

**E-COMMERCE WEBSITE A MINI PROJECT REPORT**

|  |  |
| --- | --- |
| **Submitted** | **by** |
| **BALAJI R** | **231801018** |
| **DEVANESH S M**  **DHANUSH R** | **231801009**  **231801011** |
|  |  |

In partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY IN

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS) THANDALAM

CHENNAI-602105

2024 - 2025

**CONTENTS OF BCA MAJOR PROJECT REPORT**

Project Report shall be prepared in **consultation with the Mentor.** It should clearly state the objectives and environment of the proposed Project to be undertaken. Ensure to include the following items while submitting your Project synopsis. Project Report may contain 40-45 pages and sequence of contents should be strictly in the following order:

1. Cover and Title page
2. Synopsis Approval Certificate / S/W Development Company Certificate
3. Index
4. Acknowledgement
5. Certificate of Originality
6. Abstract
7. Introduction/Aims and Objective
8. Feasibility Study
   1. Technical Feasibility
   2. Economic Feasibility
   3. Operational Feasibility
9. Analysis (Feasibility Study, DFD 0 Level, 1- Level and 2 Level/ER Diagram, and

Data structure, Table structure etc).

1. S/W & H/W Requirement Specification
2. Screen Shots
3. Coding
4. Implementation and Maintenance
5. Various types of Reports/Modules
6. Future scope of the Project

### INDEX

**TITLE PAGE No.**

**Acknowledgement Certificate of Originality Abstract**

1. **INTRODUCTION**
   1. **Features, Objective and Scope**
2. **TECHNOLOGY USED TO IMPLEMENTS**
3. **FEASIBILITY STUDY**
   1. **Technical Feasibility**
   2. **Economical Feasibility**
   3. **Operational Feasibility**
4. **PROJECT DIAGRAM**
   1. **DATA FLOW DIAGRAM (DFD)**
      1. **DFD Level-0**
      2. **DFD Level-1**
      3. **DFD Level-2**
   2. **USECASE DIAGRAM**
   3. **ENTITY RELATIONSHIP DIAGRAM (ER DIAGRAM)**
   4. **DATABASE TABLES**
5. **HARDWARE REQUIREMENTS**
6. **SOFTWARE REQUIREMENTS**
7. **SCREENSHOTS OF PROJECT**
8. **CODING OF PROJECT**
9. **IMPLEMENTATION METHODOLOGY**
10. **MAINTENANCE OF PROJECT**
11. **PROJECT OVERVIEW**
12. **PROJECT MODULES**
13. **FUTURE SCOPE OF THE PROJECT**

### ABSTRACT

The objective of this project is to develop a general purpose multilingual ecommerce store where any product (such as shoes, mobile phones, electronic items, and home appliances) can be bought from the comfort of home through the Internet. However, for implementation purposes, this project will deal with an Information Technology (IT) online shopping . It provides the user with a different catalog of products available for purchase in the store. To facilitate the online purchase, a shopping cart is provided to the user. The system is implemented using a three-tier approach, with a back-end MySQL database, a middle tier of Django, and a web browser as the front-end client. This document will discuss each of the underlying technologies to create and implement a multilingual ecommerce website and outlines all the process followed to come up with the application that is from analysis to testing.

**Keywords:** Django, ecommerce website, multilingual, MySQL database

## INTRODUCTION

Introduction E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming commonplace. The objective of this project is to develop a general purpose e-commerce store where any product (such as books, CDs, computers, mobile phones, electronic items, and home appliances) can be bought from the comfort of home through the Internet. However, for implementation purposes, this paper will deal with an online book store. An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest.

The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction.

Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e-mail notification is sent to the customer as soon as the order is placed.

E-Commerce Website Project Report explains about how to implement website for selling goods through online.This report consists of design details with source code.

**1.1. FEATURES , OBJECTIVE AND SCOPE FEATURES:\_**

* Reduce administrative and operating costs.
* Reduce inventory costs.
* Reduce the cost of procurement.
* Improve customer service and satisfaction.
* Increase communication efficiency and interaction with employees, vendors , customers and strategic partner.
* Increase revenues and profit margins.

