TRENDIFY- A CLOTHING WEBSITE

A Project Report for Industrial Internship

Submitted by

Neelotpal Sahoo

In the partial fulfillment for the award of the degree of

Btech

in the

Department of Computer Science And Engineering

Of

Heritage Institute Of Technology

at

Ardent Computech Pvt. Ltd.

Module

ARDENT
COMPUTECH PVT. LTD.
High-End Technology Training and Project
ARDENT®
GROUP OF COMPANIES
INTERNSHIP | DEVELOPMENT | PROJECT
An ISO 9001 : 2015 Certified Company

CERTIFICATE FROM SUPERVISOR

This is to certify that **"Richik Banerjee, Sadhana Kumari Mondal, Anindya Sarmadhikari, Shayeri Nandy** of **2021-22"** have successfully completed the project titled "TRENDIFY" under my supervision during the period from "17-06-2024" to "10-07-2024" which is in partial fulfilment of requirements for the award of the **Btech** degree and submitted to the Department of **"Computer Science And Engineering "** of **"Heritage Institute Of Technology".**

_____

Signature of the Supervisor

Starting date: 17/06/2024
Ending date :10/07/2024

Name of the Project :TRENDIFY

Supervisor:Subhojit Santra

ACKNOWLEDGEMENT

The achievement that is associated with the successful completion of any task would be incomplete without mentioning the names of those people whose endless cooperation made it possible. Their constant guidance and encouragement made all our efforts successful.

We take this opportunity to express our deep gratitude towards our project mentor/external guide ***[Mr. Subhojit Santra]*** and internal guide ***[Mr. Srimanta Santra]*** for giving such valuable suggestions, guidance and encouragement during the development of this project work.

Last but not the least we are grateful to all the faculty members of **Ardent Computech Pvt. Ltd.** for their support.

DECLARATION

This is to certify that the dissertation/project proposal entitled "**TRENDIFY**" is done by us, is an authentic work carried out for the partial fulfilment of the requirements for the award of the certificate of Bachelor of Technology under the guidance of Mr. Subhojit Santra. The matter embodied in this project work has not been submitted earlier for award of any certificate to the best of our knowledge and belief.

Name of the Student

- Richik Banerjee
- Sadhana Kumari Mondal
- Anindya Sarmadhikari
- Shayeri Nandy

Signature of the students

I.

II.

III.

IV.

APPROVAL SHEET

This project report entitled '**TRENDIFY**' by Richik Banerjee, Sadhana Kumari Mondal, Anindya Sarmadhikari, Shayeri Nandy is approved for the degree of Bachelor of Technology

_____

(Signature of Examiners)

Signature of Internal Guide

_____

Signature of Principal
(Sudipta Chakrabarti)

_____

Signature of Industrial Guide
(Mr. Subhojit Santra)

Date: 10/07/2024 Place:

Ardent Computech Pvt

Ltd.

Certificate by Guide

This is to certify that this project entitled "**TRENDIFY** "submitted in partial fulfilment of the certificate of Bachelor of Technology through Ardent Computech Pvt Ltd, done by the

Group Members:

- Richik Banerjee
- Sadhana Kumari Mondal
- Anindya Sharmadhikari
- Shayeri Nandy

is an authentic work carried out under my guidance & best of our knowledge and belief.

Signature of the students

Date:

Signature of the Guide

Date:

Certificate of Approval

This is to certify that this proposal of Minor project, entitled **" TRENDIFY"** is a record of bona-fide work, carried out by: **1. Richik Banerjee-211260100110148, 2. Sadhana Kumari Mondal-221260120981, 3. Anindya Sarmadhikari-221260120958, 4. Shayeri Nandy-211260100110169** under my supervision and guidance through the Ardent Computech Pvt Ltd. In my opinion, the report in its present form is in partial fulfilment of all the requirements, as specified by the Heritage Institute Of Technology as per regulations of the **Ardent®**. In fact, it has attained the standard, necessary for submission. To the best of my knowledge, the results embodied in this report, are original in nature and worthy of incorporation in the present version of the report for Bachelor of Technology.

Guide/Supervisor

Mr. Subhojit Santra

Subject Matter Expert & Technical Head (Php)

Ardent Computech Pvt Ltd (An ISO 9001:2008 Certified)

Module #132, Ground Floor, SDF Building, GP Block, Sector V, Bidhannagar, Kolkata, West Bengal 700091

External Examiner(s)

Head of the Department

Department of Bachelor of Technology, Heritage Institute Of Technology,

Project Responsibility Form

SERIAL NO.	MEMBER OF PROJECT	RESPONSIBILITY
1.	RICHIK BANERJEE	WEB PAGE DESIGNING & BACKEND CODING
2.	SADHANA KUMARI MONDAL	POWER POINT PRESENTATION
3.	ANINDYA SARMADHIKARI	DATA COLLECTION
4.	SHAYERI NANDY	DOCUMENTATION

TRENDIFY

(A CLOTHING WEBSITE)

TABLE OF CONTENTS

S.NO.	Name of the Topic	Page No.
1.	**Company Profile**	**14**
2.	**Introduction**	**15**
2.a	Objective	16
2.b	Scope	17
3	**System analysis**	**18**
3.a	Identification of need	19
3.b	Feasibility study	20
3.c	Work flow	21-25
3.d	Study of the system	26
3.e	Input and Output	27
3.f	Software requirement specification	28-29
3.g	Software engineering paradigm applied	30-32
4.	**System Design**	**33**
4.a	Data Flow Diagram	34-39
4.b	Entity Relationship Diagram	41-44
5	**Output Screen**	**54**
5.a	User interface Design	55
5.b	Snapshots	56-64
6	**Implementation and Testing**	**65**
6.a	Introduction	65
6.b	Objectives of testing	65-66

S.NO.	Name of the Topic	Page No.
6.c	Test Cases	67
6.d	White Box Testing	70
6.e	Black Box Testing	70
6.f	Output Testing	70
6.g	Goal of Testing	73
6.h	Integration Test Report	74
7	**Gantt Chart**	**76**
7.a	Project Schedule	77
7.b	Gantt Chart	78
8	**System Security Measures**	79
8.a	Database security measures	80
8.b	System security measures	80
8.c	Limitations	81
9	**Conclusion**	**82**
9	**Future Scope & Further Enhancements**	**83**
10	**Bibliography/References**	**84**

1. ARDENT COMPUTECH PVT.LTD.

Ardent Computech Private Limited is an ISO 9001-2008 certified Software Development Company in India. It has been operating independently since 2003. It was recently merged with ARDENT TECHNOLOGIES.

Ardent Technologies

ARDENT TECHNOLOGIES is a Company successfully providing its services currently in UK, USA, Canada and India. The core line of activity at ARDENT TECHNOLOGIES is to develop customized application software covering the entire responsibility of performing the initial system study, design, development, implementation and training. It also deals with consultancy services and Electronic Security systems. Its primary clientele includes educational institutes, entertainment industries, resorts, theme parks, service industry, telecom operators, media and other business houses working in various capacities.

Ardent Collaborations

ARDENT COLLABORATIONS, the Research Training and Development Department of ARDENT COMPUTECH PVT LTD is a professional training Company offering IT enabled services & industrial trainings for B-Tech, MCA, BCA, MSc and MBA fresher's and experienced developers/programmers in various platforms. Summer Training / Winter Training / Industrial training will be provided for the students of B. TECH, M. TECH, MBA and MCA only. Deserving candidates may be awarded stipends, scholarships and other benefits, depending on their performance and recommendations of the mentors.

Associations

Ardent is an ISO 9001:2008 company.

It is affiliated to National Council of Vocational Training (NCVT), Directorate General of Employment & Training (DGET), Ministry of Labor & Employment, and Government of India.

Abstract

The drivers for electronic commerce are both technological (under the tremendous pressure of innovation) and business oriented. This paper will highlight some guidelines for companies who are entering into TRENDIFY to create an TRENDIFY strategy or who already have an TRENDIFY presence to revise their existing strategy.

TRENDIFY is now seen as a reality for many businesses and a normal part of a business plan. The immediate benefits, in terms of cost savings, efficiencies and enhanced profitability are clear at every stage in the supply chain. Adopting e-business is no longer a competitive advantage, but a normal business process, without which an enterprise is unlikely to survive in the New economy [1]. Year 2000 saw many Dot-com companies built up and many companies going into TRENDIFY however now it is a different story, more and more companies are failing, and investors are becoming cautious to invest money into Internet ventures. There is more cash needed then was expected. Some of them had to get on the bandwagon as everybody else were and didn't want to be left behind, and now that the bubble has burst they are facing the consequences. This paper will aim to help companies to realize what they need to follow to have a successful business if they plan well and have a good strategy.

2. INTRODUCTION

TRENDIFY, or electronic commerce, refers to the buying and selling of goods and services over the internet. It involves the use of electronic platforms, such as websites, mobile applications, and social media, to conduct transactions between businesses and consumers or between businesses. TRENDIFY has revolutionized the way businesses operate and has created new opportunities for entrepreneurs and consumers alike.

One of the primary benefits of TRENDIFY is that it offers a more convenient and efficient way to conduct business. Consumers can browse and purchase products from the comfort of their own homes, while businesses can reach a global audience without the need for a physical storefront. TRENDIFY has also enabled businesses to streamline their operations by automating many of the tasks involved in the sales process, such as inventory management, order processing, and shipping.

2.a OBJECTIVE

TRENDIFY drives profitable growth by expanding customer reach, reducing cost-to serve, and creating differentiated customer experiences.

Utilizing this powerful tool wisely has become eminently important for business-to-business (B2B) companies. Looking at the current B2B environment, TRENDIFY has proven to be a disruptive force. More and more B2B companies are moving a significant portion of the buying journey online, and these buyers are increasingly expecting a similar TRENDIFY experience as consumers.

In order to capture this customer need, companies should first raise the bar on their expectations from their digital channel, repositioning it from merely a transactional sales channel to an engine of growth and setting goals accordingly.

2.a SCOPE

The potential for **TRENDIFY** development is enormous. Now a days one can buy products online through some sites like Flipkart and Amazon. In the age of TRENDIFY everything from gym equipment to laptops are available online. TRENDIFY is a super set of business cases. It includes E-trading, E-Franchising, E-Mailing, E- Engineering etc. Scope of TRENDIFY can be enumerated as follows:

1. Exchange of digitized information
2. Technology-enabled
3. Customers retention
4. Accounting
5. Supplier integration
6. Support the exchange

3. SYSTEM ANALYSIS

3. a IDENTIFICATION OF NEED

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem-solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system studies the minutest detail and gets analysed. The system analysist plays the role of the interrogator and dwells deep into the working of the present system. The System is viewed as a whole and the input to the system are identified. The outputs from the organization are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and Decisional variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem area are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solution is given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with personal.

3.b FEASIBILITY STUDY

Feasibility study is made to see if the project on completion will serve the purpose the organization for the amount of work.

Effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A Feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various area that were considered very carefully during the feasibility study of this project such as Technical, Economic and operational feasibilities.

3.c WORK FLOW

This Document plays a vital role in the development life cycle (SDLC) as it describes the complete requirement of the system. It is meant for use by the developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process.

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

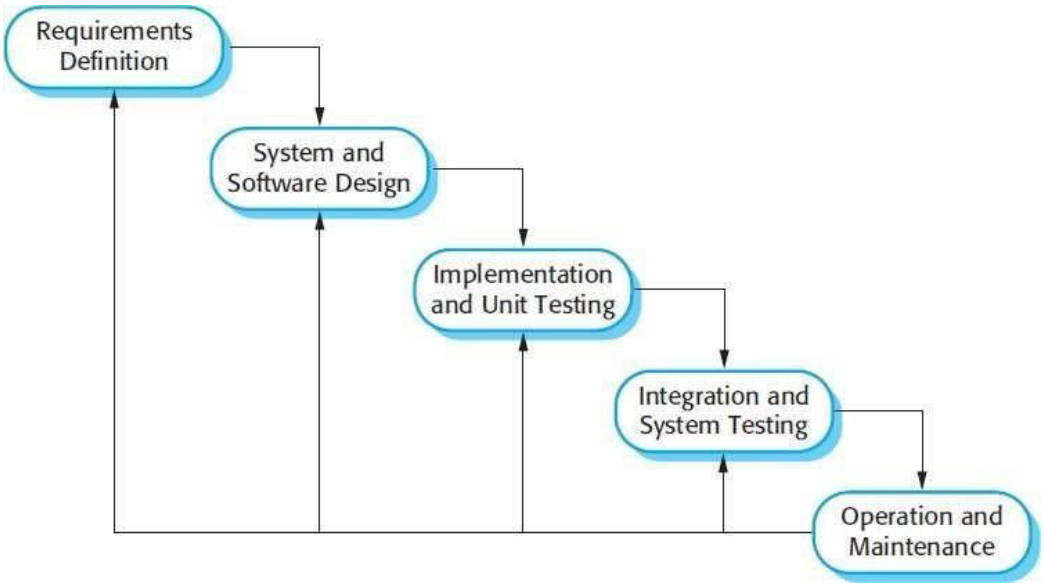
Waterfall model is the earliest SDLC approach that was used for software development.

The waterfall Model illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap.

Waterfall Model design

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

Following is a diagrammatic representation of different phases of waterfall model.


Requirements Definition
System and Software Design
Implementation and Unit Testing
Integration and System Testing
Operation and Maintenance

The sequential phases in Waterfall model are:

Requirement Gathering and analysis: All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification doc.

System Design: The requirement specifications from first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture.

Implementation: With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.

Integration and Testing: All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

Deployment of system: Once the functional and nonfunctional testing is done, the product is deployed in the customer environment or released into the market.

Maintenance: There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

Waterfall Model Application

Every software developed is different and requires a suitable SDLC approach to be followed based on the internal and external factors. Some situations where the use of Waterfall model is most appropriate are:

- 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.

The advantage of waterfall development is that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.

Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

3.d STUDY OF THE SYSTEM

Module:

The modules used in this software are as follows:

LOGIN: This module is for ADMIN and USER.

ADMIN has the authority to Insert, Update and Delete user, Products, Orders, Address, payment details. ADMINISTRATOR has the authority to insert, Update, Delete User details.

HOME: This page contains an overview of highlights for other

pages. ABOUT-US: This page contains the information about

EBS.

ADD TO CART: This page contains the information of each Product in

cart details. PAYMENT DETAILS:

SIGNIN MEMBERS: This page contains the information about the sign in members.

PAYMENT GATEWAY: This module facilitates secure online transactions by integrating with payment processors to accept various payment methods like credit cards, digital wallets, and bank transfers.

3.e INPUT AND OUTPUT

The main inputs, outputs and the major function the details are:

INPUT

User can login using id and password.

Admin can login using admin-id and password.

Admin insert and modify the Product Details, Order, Cart

Details etc.

User can make orders by registering user details.

Admin can modify user details searching them by id and name.

OUTPUT

Operator can view the details of Cart, Orders, Category.

Notifications for system events like security breaches.

Monitoring data related to system uptime.

3.f SOFTWARE REQUIREMENT SPECIFICATIONS

Software Requirements Specification provides an overview of the entire project. It is a description of a software system to be developed, laying out functional and nonfunctional requirements. The software requirements specification document enlists enough and necessary requirements that are required for the project development. To derive the requirements, we need to have clear and thorough understanding of the project to be developed. This is prepared after the detailed communication with project team and the customer.

The developer is responsible for: -

- ✓ Developing the system, which meets the SRS and solving all the requirements of the system?
- ✓ Demonstrating the system and installing the system at client's location after acceptance testing is successful.
- ✓ Submitting the required user manual describing the system interfaces to work on it and also the documents of the system.
- ✓ Conducting any user training that might be needed for using the system.
- ✓ Maintain the system for a period of one year after installation.

HARDWARE REQUIREMENTS:

- Computer that has a 1.6GHz or faster processor
- 4 GB (32 Bit) or 4GB (64 Bit) RAM (Add 512 MB if running in a virtual machine)
- HDD 20 GB Hard Disk Space and Above Hardware Requirements
- 5400 RPM hard disk drive
- DVD-ROM Drive

SOFTWARE REQUIREMENTS:

- WINDOWS OS (Windows 10, Windows 11, windows XP)
- MySQL Server (5.6.12)
- Xampp Server
- Adobe Dreamweaver CS6

3.g SOFTWARE ENGINEERING PARADIGM APPLIED

Software paradigms refer to the methods and steps, which are taken while designing the software. There are many methods proposed and are in work today, but we need to see where in the software engineering these paradigms stand. These can be combined into various categories, though each of them is contained in one another.

Programming paradigm is a subset of Software design paradigm which is further a subset of Software development paradigm.

There are two levels of reliability. The first is meeting the right requirement. A carefully and through systems study is needed to satisfy this aspect of reliability. The second level of systems reliability involves the actual working delivered to the user. At this level, the systems reliability is interwoven with software engineering and development.

There are three approaches to reliability.

1. Error avoidance: Prevents errors from occurring in software.
   Error detection and correction: In this approach errors are recognized whenever they are encountered and correcting the error by effect of error of the system does not fail.
2. Error tolerance: In this approach errors are recognized whenever they occur, but enables the
3. system to keep running through degraded perform or Appling values that instruct the system to continue process.

Maintenance:

The key to reducing need for maintenance, while working, if possible to do essential tasks.

1. More accurately defining user requirement during system development.

2. Assembling better systems documents.

3. Using some effective methods for designing, processing, and login and communicating information with project team members.

4. Making better use of existing tools and techniques.

4. SYSTEM DESIGN

4.a DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated.

DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

This context-level DFD is next "exploded", to produce a Level 1 DFD that shows some of the detail of the system being modelled. The Level 1 DFD shows how the system is divided into sub-systems (processes), each of whi1ch deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job, and shows the flow of data between the various parts of the system.

Data flow diagrams are one of the three essential perspectives of the structured-systems analysis and design method SSADM. The sponsor of a project and the end users will need to be briefed and consulted throughout all stages of a system's evolution. With a data flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented. The old system's dataflow diagrams can be drawn up and compared with the new system's data flow diagrams to draw comparisons to implement a more efficient system. Data flow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to report.

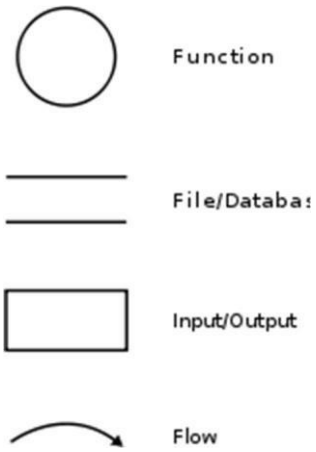
How any system is developed can be determined through a data flow diagram model.

In the course of developing a set of levelled data flow diagrams the analyst/designer is forced to address how the system may be decomposed into component sub-systems, and to identify the transaction data in the data model.

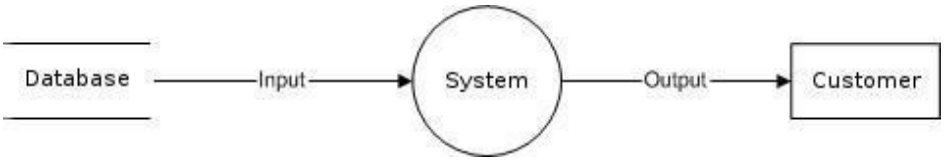
Data flow diagrams can be used in both Analysis and Design phase of the SDLC.

There are different notations to draw data flow diagrams. defining different visual representations for processes, data stores, data flow, and external entities.

DFD NOTATION



DFD EXAMPLE



Steps to Construct Data Flow Diagram: -

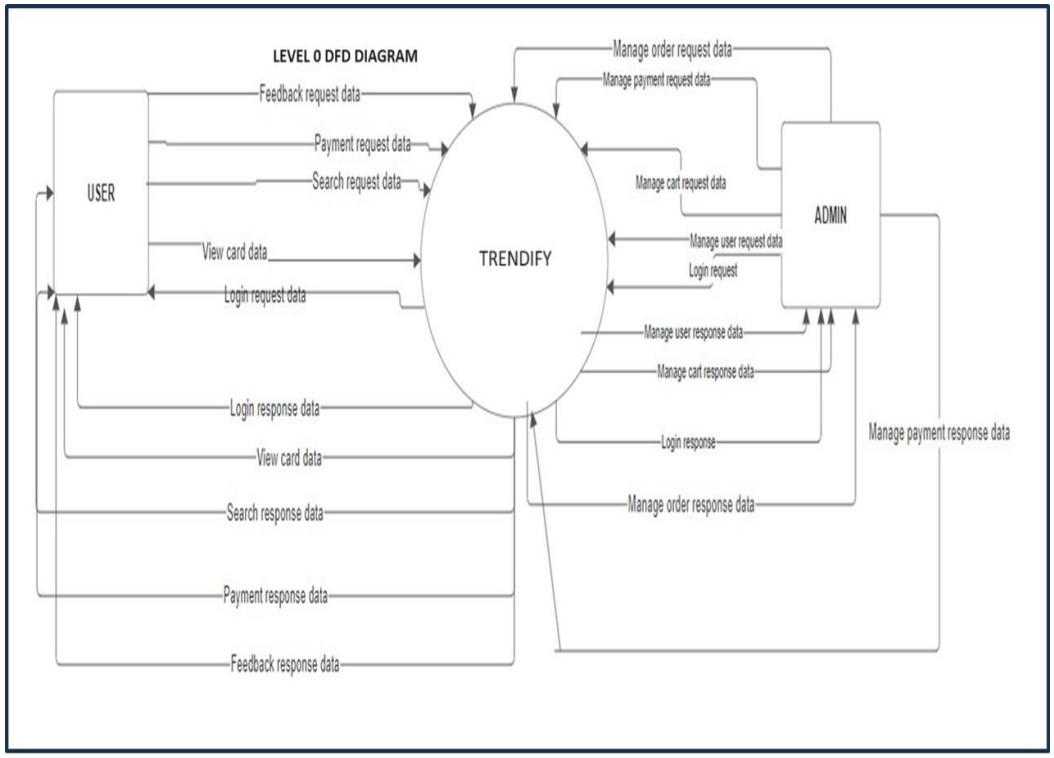
Four Steps are generally used to construct a DFD.

