source to a process and from data store or process. An arrow line depicts the flow, with arrow head pointing in the direction of the flow.</p>
	<p>Circles stands for process that converts data in to information. A process represents transformation where incoming data flows are changed into outgoing data flows.</p>
	<p>A data store is a repository of data that is to be stored for use by a one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names. If a process merely uses the content of store and does not alter it, the arrowhead goes only from the store to the process. If a process alters the details in the store, then a double-headed arrow is used.</p>
	<p>A source or sink is a person or part of an organization, which enters or receives information from the system, but is considered to be outside the context of data flow model</p>

4.5.2 DATA FLOW DIAGRAM

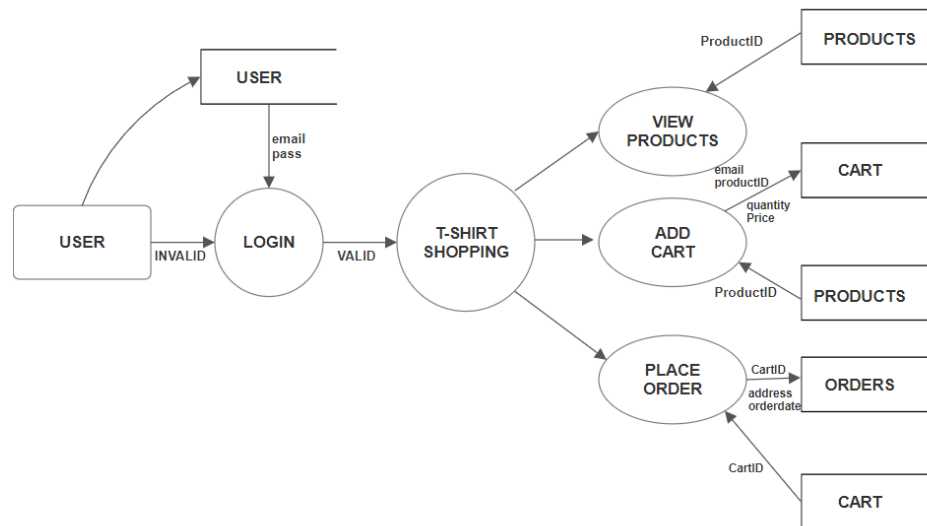
Each component in a DFD is labelled with a descriptive name. Process name is further identified with number. Context level DFD is draw first. Then the process is decomposed into several elementary levels and is represented in the order of importance. A DFD describes what data flow (logical) rather than how they are processed, so it does not depend on hardware, software, and data structure or file organization.



