

A Project Report

On

INDEV – India's Newest

Direct Ecommerce

Vendor

Abstract of The Project

This project, **INDEV – India’s Newest Direct E-Commerce Vendor**, is an advanced e-commerce platform designed to connect multiple buyers and sellers in a seamless digital marketplace. It offers a **multi-vendor multi-buyer system**, enabling sellers to list products and manage orders while allowing buyers to browse, purchase, and track deliveries efficiently.

The platform addresses key challenges in traditional eCommerce, such as **limited reach, high operational costs, and inefficiencies in order processing**. By integrating a streamlined order management system, and a user-friendly interface, INDEV ensures a smooth and efficient shopping experience.

Developed using **Microsoft Visual Studio Code, HTML, CSS and JS for frontend** and **Python Django for backend**, the system ensures scalability, security, and reliability. With features like **product listing, order placement, delivery tracking, and user management**, the platform aims to revolutionize the eCommerce landscape.

This project highlights the importance of a **direct-to-consumer approach**, reducing dependency on intermediaries and enhancing the overall efficiency of online shopping.

Index

| SR NO | Page |
|-------|--|
| 01. | Front Page |
| 02. | Abstract |
| 03. | Index |
| 04. | Project Profile |
| 05. | Identification of Problem & Proposing Solution |
| 06. | Pre-Dev Requisites & Post-Dev Requirements |
| 07. | Three Categories of Users |
| 08. | Data Flow Diagram |
| 09. | Entity Relationship Diagram |
| 10. | Data Dictionary |
| 11. | Technologies Used from Django Framework |
| 12. | Guest Side User Input |
| 13. | Buyer Side User Input |
| 14. | Seller Side User Input |
| 15. | Admin Side User Input |
| 16. | Future Enhancements |
| 17. | Bibliography |
| 18. | Back Cover |

Project Profile

| Header | Text |
|--------------------|---|
| Project Title | INDEV |
| Project Definition | Ecommerce Website |
| Project Type | Web Based Application |
| Developed By | Jaspreet Singh Raval Viraj Shaikh Sahil |
| Frontend Tool | Microsoft Visual Studio Code 2024 |
| Backend Tool | Python Django Localhost Server |
| Database | SQLite |

Identification of Problem and Proposing Solution

Problems Within Existing System

- 1. Limited Reach** – Many platforms struggle to serve Tier 2 & Tier 3 cities, leaving a large customer base underserved.
- 2. Time-Consuming Process** – Lengthy product searches, checkout delays, and slow deliveries make shopping inefficient.
- 3. High Manpower Requirements** – Traditional commerce models require extensive workforce management for sellers, logistics, and customer support.
- 4. Increased Product Prices** – Middlemen and commissions drive up costs, making products more expensive for consumers.
- 5. Resource-Intensive Operations** – Managing multiple sellers, warehouses, and logistics networks leads to higher operational costs.
- 6. Complex Supply Chain** – Multi-tier vendor systems create inefficiencies in product availability and delivery timelines.
- 7. Data Privacy & Security Risks** – Customer data is often shared with multiple vendors, increasing the risk of breaches.

Need For New System

- 1. Wider Reach** – A direct vendor model ensures seamless access to Tier 2 & Tier 3 cities, expanding market coverage.
- 2. Faster Shopping & Delivery** – Eliminating middlemen and other streamlines processes, reducing delays in order fulfillment.
- 3. Reduced Manpower Costs** – Automating order processing, logistics, and customer support lowers operational workforce dependency.
- 4. Lower Product Prices** – Direct sourcing from manufacturers minimizes extra charges, making products more affordable.
- 5. Efficient Resource Utilization** – Optimized inventory and logistics management cut down unnecessary resource consumption.
- 6. Simplified Supply Chain** – A direct vendor approach enhances product availability and delivery speed.
- 7. Improved Data Security** – Secure transactions and direct vendor interactions minimize data privacy risks.

Proposed New System – INDEV

INDEV is a **multi-vendor, multi-buyer eCommerce platform** that connects sellers directly with consumers, eliminating middlemen to ensure affordable pricing, buyer & seller authenticity.