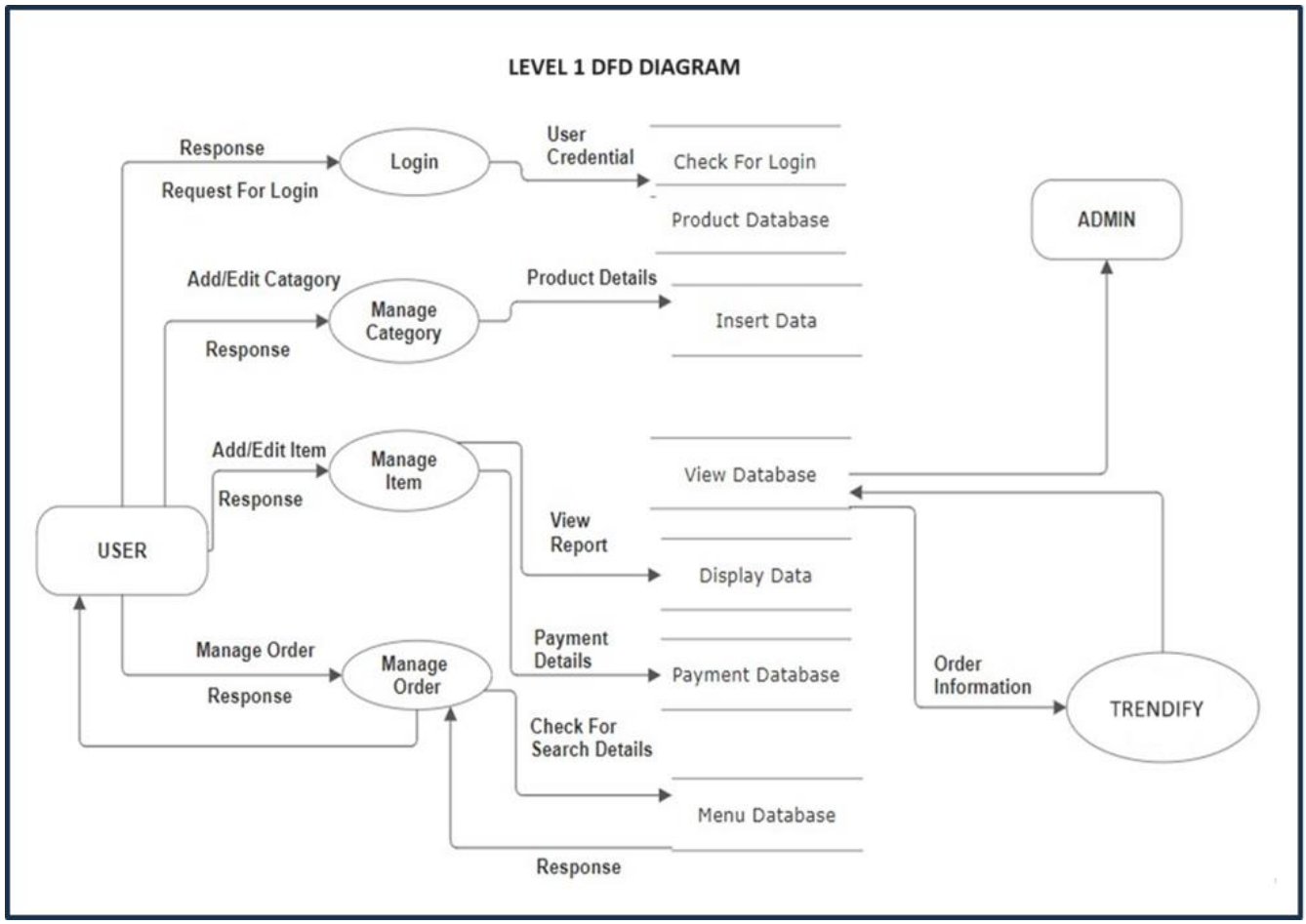
- Process should be named and referred for easy reference. Each name should be representative of the reference.
- The destination of flow is from top to bottom and from left to right.
- When a process is distributed into lower-level details they are numbered.
- The names of data stores, sources and destinations are written in capital

letters. Rules for constructing a Data Flow Diagram: -

- Arrows should not cross each other.
- Squares, Circles, Files must bear a name.
- Decomposed data flow squares and circles can have same names.
- Draw all data flow around the outside of the diagram.

-


LEVEL 0 DFD DIAGRAM
USER
TRENDIFY
ADMIN
Feedback request data
Payment request data
Search request data
View card data
Login request data
Manage order request data
Manage payment request data
Manage cart request data
Manage user request data
Login request
Manage user response data
Manage cart response data
Login response
Manage order response data
Manage payment response data
Login response data
View card data
Search response data
Payment response data
Feedback response data

- 
LEVEL 1 DFD DIAGRAM
Response
Request For Login
Login
User Credential
Check For Login
Product Database
ADMIN
Add/Edit Catagory
Response
Manage Category
Product Details
Insert Data
Add/Edit Item
Response
Manage Item
View Database
USER
View Report
Display Data
Manage Order
Response
Manage Order
Payment Details
Payment Database
Order Information
TRENDIFY
Check For Search Details
Menu Database
Response

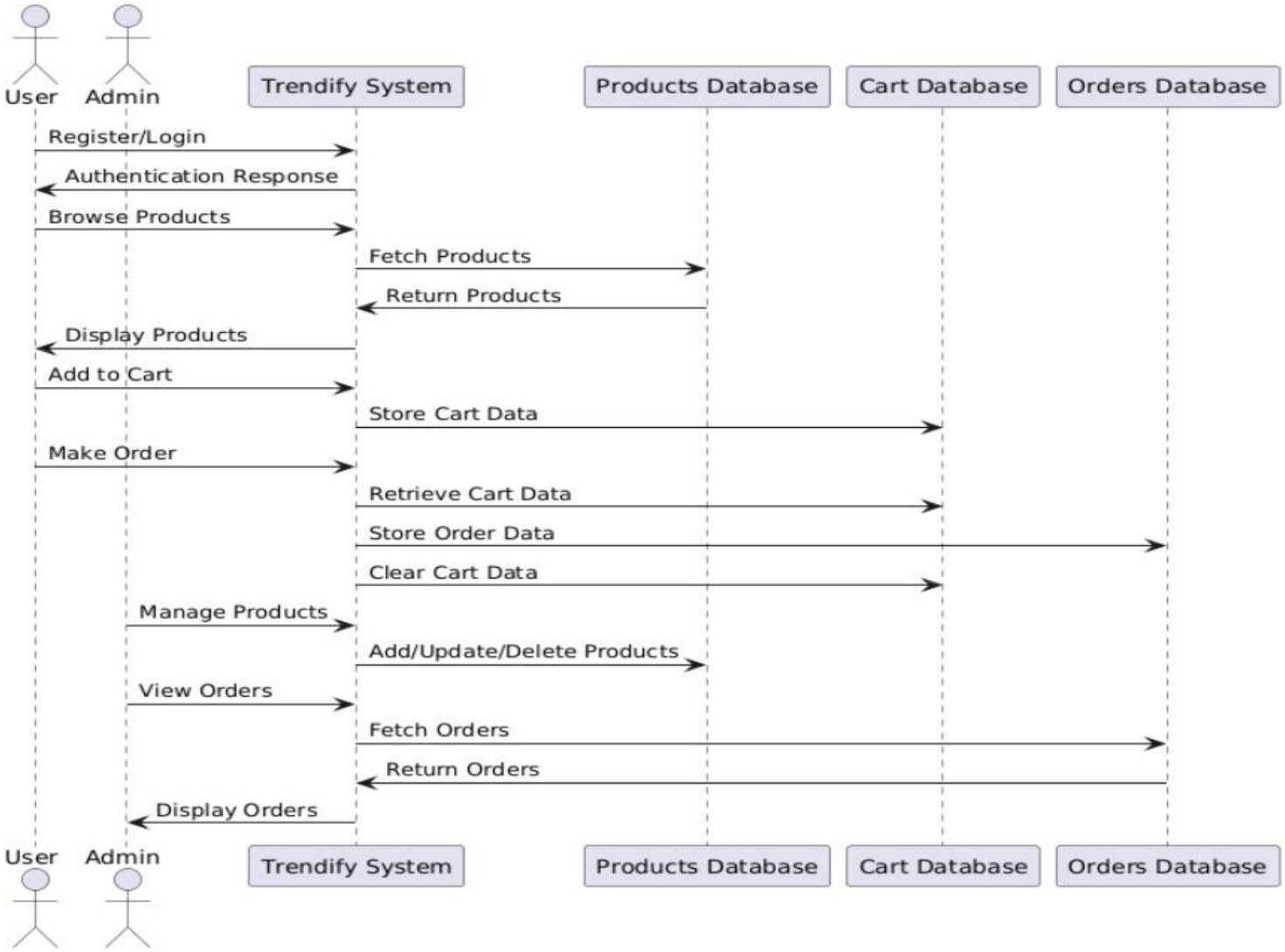
4b. SEQUENCE DIAGRAM

A **Sequence diagram** is an interaction diagram that shows how processes operate with one another and what is their order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called **event diagrams** or **event scenarios**.

A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

Sequence diagram is the most common kind of **interaction diagram**, which focuses on the **message** interchange between a number of **lifelines**.
Sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.
The following nodes and edges are typically drawn in a **UML sequence diagram**: **lifeline**, **execution-specification**, **message**, fragment, interaction, **state invariant**, **continuation**, **destruction occurrence**.

SEQUENCE DIAGRAM


User
Admin
Trendify System
Products Database
Cart Database
Orders Database
Register/Login
Authentication Response
Browse Products
Fetch Products
Return Products
Display Products
Add to Cart
Store Cart Data
Make Order
Retrieve Cart Data
Store Order Data
Clear Cart Data
Manage Products
Add/Update/Delete Products
View Orders
Fetch Orders
Return Orders
Display Orders
User
Admin
Trendify System
Products Database
Cart Database
Orders Database

4c. ENTITY RELATIONSHIP DIAGRAM

In software engineering, an entity–relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational. The main components of ER models are entities (things) and the relationships that can exist among them.

An entity–relationship model is the result of using a systematic process to describe and define a subject area of business data. It does not define business process; only visualize business data. The data is represented as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may have various properties (attributes) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity–relationship diagrams.

An ER model is typically implemented as a database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables point to indexes in other tables; such pointers are the physical implementation of the relationships.