Zeroth level DFD for Online Shopping System



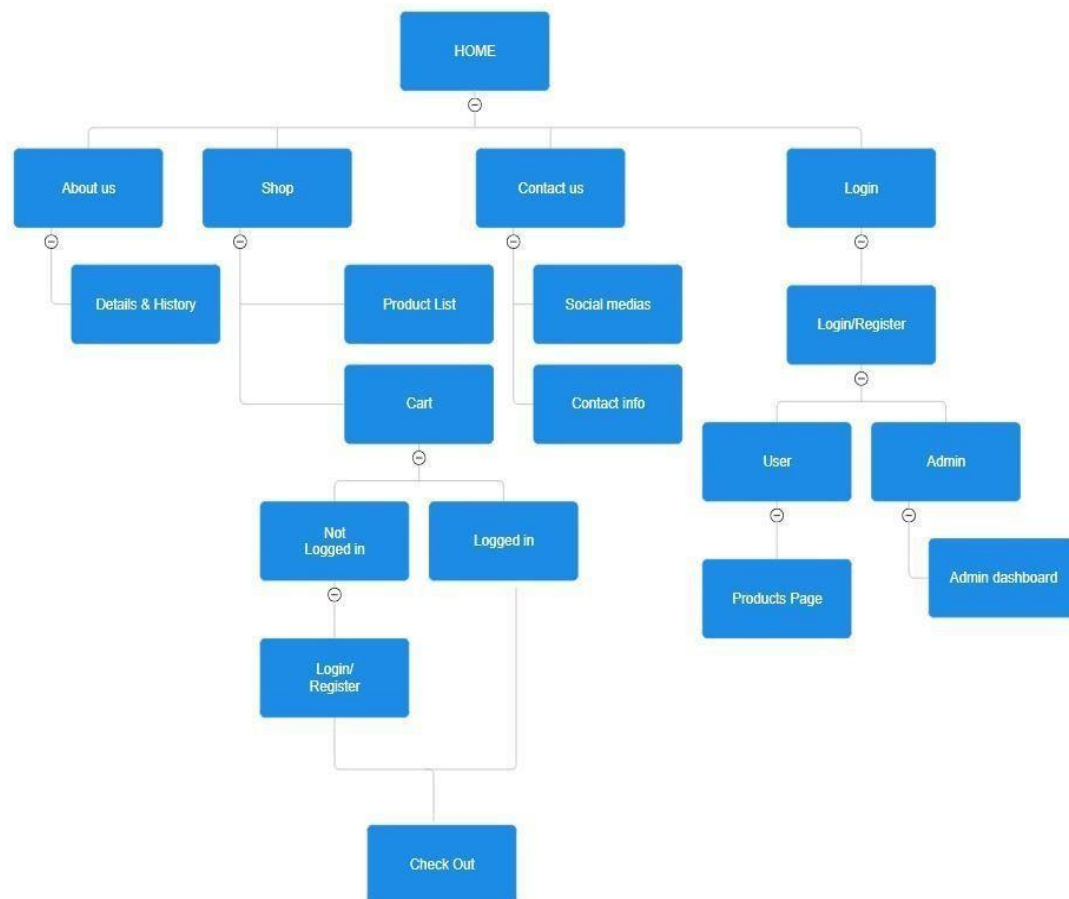
First Level DFD for Admin



First Level DFD for Customer

4.6 SITE MAP DIAGRAM

A website sitemap is a structured outline that shows the relationships between different pages, helping users and search engines navigate and understand the layout of the site effectively. It provides a clear, hierarchical view of main sections and subsections, making it easier to locate specific information. Sitemaps are valuable for both user experience and SEO, as they ensure content is organized logically and can be easily accessed, indexed, and optimized for search engines.



SITE MAP DIAGRAM

SYSTEM TESTING

5 SYSTEM TESTING

5.1 TESTING PROCEDURES

Software testing is essential for the successful development of the URBAN EDGE e-commerce platform. This phase ensures the quality and reliability of the website by verifying that all features meet specifications and function as expected, providing users with a seamless shopping experience. Testing helps identify and fix any issues in the platform, ensuring it performs smoothly in real- world usage.

Testing begins at the component level, focusing on each feature, and progresses to full platform integration. Well-designed test cases are crucial for spotting potential errors, confirming that the e-commerce site meets both customer expectations and performance standards. Detailed test plans and relevant test data are used to validate all functionalities, making testing a critical step in launching an efficient, user-friendly shopping experience on URBAN EDGE.

5.2 TESTING METHODOLOGIES

The URBAN EDGE platform undergoes various testing methodologies to assess critical aspects like response time, functionality, and security before it is ready for user acceptance testing. The following testing types are used during this process:

UNIT TESTING

Unit testing focuses on individual components of the URBAN EDGE platform, typically examining small sections of code independently. Each unit receives specific inputs, and the outputs are compared to the expected results. This process ensures that each function operates correctly in isolation, allowing for the early detection of design issues. For URBAN EDGE, unit tests validate core functionalities like user login, product display, secure checkout, and payment processing. These tests not only improve code quality but also help catch minor errors that could impact the user experience, such as product details not displaying correctly or errors in adding items to the cart. Unit testing also enhances maintainability, as it allows developers to identify and correct errors quickly, keeping the platform reliable and responsive. Furthermore, by running tests consistently, developers can track improvements over time, making it easier to add new features while preserving the stability of existing functions.