Key Features of the Proposed System :-

- 1. Multi-Vendor System** – Allows multiple vendors to list and sell products directly, ensuring authenticity and competitive pricing.
- 2. Multi-Buyer System** – Supports multiple customers simultaneously with a seamless shopping experience.
- 3. Product Listing** – Vendors can easily upload product details, images, and pricing.
- 4. Placing Orders** – Customers can browse, add items to the cart, and place orders for one or multiple products simultaneously.
- 5. Delivery Tracking System** – Real-time shipment tracking keeps customers updated on order status.
- 6. Report Generation** – Registered vendors are provided additional features of generating reports of their own.
- 7. Scalable & Future-Ready** – Built on Python Django with a user-friendly frontend, designed for future expansion and enhanced features.

INDEV – INDIA’S NEWEST DIRECT ECOMMERCE VENDOR

- **INDEV** – INDEV is an Acronym
- **I – India’s** : Focused on transforming India’s eCommerce landscape.
- **N – Newest** : A fresh, innovative platform redefining online shopping.
- **D – Direct** : Eliminates middlemen for better pricing and authenticity.
- **E – E-commerce** : A seamless digital marketplace for diverse products.
- **V – Vendor** : Connecting consumers directly to trusted brands.

□ Introduction to INDEV

❖ What is INDEV?

- ✓ A direct eCommerce platform offering a seamless shopping experience.
- ✓ Eliminates middlemen to ensure better pricing and transparency.

❖ Why INDEV?

- ✓ Because the name easily conveys the meaning and its purpose.
- ✓ It has a flare of patriotism to it.

Pre-Development Requisites and Post-Development Requirements

Pre-Development Requisites

➤ Hardware :-

- 1. Device :** Desktop, Laptop
- 2. Processor :** 2.5 GHZ +
- 3. RAM :** At least 8GB RAM (16GB recommended for better performance)
- 4. Storage :** 10 GB of free space (for media files)

➤ Software :-

- 1. Operating System :** Windows, macOS, Linux
- 2. Browser Support :** Chrome, Edge , etc (latest versions)
- 3. Development Environment :** Python and its modules
 - 1) Pip**
 - 2) Virtualenv**
 - 3) Django**
 - 4) Pillow**
 - 5) Csv**
- 4. Software :** Visual Studio Code (latest version)

Post-Dev Hardware Requirements

➤ Client Side :-

- 1. Device :** Desktop, Laptop
- 2. Processor :** 2.0 GHZ +
- 3. RAM :** At least 4GB RAM (8GB recommended for better performance)
- 4. Storage :** 1 GB of free space (for browser cache)

➤ Server Side :-

- 1. Device :** Desktop, Laptop
- 2. Processor :** 2.5 GHZ +
- 3. RAM :** At least 8GB RAM (16GB recommended for better performance)
- 4. Storage :** 10 GB of free space (for media files)

Post-Dev Software Requirements

➤ Client Side :-

- 1. Operating System :** Windows, macOS, Linux
- 2. Browser Support :** Chrome, Edge , etc (latest versions)

➤ Server Side :-

- 1. Operating System :** Windows, macOS, Linux
- 2. Browser Support :** Chrome, Edge , etc (latest versions)
- 3. Server Support :** Python Django Localhost
- 4. Database Support :** Python Django Sqlite

The Three Category of Users and Their Roles

Admin Category and Roles

| ROLE | ADMIN |
|-------------|---------------------|
| 1. | Login |
| 2. | Superuser |
| 3. | Add New Category |
| 4. | Update Order Status |

Buyer Category and Roles

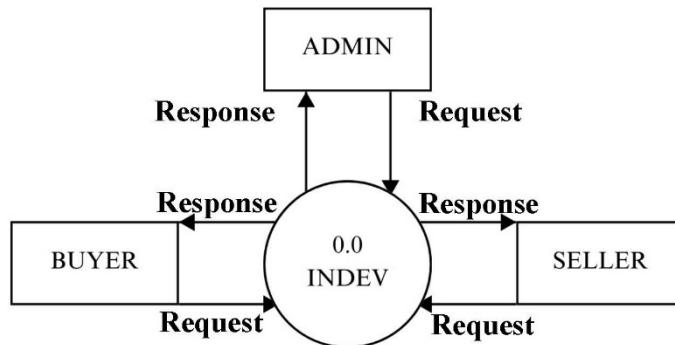
| ROLE | BUYER |
|-------------|---|
| 1. | Register/Login |
| 2. | View All Products, Shops and Categories View Their Cart and Orders |
| 3. | Add Remove Products in Cart Increase Decrease Quantity in Cart |
| 4. | Create Order Cancel Placed Order and Delete Delivered Order |