The three schema approach to software engineering uses three levels of ER models that may be developed.

Conceptual data model

The conceptual ER model normally defines master reference data entities that are commonly used by the organization. Developing an enterprise-wide conceptual ER model is useful to support documenting the data architecture for an organization. A conceptual ER model may be used as the foundation for one or more logical data models. The purpose of the conceptual ER model is then to establish structural metadata commonality for the master data entities between the set of logical ER models. The conceptual data model may be used to form commonality relationships between ER models as a basis for data model integration.

Logical data model

The logical ER model contains more detail than the conceptual ER model. In addition to master data entities, operational and transactional data entities are now defined. The details of each data entity are developed and the relationships between these data entities are established. The logical ER model is however developed independent of technology into which it can be implemented.

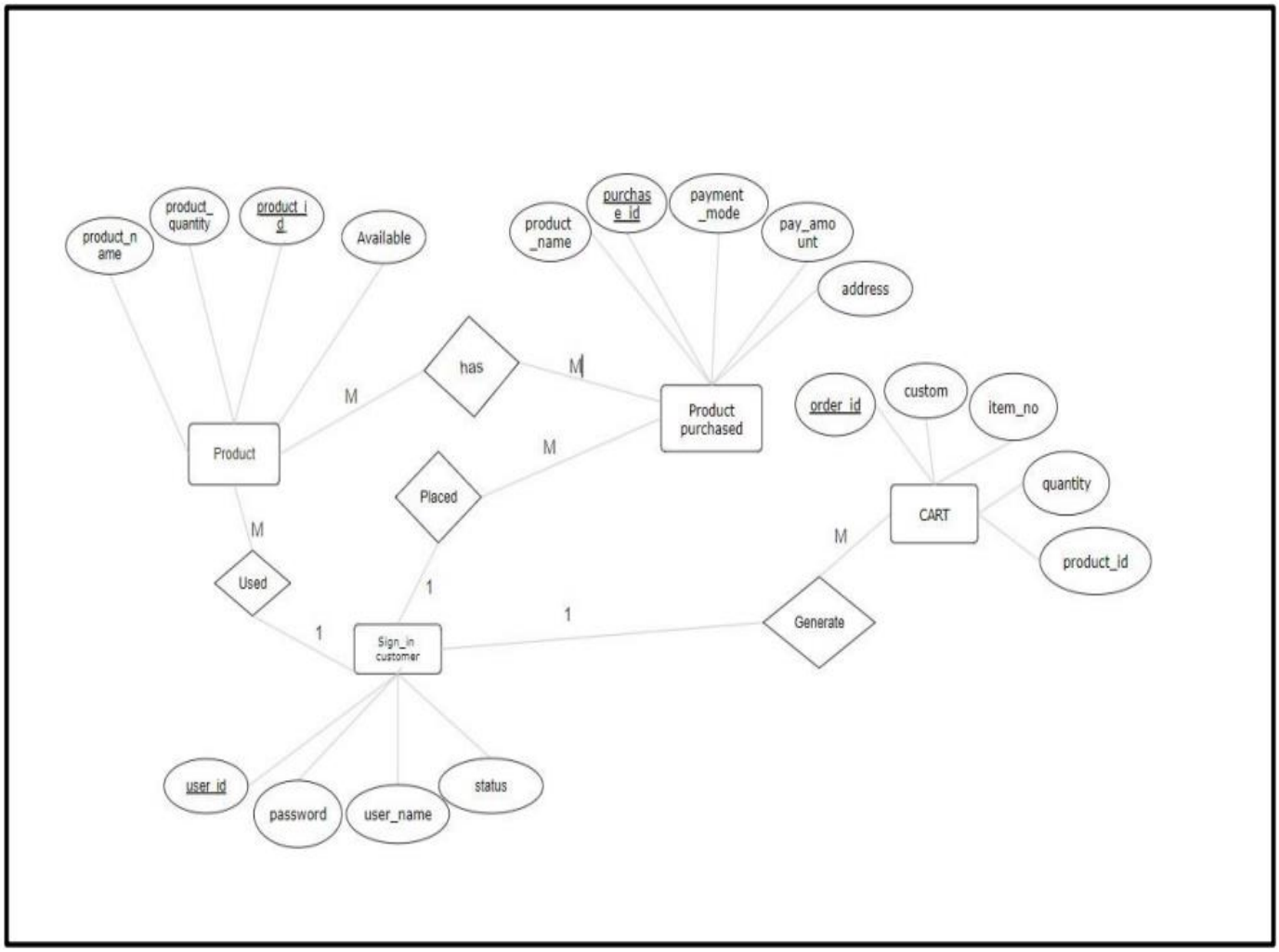
Physical data model

One or more physical ER models may be developed from each logical ER model. The physical ER model is normally developed to be instantiated as a database. Therefore, each physical ER model must contain enough detail to produce a database and each physical ER model is technology dependent since each database management system is somewhat different.

The physical model is normally instantiated in the structural metadata of a database management system as relational database objects such as database tables, database indexes such as unique key indexes, an database constraints such as a foreign key constraint or a commonality constraint. The ER model is also normally used to design modifications to the relational database objects and to maintain the structural metadata of the database.

The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in turn is mapped to a physical model during physical design. Note that sometimes, both of these phases are referred to as "physical design". It is also used in database management system.

ENTITY RELATIONSHIP DIAGRAM



4d. USE CASE DIAGRAM

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.
So only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. So use case diagrams are consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.
So to model the entire system numbers of use case diagrams are used.
The purpose of use case diagram is to capture the dynamic aspect of a system. But this definition is too generic to describe the purpose

Because other four diagrams (activity, sequence, collaboration and State chart) are also having the same purpose. So we will look into some specific purpose which will distinguish it from other four diagrams.
Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analyzed to gather its functionalities use cases are prepared and actors are identified.
Now when the initial task is complete use case diagrams are modelled to present the outside view.

So in brief, the purposes of use case diagrams can be as follows:

- ***Used to gather requirements of a system.***
- ***Used to get an outside view of a system.***
- ***Identify external and internal factors influencing the system.***
- ***Show the interacting among the requirements are actor***

How to draw Use Case Diagram?

Use case diagrams are considered for high level requirement analysis of a system. So when the requirements of a system are analyzed the functionalities are captured in use cases.
So we can say that uses cases are nothing but the system functionalities written in an organized manner. Now the second things which are relevant to the use cases are the actors. Actors can be defined as something that interacts with the system.

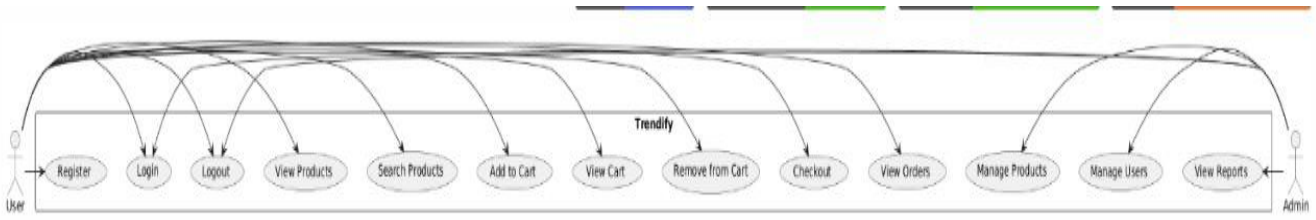
The actors can be human user, some internal applications or may be some external applications. So in a brief when we are planning to draw a use case diagram we should have the following items identified.

- Functionalities to be represented as an use case
- Actors
- Relationships among the use cases and actors.

Use case diagrams are drawn to capture the functional requirements of a system. So after identifying the above items we have to follow the following guidelines to draw an efficient use case diagram.

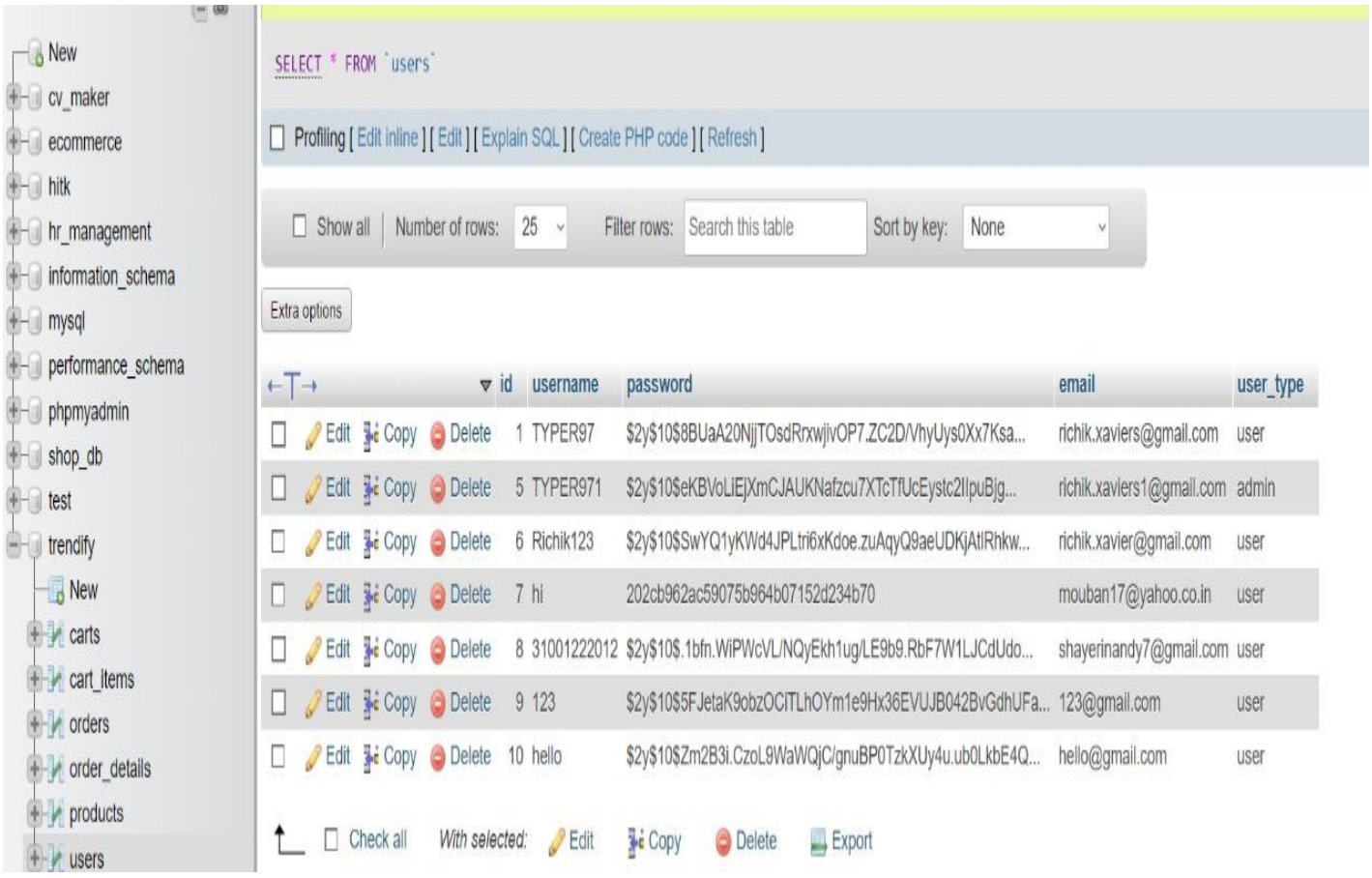
- The name of a use case is very important. So the name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.
- Do not try to include all types of relationships. Because the main purpose of the diagram is to identify requirements.
- Use note whenever required to clarify some important points.