INTEGRATION TESTING

After successful unit testing, integration testing is performed to evaluate how combined modules of the URBAN EDGE platform interact with each other. This phase is essential for verifying that individual components, such as user accounts, product listings, and checkout processes, work together smoothly. Integration testing follows predefined test plans to confirm that critical functions like adding items to the cart and processing payments operate as expected when integrated. This testing phase helps identify any issues that may emerge from the interactions between different modules, such as data inconsistencies or unexpected behaviors. By ensuring all parts of the platform work harmoniously, integration testing supports the performance, reliability, and user experience goals of the URBAN EDGE site, reducing the risk of errors in the live environment. This approach also makes it easier to pinpoint and resolve any issues that arise, contributing to a seamless and cohesive user experience.

VALIDATION TESTING

Validation testing for the URBAN EDGE website focuses on ensuring that user inputs, such as email addresses, passwords, and contact numbers, meet specific criteria before being processed. This testing checks that email formats are correct, ensuring they contain the necessary components (e.g., an "@" symbol and a valid domain) to prevent invalid registrations. Password validation enforces strength requirements, such as a minimum length and a mix of characters, to enhance security. Additionally, the contact number input is validated to ensure it matches the expected format for phone numbers, reducing errors in communication. By implementing robust validation testing, URBAN EDGE enhances user experience, improves data integrity, and protects against potential security vulnerabilities.

SYSTEM TESTING

Following the integration phase, system testing evaluates the entire URBAN EDGE platform against specified requirements. This black-box testing technique assesses the website's functionality from an end-to-end perspective, ensuring all components work together seamlessly. Independent testing teams conduct comprehensive evaluations of both functional aspects such as product searches and order processing—and non-functional aspects like performance and security. System testing verifies compliance with industry standards, checks data accuracy, and assesses user interface responsiveness. This thorough evaluation

ensures a reliable and enjoyable shopping experience for customers and helps identify any issues before the platform goes live.

USER ACCEPTANCE TESTING

Finally, acceptance testing is conducted to validate the functionality of the URBAN EDGE platform using real-world data and scenarios. This phase is crucial for demonstrating the system's effectiveness to stakeholders and ensuring it meets their operational needs. Acceptance testing involves end-users and administrators testing the platform to confirm that it fulfils all specified requirements and is ready for deployment in a live environment.

In conclusion, thorough testing across all levels unit, integration, system, and acceptance is vital for the successful implementation of URBAN EDGE. This structured approach to testing ensures the platform's reliability and efficiency while confirming its readiness to meet the diverse demands of customers in the e-commerce sector.

IMPLEMENTATION
AND
MAINTENANCE

6 IMPLEMENTATION AND MAINTENANCE

SYSTEM IMPLEMENTATION

The implementation phase is crucial for transitioning the URBAN EDGE platform from theoretical design to a functional e-commerce application. This stage involves deploying the website in a live environment, where its performance is closely monitored, and results are assessed. Effective implementation requires careful planning, an analysis of existing processes, and an understanding of any limitations.

Key activities during implementation include migrating existing product data into a format compatible with the new system, training users (such as administrators and customer service representatives) to utilize the platform effectively, and verifying the accuracy of reports generated by the website. The primary goal is to ensure that the tested platform is fully operational while minimizing costs, risks, and disruptions for users. This involves preparing digital files, setting up necessary hardware, and providing comprehensive training to the operational staff.

The transition to the new e-commerce platform is managed with clear strategies, including establishing timelines, ensuring data accuracy, and preparing support materials for users. A successful implementation will not only facilitate a smooth transition but also enhance the overall shopping experience by effectively integrating the system into daily operation.

SYSTEM MAINTENANCE

Maintenance activities for the URBAN EDGE platform begin immediately after the implementation phase, ensuring the website operates efficiently and adapts to evolving user needs. Maintenance is managed using the same planning and control methodologies applied during the project and can be classified into three categories: corrective, adaptive, and perceptive.

Corrective Maintenance: This involves addressing any performance failures or issues that arise after the website goes live. It includes fixing bugs, repairing processes, and making adjustments to resolve previously unrecognized problems.