Seller Category and Roles

| ROLE | SELLER |
|-------------|--|
| 1. | Register/Login |
| 2. | View Their Products, Categories and Orders |
| 3. | Add Product, Update Stock and Delete Product |
| 4. | Accept or Reject Order |

Data Flow Diagram

Context Level DFD



❖Explanation of the Context Level DFD for INDEV

A **Context Level Data Flow Diagram (DFD)** provides a high-level overview of how data moves within the system, showing the interaction between external **entities (Admin, Buyer, Seller)** and the central system (**INDEV eCommerce Platform**).

❖ Entities and Their Interactions:

1. Admin:

- **Request:** Admin interacts with INDEV to manage categories, update order status, and oversee the system.
- **Response:** The system provides requested data and confirmation of actions performed.

2. Buyer:

- **Request:** Buyers send requests to the system for actions such as browsing products, adding/removing items in the cart, and placing/canceling orders.
- **Response:** INDEV responds with product listings, order updates, and cart modifications.

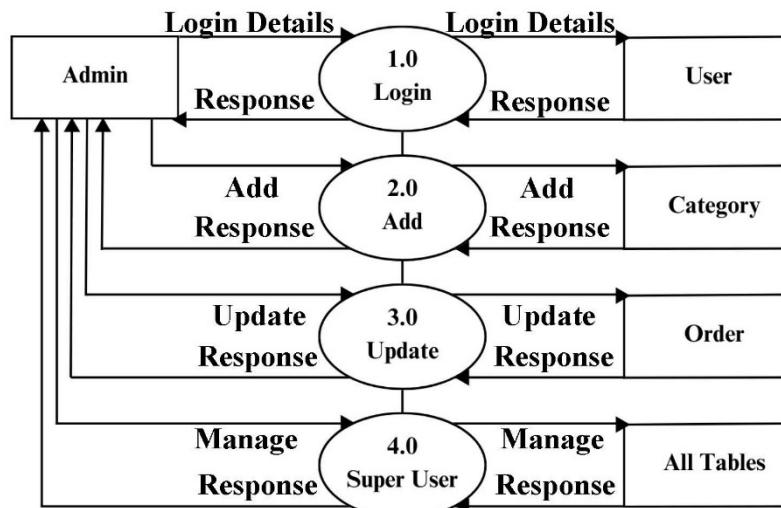
3. Seller:

- **Request:** Sellers submit requests to list new products, update product details, and accept/reject orders.
- **Response:** INDEV provides order status, product management feedback, and sales reports.

❖ Key Features of This Diagram:

- **The central circle (INDEV) represents the eCommerce system** that processes all user requests and responses.
- **Arrows indicate the direction of data flow**, showing how requests and responses travel between entities.
- **Simplified representation** of the entire system in a single diagram, without internal process details.

Admin Level DFD



❖Explanation of the Admin Level DFD for INDEV

This **Admin Level Data Flow Diagram (DFD)** provides a more detailed view of how the Admin interacts with the **INDEV system**, focusing on key administrative functionalities.

❖ Entities and Data Flow:

1. Admin (External Entity)

- The **Admin** is responsible for managing users, categories, orders, and system data.
- Interacts with different modules of the system for login, adding new data, updating records, and superuser access.

2. Processes (Circles in the DFD)

- **1.0 Login:** The admin logs in by entering credentials, and the system verifies the details, responding with authentication success or failure.
- **2.0 Add:** The admin can add new categories, and the system responds by updating the database and confirming the addition.
- **3.0 Update:** Admin updates existing order statuses (e.g., pending, shipped, delivered), and the system processes and reflects these updates.
- **4.0 Super User:** This process grants special access to manage all system data tables, allowing modifications at a higher level.