USE CASE DIAGRAM

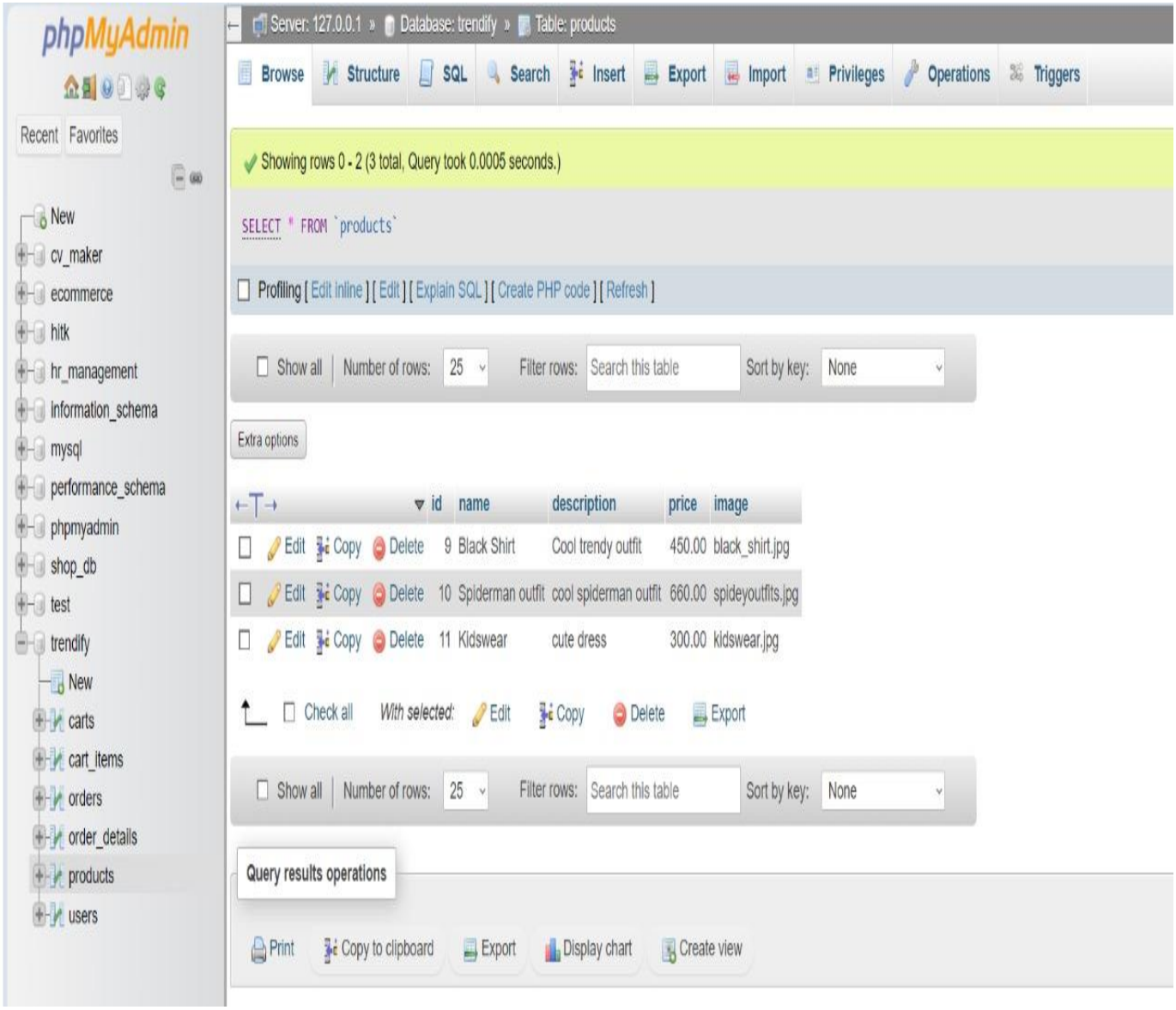


DATA DICTIONARY

User And Admin Login Table Data



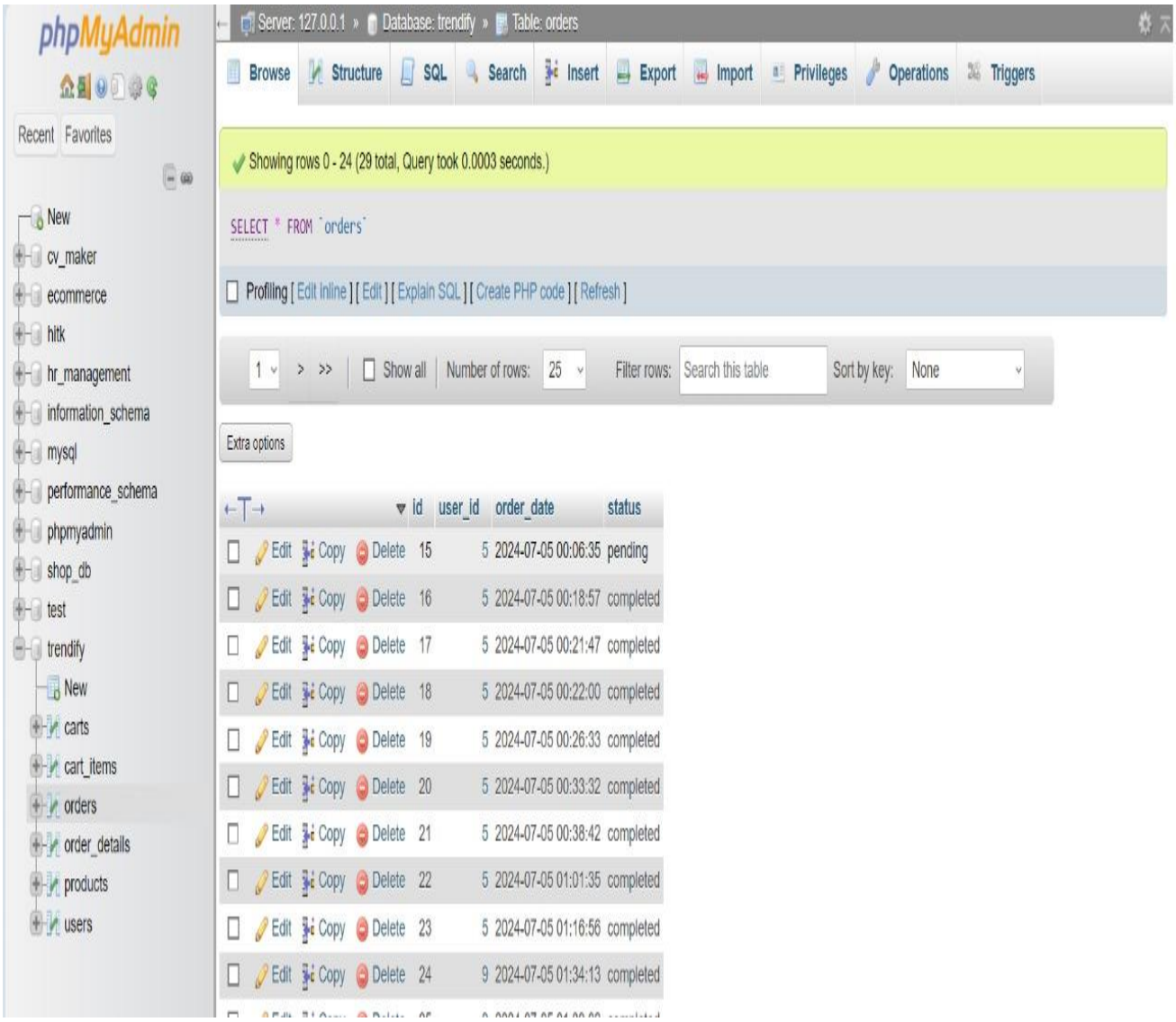
Add To The Product Into The Cart



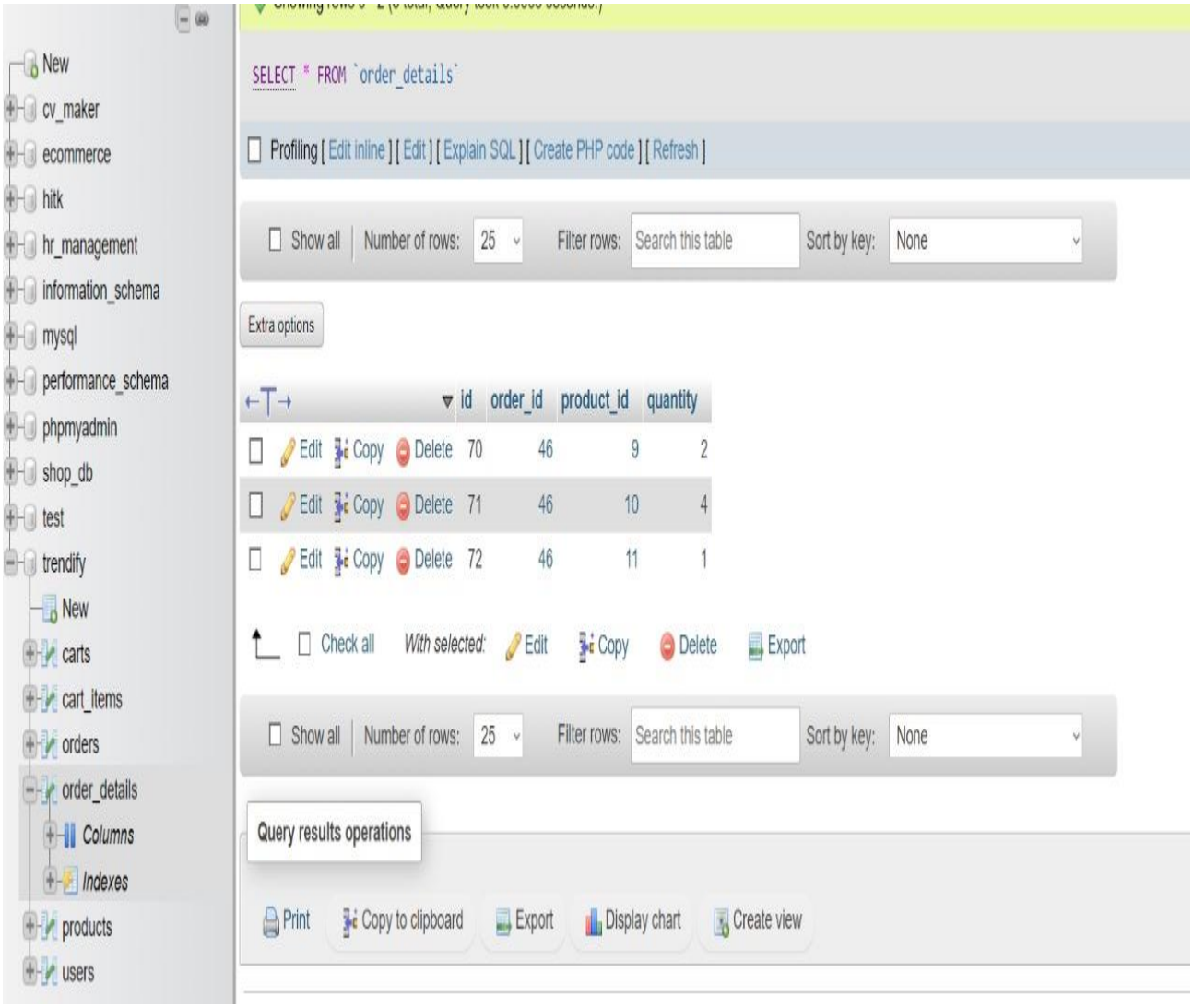
Snapshot Of Cart List

		id	cart_id	product_id	quantity
☐	Edit Copy Delete	64	5	10	4
☐	Edit Copy Delete	65	5	9	2
☐	Edit Copy Delete	66	5	11	1

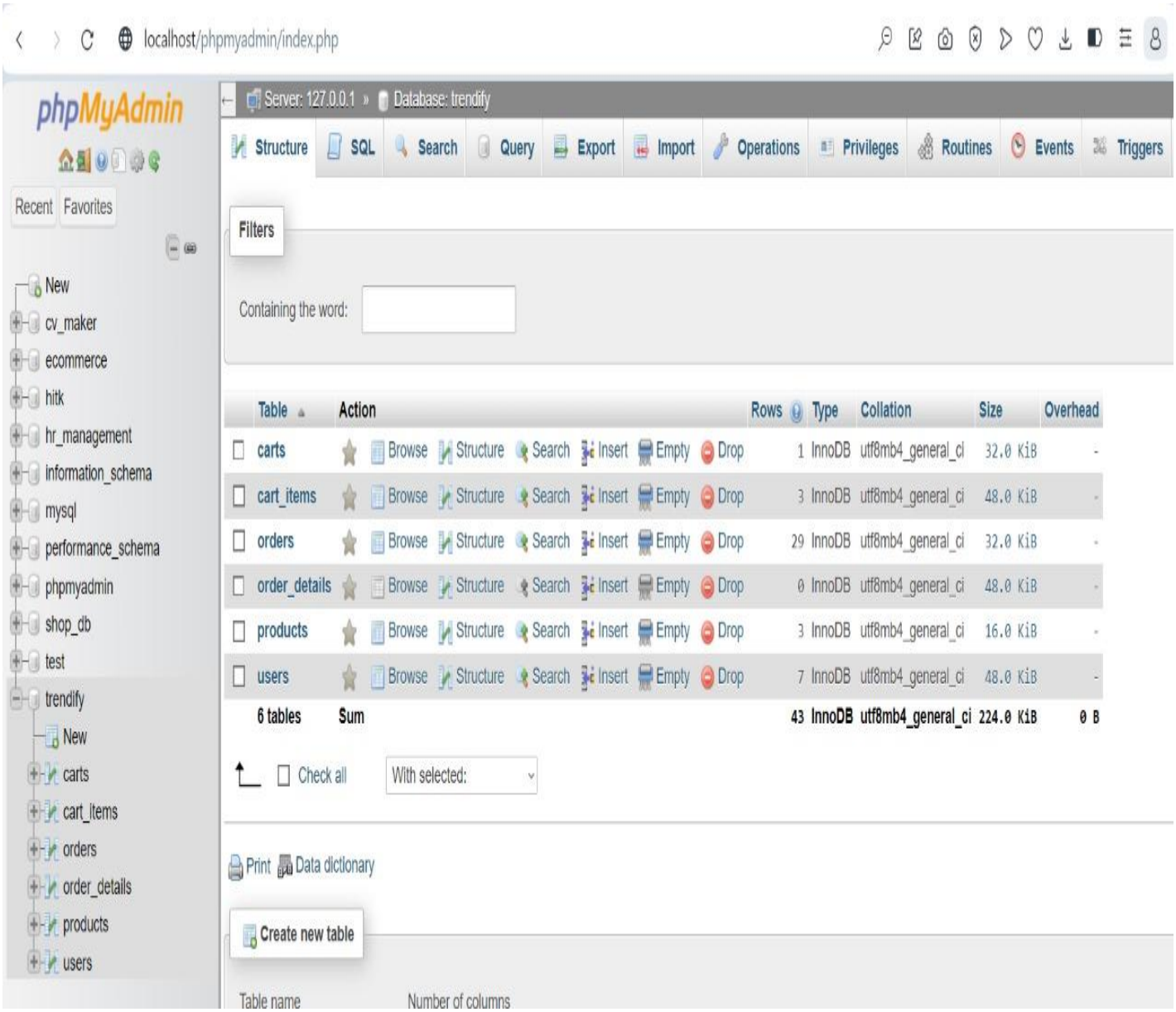
Snapshot Of Orders



Snapshots Of Order_Details



Database


localhost/phpmyadmin/index.php
phpMyAdmin
Recent Favorites
New
cv_maker
ecommerce
hitk
hr_management
information_schema
mysql
performance_schema
phpmyadmin
shop_db
test
trendify
New
carts
cart_items
orders
order_details
products
users
Server: 127.0.0.1 » Database: trendify
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers
Filters
Containing the word:
Table Action Rows Type Collation Size Overhead
carts Browse Structure Search Insert Empty Drop 1 InnoDB utf8mb4_general_ci 32.0 KiB -
cart_items Browse Structure Search Insert Empty Drop 3 InnoDB utf8mb4_general_ci 48.0 KiB -
orders Browse Structure Search Insert Empty Drop 29 InnoDB utf8mb4_general_ci 32.0 KiB -