Adaptive Maintenance: This type focuses on modifying the platform's functions to accommodate changes in market demands or advancements in technology. For example, integrating new features such as enhanced product filtering or improved payment options would fall under adaptive maintenance.

Perceptive Maintenance: This encompasses enhancements to the website's performance, ensuring it meets users' changing needs over time. This may involve upgrading the user interface, optimizing database queries for faster load times, or implementing user feedback to improve usability.

The implementation view of software requirements serves as a practical guide for maintaining the URBAN EDGE platform. This includes recognizing constraints imposed by predefined system elements and ensuring that the platform remains responsive to user needs. By prioritizing maintenance, can evolve alongside technological advancements and shopping trends, thus sustaining its effectiveness and relevance in the competitive e-commerce landscape.

FUTURE SCOPE OF THE PROJECT

7 FUTURE SCOPE OF THE PROJECT

Future expansions of the project depend upon automatically generating parts for purchase and sales, online purchasing facility for customers. Shops could implement the URBAN EDGE, as it is ample and user friendly. Making enhancements is all about perfective maintenance. It means adding, modifying or redeveloping the code to support changes in the specifications. It is necessary to keep up with changing user needs and the operational environment. More money and time are spent on perfective maintenance than on corrective or adaptive maintenance together. Some features of the program are listed below:

- User Feedback Integration: A proactive approach that demonstrates commitment to customer needs; ensures continuous improvement.

CONCLUSION

8 CONCLUSION

The **URBAN EDGE** project has successfully met its defined objectives, with a trial run yielding highly satisfactory results. The system streamlines processes, offering straightforward, consistent, and efficient workflows for users and administrators alike. By addressing the limitations of the previous manual process, the new system provides faster services, with features such as easy report generation being particularly advantageous for users.

Designed with flexibility in mind, the system can adapt to changes in the business environment, ensuring it remains relevant. However, to maintain optimal performance and accurate reporting, users must prioritize data accuracy and completeness in their entries. While the system boasts numerous benefits, it's important to acknowledge potential challenges; ongoing adjustments may be necessary to enhance functionality and tackle unforeseen issues.

In conclusion, the **URBAN EDGE** system represents a significant advancement over the manual process. Users are encouraged to embrace data accuracy and remain open to ongoing improvements to fully realize the system's potential benefits.

BIBLIOGRAPHY

9 BIBLIOGRAPHY

BOOKS REFERRED:

- ❖ Ramus Lerdorf and Kevin Tatore, “Programming PHP”
- ❖ Robin Nixon,” Learning PHP, MySQL, and JavaScript”
- ❖ Elias M. Awed, “System Analysis and Design”
- ❖ K K Aggarwal, “Software Engineering 3rd Edition”
- ❖ C. J. Date, “An Introduction to Database System”
- ❖ James Martin, “Principles of Database Management System”

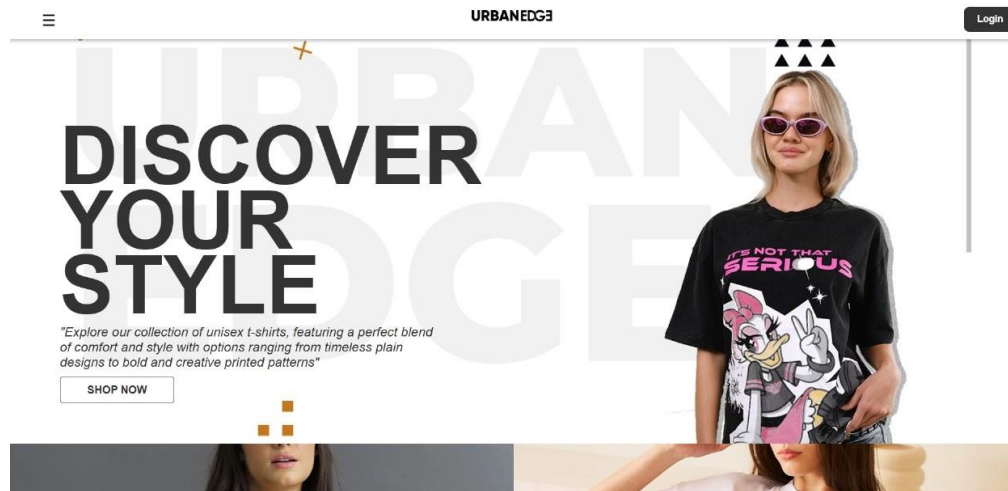
WEBSITES REFERRED:

- ❖ <https://www.stackoverflow.com>
- ❖ <https://www.sqlsolutions.com>
- ❖ https://www.tutorialspoint.com/system_analysis_and_design.html
- ❖ <http://softwaretestingfundamentals.com/test-case/>
- ❖ <https://www.geeksforgeeks.org/php-coding-standards/>
- ❖ <http://www.wampserver.com/en/>
- ❖ https://en.wikipedia.org/wiki/Microsoft_Word
- ❖ <https://en.wikipedia.org/wiki/SmartDraw>
- ❖ <https://www.w3schools.com/>

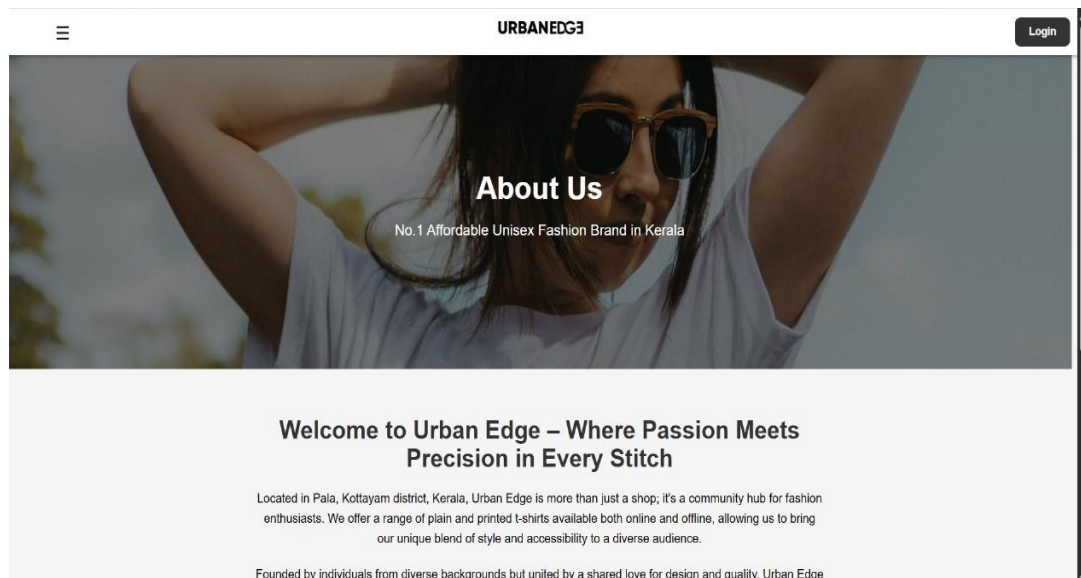
APPENDIX

10 SCREENSHOTS

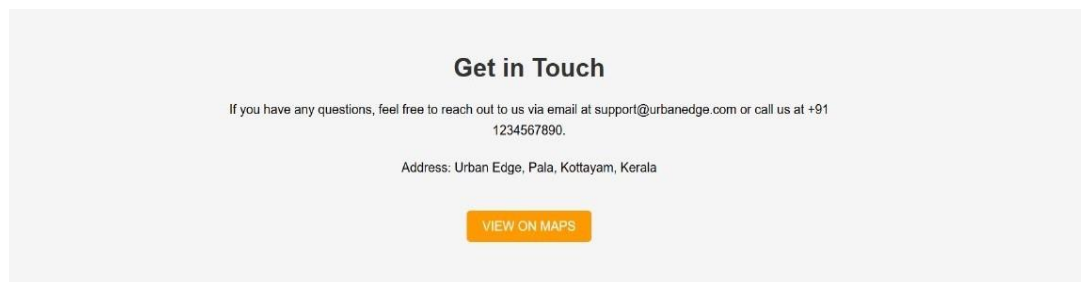