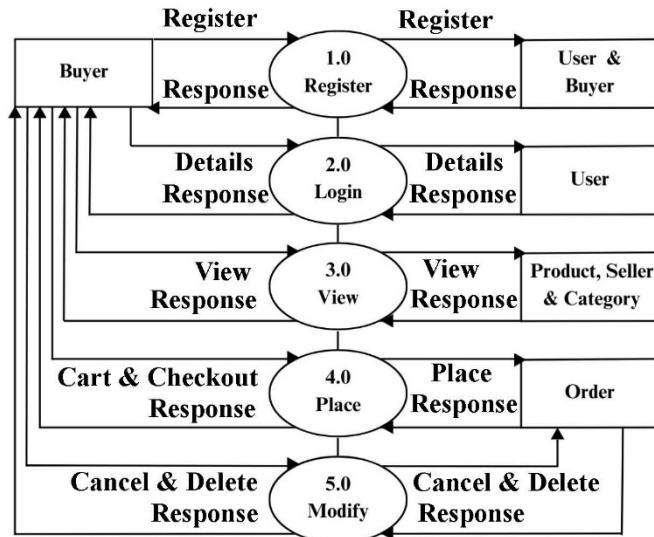
3. External Interactions (Models on the Right Side)

- **User:** Provides login details, and the system authenticates.
- **Category:** New categories added by the admin.
- **Order:** Receives updates related to orders.
- **All Tables:** The superuser manages all system tables, making changes where necessary.

❖ Key Takeaways:

- The Admin is the **central controller** of the system, managing categories, users, and orders.
- The **system responds to admin actions** by providing confirmation or updates.
- The **Super User role grants access to all tables**, ensuring complete administrative control.

Buyer Level DFD



❖Explanation of the Buyer Level DFD for INDEV

This **Buyer Level Data Flow Diagram (DFD)** shows how a buyer interacts with the **INDEV eCommerce system** and the data flow across different processes.

❖ Entities and Data Flow:

1. Buyer (External Entity)

- The **Buyer** interacts with the system by registering, logging in, viewing products, placing orders, and modifying orders.

2. Processes (Circles in the DFD)

- **1.0 Register:** The buyer provides registration details, and the system responds by storing the user data and confirming registration.
- **2.0 Login:** The buyer logs into the system by providing login credentials, and the system verifies and grants access.
- **3.0 View:** Buyers can view products, sellers, and categories, receive relevant data from the system.
- **4.0 Place Order:** Buyers add products to their cart, proceed to checkout, and place orders, with the system processing and storing order details.
- **5.0 Modify Order:** Buyers can cancel or delete their placed orders, and the system updates the order status accordingly.

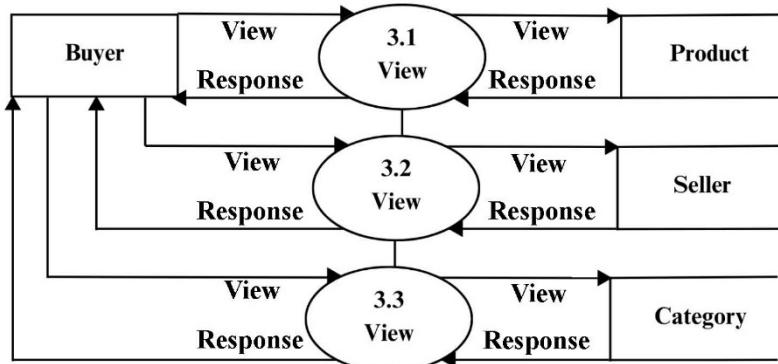
3. External Interactions (Models on the Right Side)

- **User & Buyer:** Stores registration and login details.
- **Product, Seller & Category:** Provides product listings and seller details.
- **Order:** Stores placed orders and modifications.

❖ Key Takeaways:

- Buyers follow a **structured process**, from registration to order placement and modification.
- The system **responds dynamically** to user actions, such as login verification, product listing, and order confirmation.
- Data flows seamlessly between **buyers and system components** for an efficient shopping experience.

Buyer Level 3.1 DFD



❖Explanation of Buyer Level 3.1 DFD

The **Buyer Level 3.1 Data Flow Diagram (DFD)** provides a detailed breakdown of the "View" process in the **INDEV eCommerce system**. It highlights how a buyer interacts with different elements of the marketplace.

❖Entities and Data Flow:

1. Buyer (External Entity)

- The **Buyer** initiates a request to view details in the system, which can be related to **Products, Sellers, or Categories**.

2. Processes (Circles in the DFD)

- **3.1 View Product:** The buyer sends a request to view product details, and the system fetches product data and sends a response.
- **3.2 View Seller:** The buyer requests information about sellers, and the system retrieves the seller data.
- **3.3 View Category:** The buyer requests product categories, and the system returns the relevant category details.