order_details Browse Structure Search Insert Empty Drop 0 InnoDB utf8mb4_general_ci 48.0 KiB -
products Browse Structure Search Insert Empty Drop 3 InnoDB utf8mb4_general_ci 16.0 KiB -
users Browse Structure Search Insert Empty Drop 7 InnoDB utf8mb4_general_ci 48.0 KiB -
6 tables Sum 43 InnoDB utf8mb4_general_ci 224.0 KiB 0 B
Check all
With selected:
Print Data dictionary
Create new table
Table name
Number of columns

5. OUTPUT SCREEN

5.a USER INTERFACE DESIGN

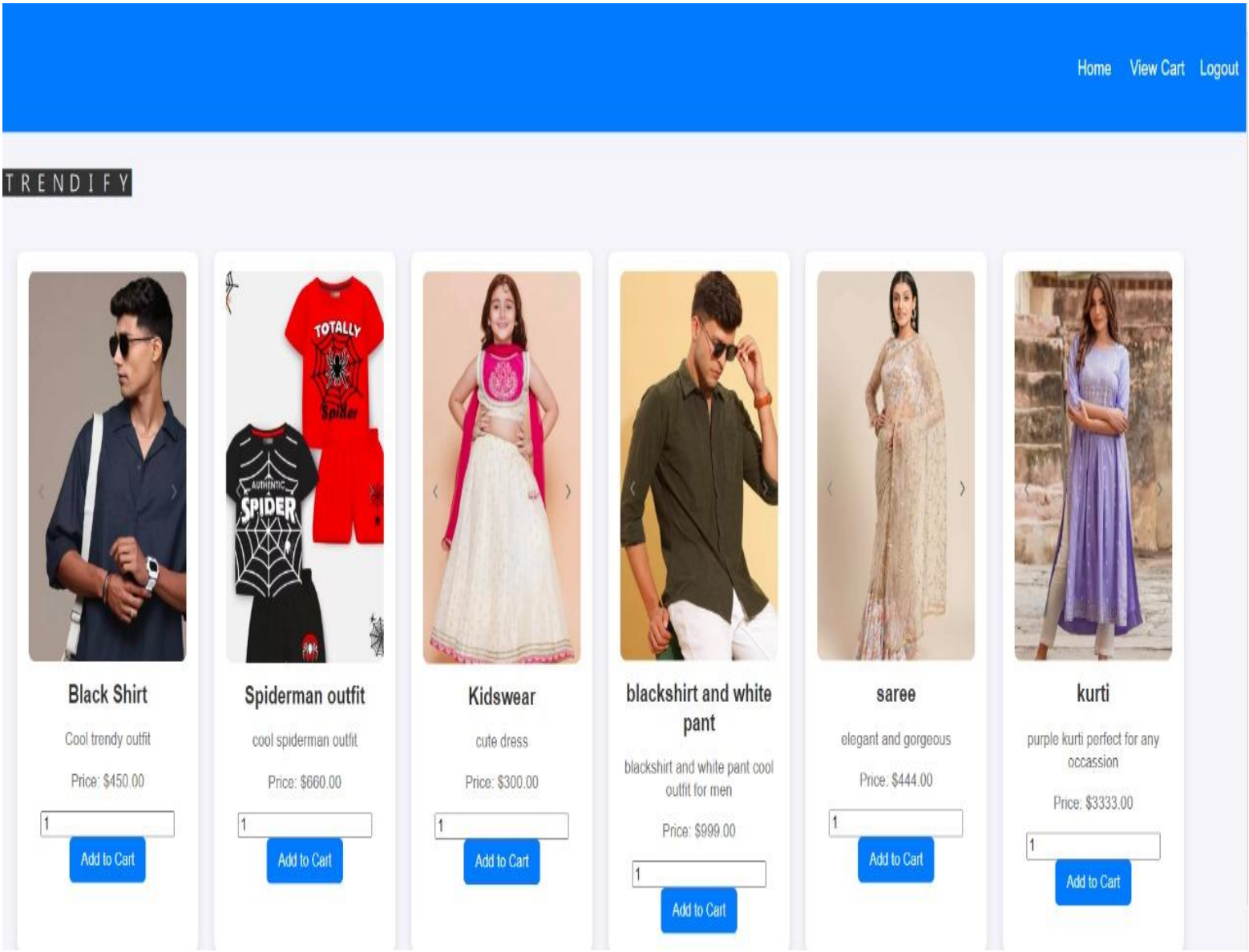
User interface design (UID) or user interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user- centered design).

Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. Graphic design and typography are utilized to support its usability, influencing how the user performs certain interactions and improving the aesthetic appeal of the design; design aesthetics may enhance or detract from the ability of users to use the functions of the interface. The design process must balance technical functionality and visual elements (e.g., mental model) to create a system that is not only operational but also usable and adaptable to changing user needs.