1.HOME PAGE




2.ABOUT US



3.CONTACT US

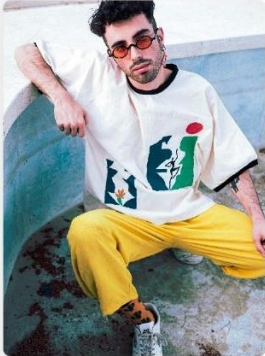


4.SIGNUP PAGE



URBANEDGE3

Login



Register Now

Create a new account with an email and password

Email
newuser@gmail


Password

☒ I agree to the [Terms and Conditions](#)

Register


Already have an account? [Login](#)

5.SIGNIN PAGE



URBANEDGE3

Login



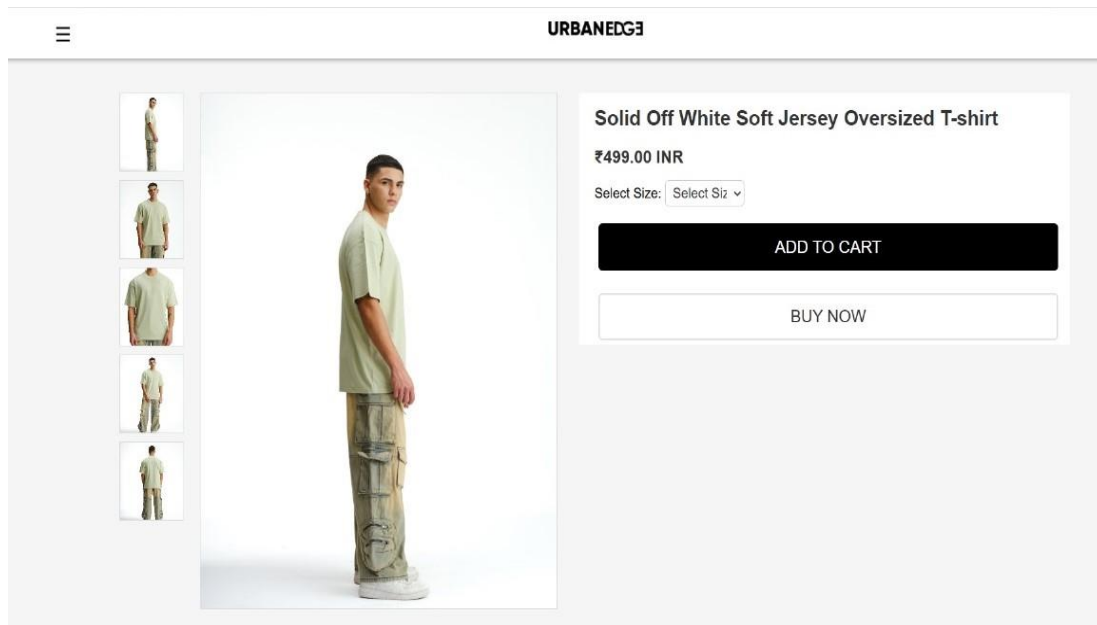
Log In to Your Account

urbanedge123@gmail.com

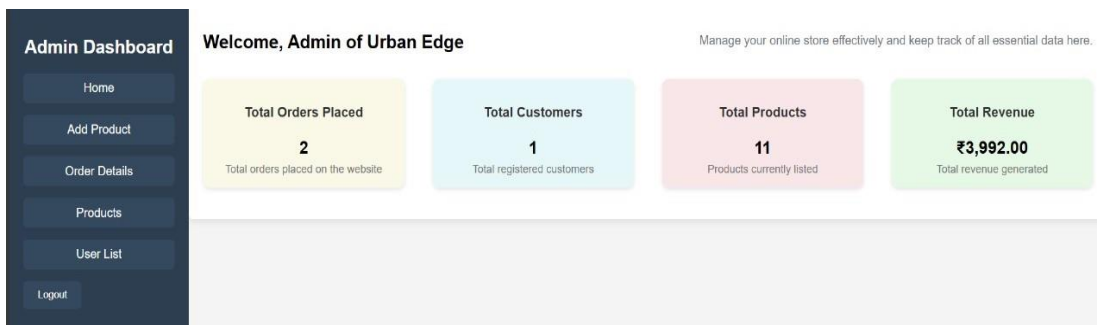
Login

New to here? [Register](#)