**OBJECTIVE OF E-COMMERCE:-**

* + Improve service.
  + Save time
  + Reduce process errors.
  + Reduce the cost of core service provision.
  + Improve morale
  + The system helps in buying of goods, products and services online by choosing the listed products from website(E-Commerce site).

**SCOPE OF THE PROJECT:-**

Our E commerce project divides into two main parts:

* + The first part is dynamic creation of E-commerce web sites and managed them through our website.
  + Second part is shopping transactions which allow to company owners who we create an E-commerce web site to them to sell their products through our web site. This can be done by showing their products to consumers and then consumers can perform all shopping process.

### TECHNOLOGY USED TO IMPLEMENTS

* 1. **Project Category**

Web application made in Django

* 1. **Language(s) to be used :**

**Front End :**

Python, CSS, Bootstrap

**Back End :**

MySql

* 1. **Key Concept**
     + Online Shopping
     + Insert Data
     + Delete Data
     + Update Data
     + Detail of stock

### FEASIBILITY STUDY

Feasibility study is conducted once the problem is clearly understood . Feasibility study is a high level capsule version of the entire system analysis and design process. The objective is to determine quickly at a minimum expense how to solve a problem. The purpose of feasibility is not to solve the problem but to determine if the problem is worth solving.

The system has been tested for feasibility in the following points:

* + - Technical Feasibility
    - Economical Feasibility
    - Operational Feasibility
  1. **Technical Feasibility**

The project entitles “ E-Commerce Website” is technically feasibility because of the below mentioned feature. The project was developed in Django . It provides the high level of reliability, availability and compatibility .

* 1. **Economical Feasibilty**

The computerized system will help in automate the selection leading the profits and details of the organization. With this software , the machine and manpower utilization are expected to go up by 80-90% approx.

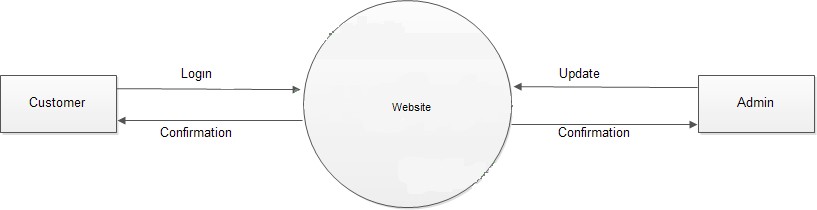
* 1. **Operational Feasibility**

In this project the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any queries for that particular contract can be known as per their requirements and necessaries.