Interface design is involved in a wide range of projects from computer systems, to cars, to commercial planes; all of these projects involve much of the same basic human interactions yet also require some unique skills and knowledge. As a result, designers tend to specialize in certain types of projects and have skills centered on their expertise, whether that be software design,user research, web design, or industrial design.

5.b SNAPSHOTS

Home page



User login Page

USER LOGIN CODE

EXPLORER

TRENDIFY

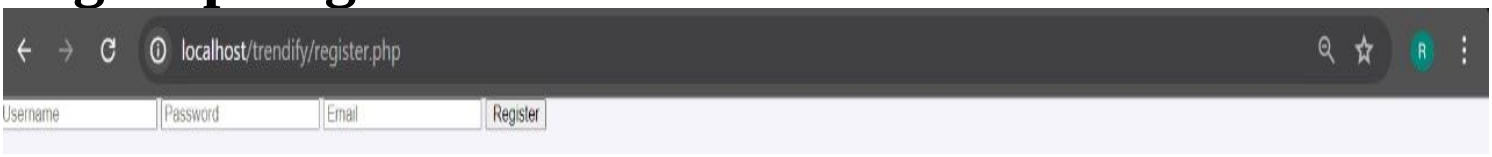
- admin
  - add_product.php
  - delete_product.php
  - edit_product.php
  - manage_products.php
- css
  - styles.css
- images
  - black_shirt.jpg
  - blackshirtwhitepantcombo.jpg
  - bluetrendy.jpg
  - IMG-20240708-WA0013.jpg
  - IMG-20240708-WA0014.jpg
  - IMG-20240708-WA0015.jpg
  - IMG-20240708-WA0017.jpg
  - IMG-20240708-WA0018.jpg
  - IMG-20240708-WA0020.jpg
  - kidswear.jpg
  - kurti.jpg
  - saree.jpg
  - spideyoutfits.jpg
  - WhatsApp Image 2024-07-07 at 19....
- js
  - scripts.js
- add_to_cart.php
- admin_dashboard.php
- admin_login.php
- admin_register.php
- checkout.php
- db.php

db.php | index.php | register.php | One time user Registeration | login.php

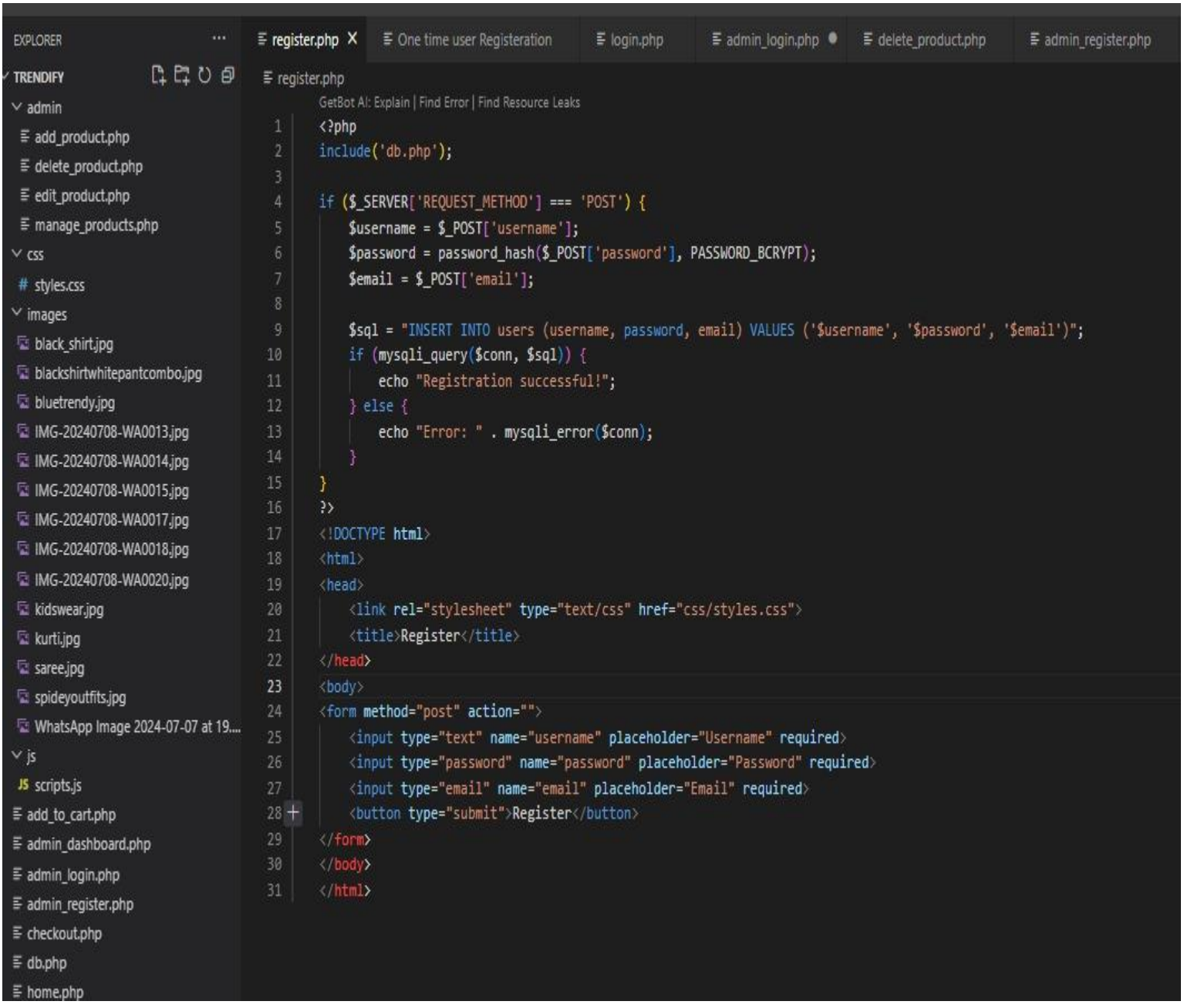
login.php

```php
<?php
include('db.php');
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = $_POST['username'];
    $password = $_POST['password'];
    $sql = "SELECT * FROM users WHERE username='$username'";
    $result = mysqli_query($conn, $sql);
    $user = mysqli_fetch_assoc($result);
    if ($user && password_verify($password, $user['password'])) {
        session_start();
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['user_type'] = $user['user_type'];
        header('Location: home.php');
    } else {
        echo "Invalid credentials!";
    }
}
?>
<!DOCTYPE html>
<html>
<head>
    <link rel="stylesheet" type="text/css" href="css/styles.css">
    <title>Login</title>
</head>
<body>
<form method="post" action="">
    <input type="text" name="username" placeholder="Username" required>
    <input type="password" name="password" placeholder="Password" required>
    <button type="submit">Login</button>
</form>
</body>
</html>
```

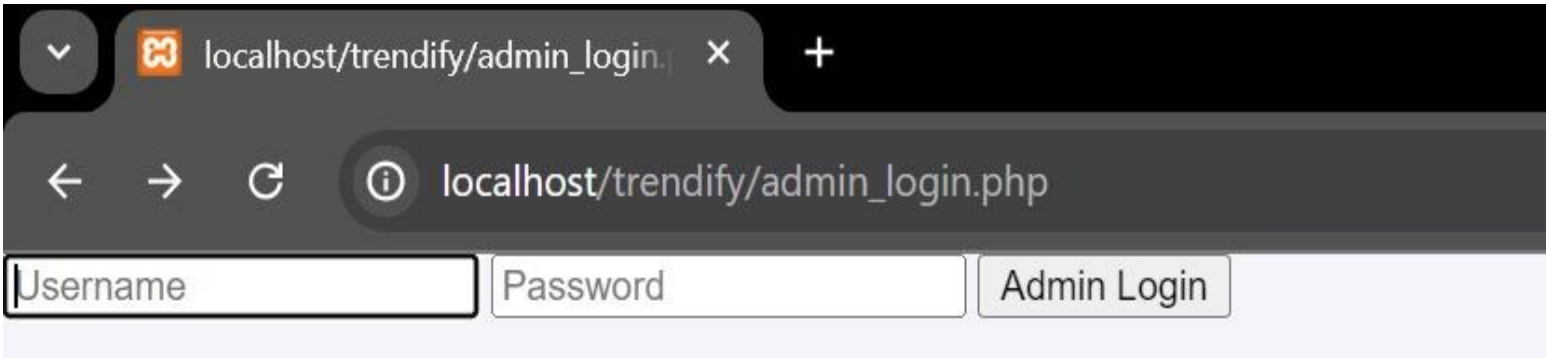
Sign Up Page



SIGN UP REGISTRATION CODE



Admin Login Page



Admin Login Code

```php
<?php
include('db.php');
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = $_POST['username'];
    $password = $_POST['password'];
    $sql = "SELECT * FROM users WHERE username='$username' AND user_type='admin'";
    $result = mysqli_query($conn, $sql);
    $admin = mysqli_fetch_assoc($result);
    if ($admin && password_verify($password, $admin['password'])){
        session_start();
        $_SESSION['user_id'] = $admin['id'];
        $_SESSION['user_type'] = $admin['user_type'];
        header('Location: admin_dashboard.php');
    }else {
        echo "Invalid admin credentials!";
    }
}
?>
<!DOCTYPE html>
<html>
<head>
    <link rel="stylesheet" type="text/css" href="css/styles.css">
</head>
<body>
<form method="post" action="">
    <input type="text" name="username" placeholder="Username" required>
    <input type="password" name="password" placeholder="Password" required>
    <button type="submit">Admin Login</button>
</form>
</body>
</html>
```

6. IMPLEMENTATION AND TESTING

A software system test plan is a document that describes the objectives, scope, approach and focus of software testing effort. The process of preparing a test plan is a usual way to think the efforts needed to validate the acceptability of a software product. The complete document will help people outside the test group understand the "WHY" and "HOW" product validation. It should be through enough to be useful but not so through that no one outside the test group will read it.

6.a INTRODUCTION

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

The implementation is the final and important phase. It involves user- training, system testing in order to ensure successful running of the proposed system. The user tests the system and changes are made according to their needs. The testing involves the testing of the developed system using various kinds of d

6.b OBJECTIVES OF TESTING:

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. Our user interface to utilize these functions is designed to be user-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability.

Process Overview

The following represents the overall flow of the testing process:

1. Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.
2. Identify which particular test(s) will be used to test each module.
3. Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.
4. Identify the expected results for each test.
5. Document the test case configuration, test data, and expected results. Perform the test(s).
6. Document the test data, test cases, and test configuration used during the testing process. This information shall be submitted via the Unit/System Test Report (STR).
7. Successful unit testing is required before the unit is eligible for component integration/system testing.
8. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.
9. Test documents and reports shall be submitted. Any specifications to be reviewed, revised, or updated shall be handled immediately.