6.PRODUCTS IMAGE



7.ADMIN DASHBOARD



8.ORDER MANAGEMENT

Order Details

Placed Orders

Shipped Orders

Delivered Orders

Canceled Orders

SL. NO.	CUSTOMER EMAIL	TOTAL AMOUNT	DATE ORDERED	STATUS	VIEW DETAILS
1	thomas@gmail	499.00	2024-12-25 18:02:31	placed	<button>View</button>

8.ADD PRODUCT

Add Product

Upload Product Images (up to 5)

No file chosen

Product Name

Product Description

Enter Available Quantity for Each Size

S	M	L	XL	XXL
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

Price

Product Category


▼

9.PRODUCT MANAGEMENT

Product List

Total Products: 11

1



Solid Aqua Soft Jersey Oversized T-shirt


Price: ₹499.00

S	M	L	XL	XXL
5	44	88	5	5

Update

Delete

2



Solid Off White Soft Jersey Oversized T-shirt


Price: ₹499.00

S	M	L	XL	XXL
No Stock	4	6	5	10

Update

Delete

3



Solid Bottle Green Soft Jersey Oversized T-shirt

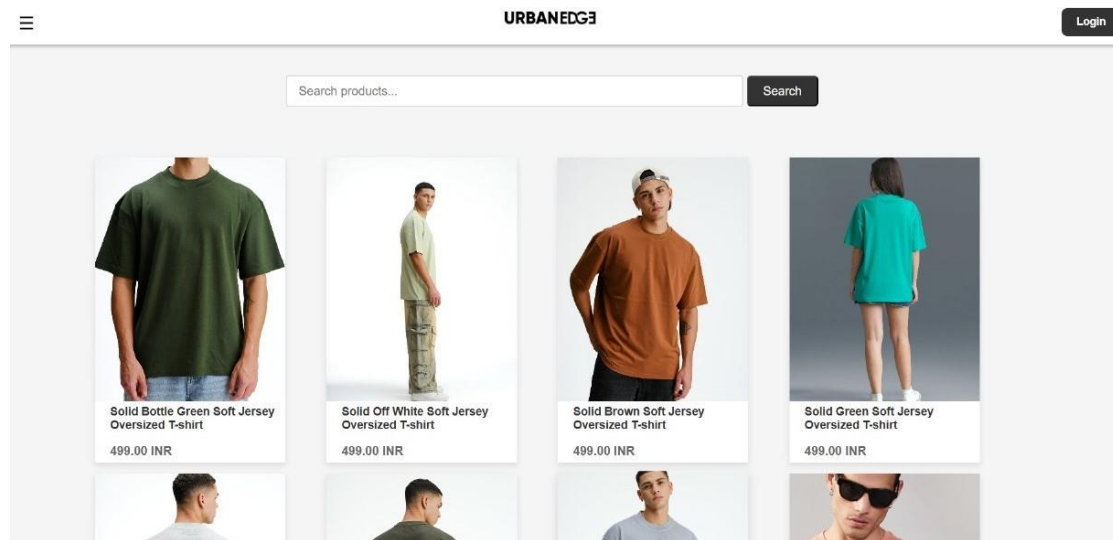
Price: ₹499.00

S	M	L	XL	XXL
8	9	4	5	No Stock

Update

Delete

10.PRODUCT PAGE



11.USER MANAGEMENT

Admin Dashboard

- Home
- Add Product
- Order Details
- Products
- User List
- Logout

User Details

Total Users: 3

SL No.	Email	Role	Actions
1	admin@gmail.com	Admin	<button>Remove</button>
2	neswin@gmail.com	User <button>Update</button>	<button>Remove</button>
3	thomas@gmail	User <button>Update</button>	<button>Remove</button>