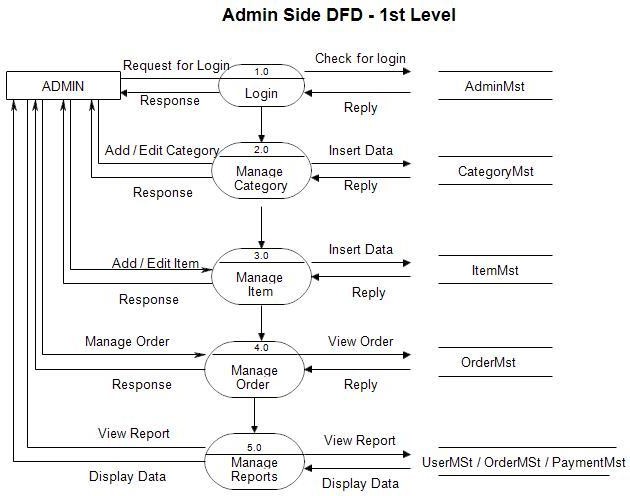
# PROJECT DIAGRAM

### DATA FLOW DIAGRAM (DFD)

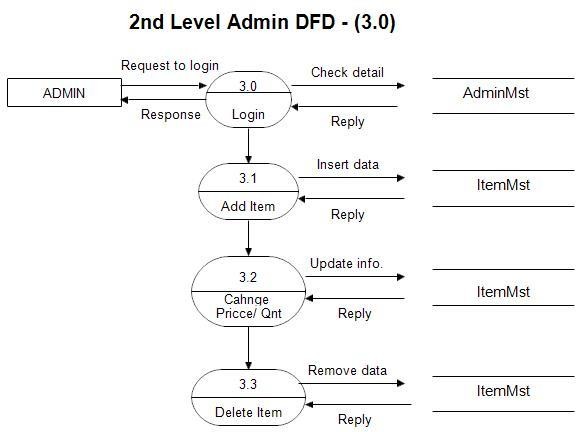
#### DFD LEVEL-0

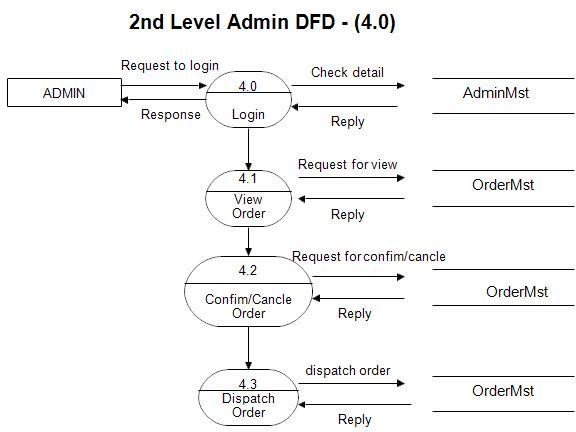


* + 1. **DFD LEVEL-1**

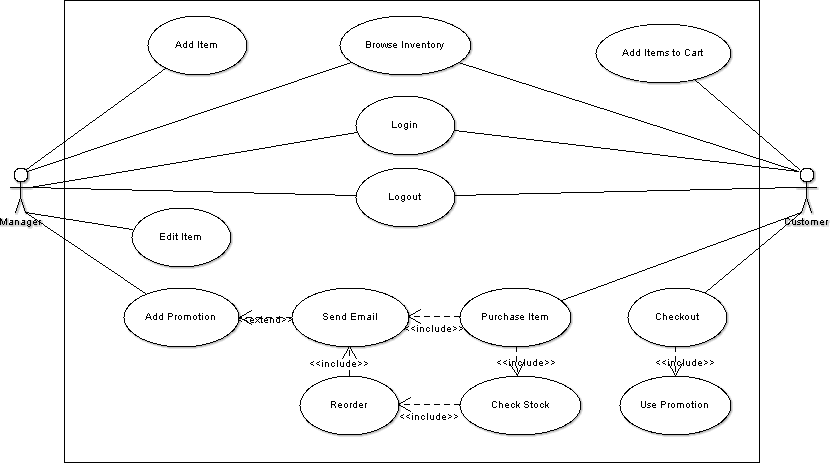


* + 1. **DFD LEVEL-2**

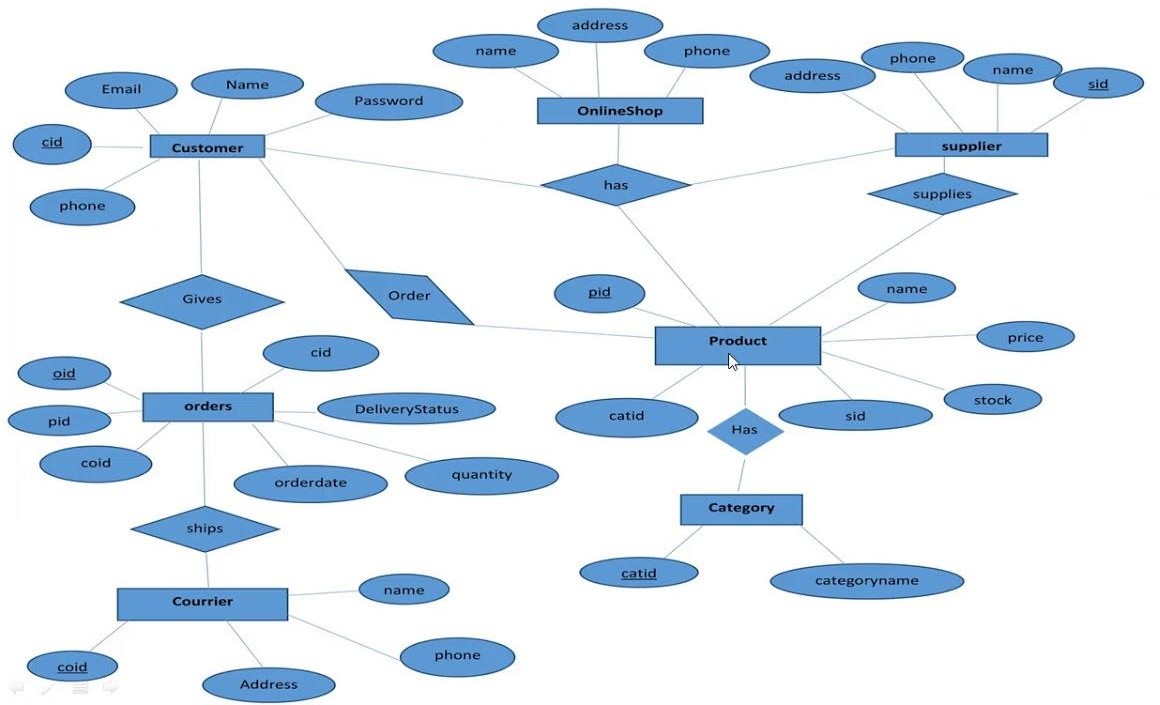