6.c TEST CASES

A test case is a document that describe an input, action, or event and expected response, to determine if a feature of an application is working correctly. A test case should contain particular such as test case identifier, test condition, input data

Requirement expected results. The process of developing test cases can help find problems in the requirement or design of an application, since it requires completely thinking through the operation of the application.

TESTING STEPS

Unit Testing:

Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.

Integration Testing:

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. The objective is to take unit tested modules and builds program structure. All the modules are combined and tested as a whole.

Validation:

At the culmination of the integration testing, Software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

The function or performance characteristics confirm to specification and are accepted. A deviation from specification is uncovered and a deficiency lists is created.

Proposed system under consideration has been tested by using validation test and found to be working satisfactory.

Tested By:	Richik Banerjee
Test Type	Unit Testing
Test Case Number	1
Test Case Name	Admin Identification
Test Case Description	The admin should enter his/ her accurate admin id and password so that he/she can able to go for the further options. The test case will check the application for the same since a user can only login with the correct user id, password.

Item(s) to be tested	
1	Verification of the user id and password with the record in the database.

Specifications	
Input	Expected Output/Result
Correct User id and password	Successful login
Incorrect Id or Password	Failure Message

SNAPSHOT OF ADMIN DASHBOARD

Admin Dashboard

Add New Product Manage Products

Products

Black Shirt

Cool trendy outfit

$450.00

6.c WHITE BOX TESTING

In white box testing, the UI is bypassed. Inputs and outputs are tested directly at the code level and the results are compared against specifications. This form of testing ignores the function of the program under test and will focus only on its code and the structure of that code. Test case designers shall generate cases that not only cause each condition to take on all possible values at least once, but that cause each such condition to be executed at least once. To ensure this happens, we will be applying Branch Testing. Because the functionality of the program is relatively simple, this method will be feasible to apply.

Each function of the binary tree repository is executed independently; therefore, a program flow for each function has been derived from the code.

6.d BLACK BOX TESTING

Black box testing typically involves running through every possible input to verify that it results in the right outputs using the software as an end-user would. We have decided to perform Equivalence Partitioning and Boundary Value Analysis testing on our application.

System Testing

The goals of system testing are to detect faults that can only be exposed by testing the entire integrated system or some major part of it. Generally, system testing is

mainly concerned with areas such as performance, security, validation, load/stress, and configuration sensitivity. But in our case well focus only on function validation and performance. And in both cases we will use the black-box method of testing.

6.e OUTPUT TESTING

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen is found to be correct. The format was designed in the system design time according to the user needs. For the hard copy also; the output comes as per the specified requirements by the user. Hence output testing did not result in any correction for the system.

.

User Acceptance Testing:

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes whenever required

This is done in regard to the following point:

a. **Input Screen Design**
b. **Output Screen Design**
c. **Format of reports and other outputs**

6.f GOAL OF TESTING

"Program testing can be used to slow the presence of bug, but never to slow their absence." If the results delivered by the system are different from the excepted ones then the system is incorrect and these bugs should be fixed.

.

6.h INTEGRATION TEST REPORTS

Software testing is always used in association with verification and validation. In the testing phase of this project our aim is to find the answer to following two questions.

- Whether the software matches with the specification (i.e. process base) to verify the product.
- Whether this software in one client what wants (i.e. product base) to validate the product.

Unit testing and integration testing has been carried out to find the answer to above questions. In unit testing each individual module was test to find any unexpected behaviour if exists. Later all the module was integrated and flat file was generated.

FUNCTIONAL TESTING

These are the points concerned during the stress test:

Nominal input: character is in putted in the place of digits and the system has to flash the message "Data error"

Boundary value analysis: exhaustive test cases have designed to create an output report that produces the maximum (and minimum) allowable number of table entries.

Testing Method Used

We have adopted a testing method which is a mix of both (structural) and black box (functional) testing. For modules we have adopted white box testing. Then we integrated the module into sub - systems and further into the system. These we adopted black box testing for checking the correctness of the system.

Requirements Validated and Verified:

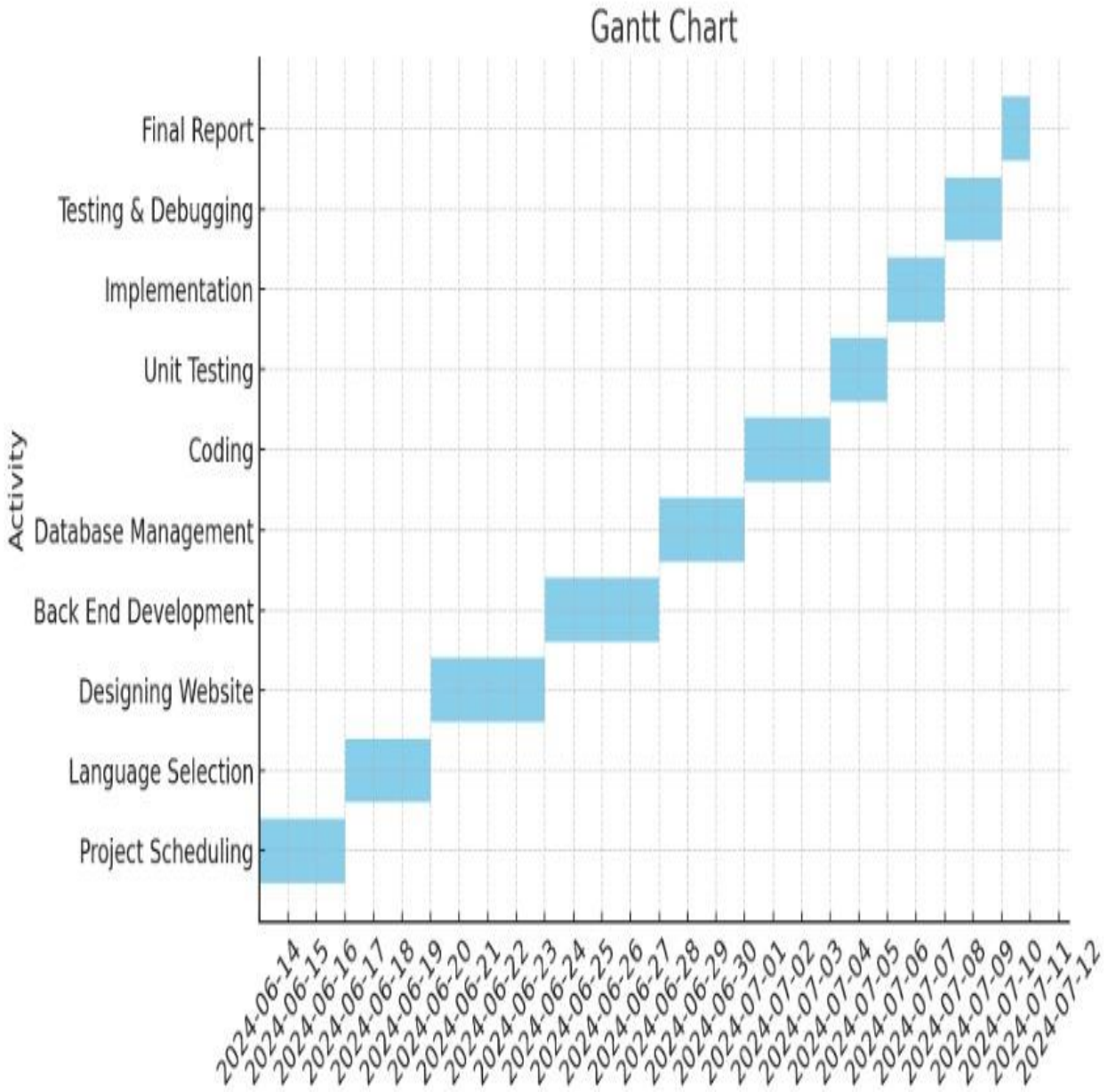
- The data is getting entered properly into database.
- The Screens are being loaded correctly.
- The Various functions specified are being performed completely.

7.GANTT CHART

PROJECT SCHEDULE

#	Activity Description	Planned Date	Actual Date	Budget Days	Status
1	**Project Scheduling**	14-06-2024	16-06-2024	03	**Done**
2	**Language Selection**	17-06-2024	19-06-2024	03	**Done**
3	**Designing Website**	20-06-2024	23-06-2024	04	**Done**
4	**Back End Development**	24-06-2024	27-06-2024	04	**Done**
5	**Database Management**	28-06-2024	30-06-2024	03	**Done**
6	**Coding**	01-07-2024	03-07-2024	03	**Done**
7	**Unit Testing**	04-07-2024	05-07-2024	02	**Done**
8	**Implementation**	06-07-2024	07-07-2024	02	**Done**
9	**Testing & Debugging**	08-07-2024	09-07-2024	02	**Done**
10	**Final Report**	09-07-2024	10-07-2024	02	**Done**

GANTT CHART


Gantt Chart
Activity
Final Report
Testing & Debugging
Implementation
Unit Testing
Coding
Database Management
Back End Development
Designing Website
Language Selection
Project Scheduling
2024-06-14
2024-06-15
2024-06-16
2024-06-17
2024-06-18
2024-06-19
2024-06-20
2024-06-21
2024-06-22
2024-06-23
2024-06-24
2024-06-25
2024-06-26
2024-06-27
2024-06-28
2024-06-29
2024-06-30
2024-07-01
2024-07-02
2024-07-03
2024-07-04
2024-07-05
2024-07-06
2024-07-07
2024-07-08
2024-07-09
2024-07-10
2024-07-11
2024-07-12

8.SYSTEM SECURITY MEASURES

8.a DATABASE SECURITY

System security measure is meant to be provided to make your system reliable and secured from unauthorized user may create threats to the system. So, you should follow some security measures. We have used security levels in database level at system level.

8.b SYSTEM SECURITY

If we talk about the system security in our proposed system we have implemented with the help of maintain the session throughout the system's use. Once a user has logged out than he/she will not be able to perform any task before signing back again.

A high level of authentic login is given to the system so this is a very tedious task to enter without authorization and authentication.

8.c LIMITATIONS

Since it is an online project, customers need internet connection to use it.

People who are not familiar with computers can't use this software.

Customer must have debit card or credit card to book tickets

.

9. CONCLUSION

This project has been appreciated by all the users in the organization. It is easy to use, since it uses the GUI provided in the user dialog. User friendly screens are provided. The usage of software increases the efficiency, decreases the effort. It has been efficiently employed as a Site management mechanism. It has been thoroughly tested and implemented.

10. FUTURE SCOPE AND FURTHER ENHANCEMENTS

In future we would like to keep working on this project and make new additions to provide users with more advanced features and more detailed information. We have set our sights on the following additions in future: -

Forget Password for admin and player.

Online payment process through debit and credit cards.

Auto mail will be sent to the ’s email-id when new admission takes place.

11. BIBLIOGRAPHY

https://www.w3schools.com

https://www.slideshare.com

https://www.scribd.com

https://www.tutorialspoint.com

https://www.youtube.com