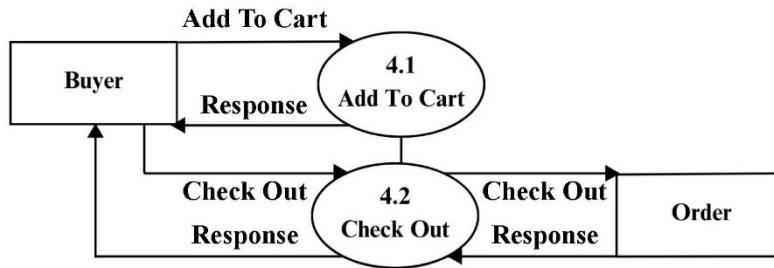
3. External Interactions (Models on the Right Side)

- **Product Database:** Stores product details.
- **Seller Database:** Contains information about registered sellers.
- **Category Database:** Organizes products into different categories for easy navigation.

❖ Key Insights:

- This DFD **decomposes the general "View" function** from the main Buyer Level DFD into specific **sub-processes** for products, sellers, and categories.
- Buyers **receive dynamic responses** based on their requests, allowing for an interactive shopping experience.
- The system **ensures quick retrieval** of relevant data to enhance user experience.

Buyer Level 4.1 DFD



❖Explanation of Buyer Level 4.1 DFD

The **Buyer Level 4.1 Data Flow Diagram (DFD)** represents the **shopping cart and checkout process** in the INDEV eCommerce system. It details how a buyer interacts with the system while adding items to the cart and placing an order.

❖Entities and Data Flow:

1. Buyer (External Entity)

- The buyer initiates two key actions:
 - **Add to Cart (4.1)**
 - **Check Out (4.2)**

2. Processes (Circles in the DFD)

- **4.1 Add To Cart:**

- The buyer selects a product and adds it to their shopping cart.
- The system updates the cart and provides a response (confirmation or error message).

- **4.2 Check Out:**

- The buyer proceeds to checkout after finalizing their cart.
- The system processes the order and sends a confirmation response.
- The final order is placed and stored in the system.

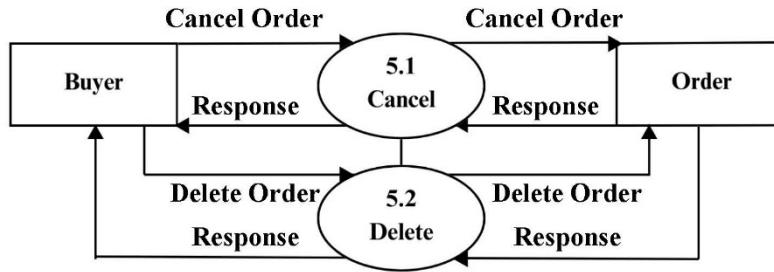
3. External Interactions (Entities on the Right Side)

- **Order System:** After checkout, the order details are stored in the system for further processing (such as payment and delivery).

❖ Key Insights:

- This **DFD models the buyer's shopping experience**, from adding products to the cart to finalizing a purchase.
- The **flow ensures a smooth transaction process**, guiding the buyer through essential steps.
- The **responses ensure real-time feedback**, such as confirming a successful cart update or checkout process.

Buyer Level 5.1 DFD



❖Explanation of Buyer Level 5.1 DFD

The **Buyer Level 5.1 Data Flow Diagram (DFD)** represents the **order cancellation and deletion process** in the INDEV eCommerce system. It outlines how a buyer interacts with the system when they want to cancel or delete an order.

❖Entities and Data Flow:

1. Buyer (External Entity)

- The buyer initiates two main actions:
 - **Cancel Order (5.1)**
 - **Delete Order (5.2)**

2. Processes (Circles in the DFD)

- **5.1 Cancel Order:**

- The buyer requests to cancel an order.
- The system processes the cancellation request and sends a response.
- The order status is updated accordingly.

- **5.2 Delete Order:**

- The buyer may choose to delete an order from their order history.
- The system processes the delete request and sends a confirmation response.