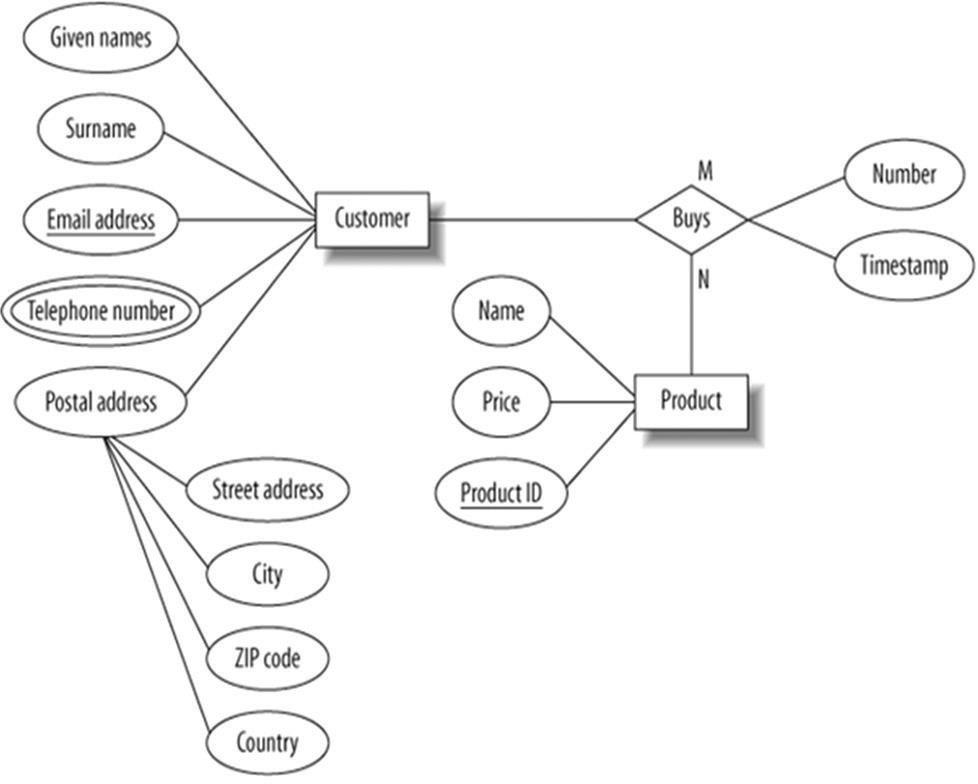


### USE CASE DIAGRAM



* 1. **Entity Relationship Diagram(ER-Diagram)**





**Entity-relationship diagram for the customer model**

* 1. **DATABASE TABLES**

**Admin Table:**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| Uname | Varchar | 20 |
| Password | Varchar | 20 |

**Product Table:**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Length** |
| PId | Int | 4 |
| ProductName | Varchar | 100 |
| ProductType | Varchar | 50 |
| Image | Varchar | 100 |
| Price | Varchar | 4 |

### HARDWARE REQUIREMENTS

* Internet
* Hard Disk
* Cache Memory
* Database System

### SOFTWARE REQUIREMENTS

* Web Technologies: Django
* Language: Python
* Database: MySQL Server
* Operating System: Windows 10,7,8

### IMPLEMENTATION METHODOLOGY

Model View Controller is a software design pattern for developing web applications.

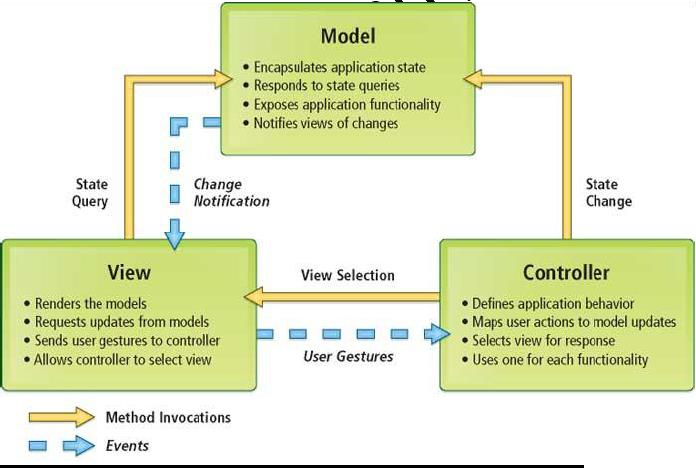
A model view controller part is made up of three parts:

* **Model:-** The lowest level pattern which is responsible for maintaining data.
* **View:-** It is responsible for displaying all or a portion of data to the user.
* **Controller:-** It is the software code that controls the interaction between the view and the model.

MVC is popular as it isolates the application logic from the user interface layer. Here the controller receives all request from the application and works with the model to prepare any data needed by the view.

The view then uses the data prepared by the controller to generate a final presentable response.

The MVC abstraction can graphically be represented as follows:



### MAINTENANCE OF PROJECT

Website maintenance is very important for any website irrespective of any business and industry. It includes maintenance of existing or a functioning website to be secured,ensures data prevention without any data being loss during the course of time. It also includes checks for any broken links periodically and makes sure that there are more recent updates. The main aim is to make sure website performance is driving results. This will help to attract new customers visiting the webpage and also retains the existing customers. It also helps in implementing new and advanced search engine techniques to showcase the latest products and services they offer to the common public. Website maintenance is applicable to any size of the company as it delivers a good business and revenue.

E-Commerce Website Maintenance will follow the above mentioned services on a common note, and also follow a lot more superior services

incorporated in E-Commerce model with utmost care. Unlike the other dynamic and static websites, E-Commerce sites are into direct selling of any product that are posted and customers can opt to buy it online on the go or from where they are. Hence, E-Commerce Website Maintenance services play a humungous role and can correlate to the business model and revenue model.

### PROJECT OVERVIEW

Once customer entered with his own username and password, at that time automatically one shopping cart will be created, once user select an item it will add to cart. In case user thinks the selected item is not useful for me, then deleted that item from shopping cart.

Customer selected some items, but in his credit or debit card haven’t that much balance, then he was logout from website, the selected items are stored at cart with specific users with his allotted carts, after some days he bought those items then automatically deleted from the cart.

### PROJECT MODULES

**Modules:** This project contains 3 modules, those are

##### Admin

* + **Products**

##### User

**Modules Description:**

##### Admin:-

When admin login, he saw the customer’s database, means how many users are authenticated to this website and how many users are transact everyday, and newly items are inserting into products.

##### Products:-

This module contains product name, and related image, and cost of its. Like home appliances, shoes, clothes,etc.

Whatever customer wants from the shopping cart.

##### User:-

User entered into with his username and password, when he entered into this, he saw what items are available today, this facility is available for this site. Chosen different items from website get those through door delivery.

**FUNCTION MODEL OF E-COMMERCE WEBSITE:**

**Customer Registration:**

In the page of Customer Registration, the customer fill personal data which contains Full name, User name, password, address, E-mail and another crucial data.

**Admin Registration:**

In the page of Admin Registration, the admin fill personal data which contains Full name, User name, password, address, E-mail and another crucial data.

**Login:**

In the Home Page, User asked to enter username and password to login, our task to check if the user logged in were admin, customer or vendor. If user name of logged user in customers table so the logged user was customer if not we will check admin table. If password not match we will redirect to error page to show error message (Please Enter Right Password) and from that page user can use back button to return to login form to be able to enter right password.

And if password match customer will be redirect to customer Home Page to see his/her own personal information, and if user enters as admin he/she will be redirect to Admin Page to get all his authorized to manage different aspects in our site (Add or Update any Department). And if user name not found it also redirect To Error Page to show error message (Your User Name Not Found!! Please Enter Right One).

### CONCLUSION

The **Student Course Registration System** project has successfully met its primary objectives, demonstrating key functionalities required for a robust academic registration system. The development of this system showcases an effective platform where students can easily register, browse, and enroll in courses, while administrators can efficiently manage courses and student enrollments.

From a **user perspective**, the system provides a smooth and seamless experience. The registration and login processes are intuitive, allowing students to easily sign up and access their accounts. The ability to browse and select courses based on course name, department, and schedule streamlines the decision-making process for students, ensuring they can enroll in the most relevant courses without hassle.

For **administrators**, the system provides powerful tools for managing courses and student data. Admins can effortlessly add, update, and delete courses, track student registrations, and approve or reject course enrollments. This functionality ensures that the system operates smoothly, with administrators maintaining control over the registration process.

The **email integration** feature significantly improves communication. Automated emails confirm registrations and course enrollments, providing students with real-time updates and reducing the administrative burden. This enhances both the user experience and operational efficiency, keeping all stakeholders informed throughout the process.

In terms of **performance**, the system has shown stability and responsiveness in handling student registrations and course enrollments, performing well under regular testing. However, for future scalability, the platform can benefit from further stress testing and performance optimization, ensuring it can handle higher traffic volumes and user interactions during peak registration periods.

Overall, the project has successfully delivered a functional, user-friendly, and administratively efficient system. The foundation built is solid, offering an intuitive interface for both students and administrators. As the system continues to evolve, there are opportunities to expand its capabilities, such as adding more advanced reporting features, improving the user interface for mobile platforms, and incorporating additional functionality like course waitlists and automated reminders.

**14.REFERENCE**

**Python Development and Tkinter Resources:**

* **Python Documentation: Official documentation for Python, including libraries such as Tkinter for building GUI applications. Available at:** [**https://docs.python.org**](https://docs.python.org)
* **Tkinter Documentation: Detailed resources for creating graphical user interfaces with Tkinter in Python. Available at:** [**https://tkdocs.com**](https://tkdocs.com)

**Database Management:**

**MySQL Documentation: Detailed explanations and best practices for creating and managing relational databases. Available at:** [**https://dev.mysql.com/doc**](https://dev.mysql.com/doc)

**Project Management and Development Tools:**

**GitHub: For version control and project collaboration. Documentation available at:** [**https://docs.github.com**](https://docs.github.com)