3. External Interactions (Entities on the Right Side)

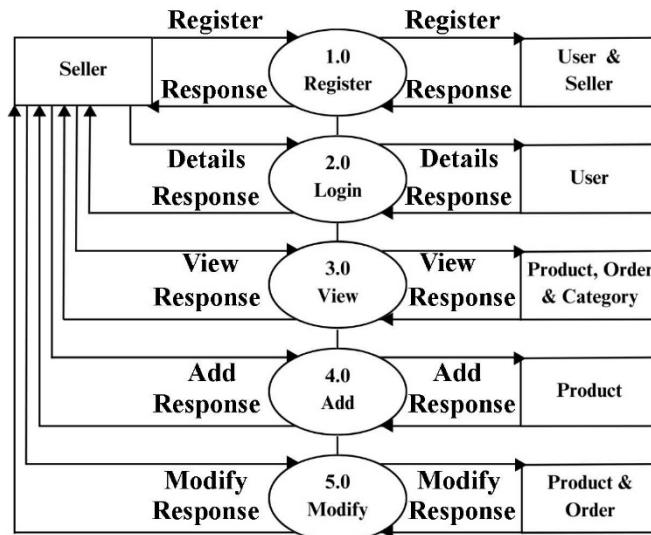
- **Order System:**

- Both **cancellation and deletion requests** interact with the **order database/system** to modify order records.

Key Insights:

- This **DFD models the post-purchase actions** available to buyers.
- **Order cancellation ensures flexibility**, allowing buyers to revoke an order before fulfilment.
- **Order deletion removes order records**, potentially for privacy or organization.
- The **response flow ensures feedback to the buyer**, such as confirming successful cancellation or deletion.

Seller Level DFD



❖Explanation of the Seller Level DFD for INDEV

This **Seller Level Data Flow Diagram (DFD)** illustrates how a seller interacts with the **INDEV eCommerce system**, handling products, order management, and modifications.

Entities and Data Flow:

1. Seller (External Entity)

- The **Seller** interacts with the system by registering, logging in, viewing product details, adding new products, and modifying existing listings or orders.

2. Processes (Circles in the DFD)

- **1.0 Register:** The seller provides registration details, and the system stores the details and confirms registration.
- **2.0 Login:** The seller logs in by providing credentials, system verifies access and retrieves seller details.
- **3.0 View:** The seller can view product categories, orders, and other relevant details. The system fetches data based on the seller's requests.
- **4.0 Add:** Sellers can add new products, and the system updates the product database accordingly.
- **5.0 Modify:** Sellers can update product details or manage orders, with the system reflecting these changes in the database.

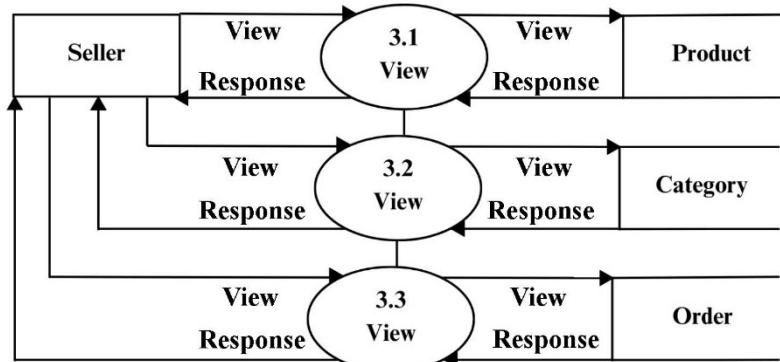
3. External Interactions (Entities on the Right Side)

- **User & Seller:** Stores registration and login details.
- **Product, Order & Category:** Provides sellers with relevant data for viewing and modifying listings.
- **Product & Order:** Stores newly added products and updated orders.

Key Takeaways:

- Sellers play a crucial role in maintaining **product availability** and managing orders.
- The system ensures **seamless data handling**, allowing sellers to add and modify products dynamically.
- The interaction between **sellers and the order system** keeps inventory and transactions up to date.

Seller Level 3.1 DFD



❖Explanation of Seller Level 3.1 DFD

The **Seller Level 3.1 Data Flow Diagram (DFD)** represents how a seller interacts with the system to view different types of information related to their business.

❖Entities and Data Flow:

1. Seller (External Entity)

- The **Buyer** initiates a request to view details in the system, which can be related to **Products, Categories, or Orders**.

2. Processes (Circles in the DFD)

- **3.1 View Product**

- The seller requests to view product details.
- The system retrieves product information and sends a response.

- **3.2 View Category**

- The seller requests to view categories of products.
- The system retrieves category information and sends a response.

- **3.3 View Order**

- The seller requests to view order details.
- The system retrieves order-related information and sends a response.

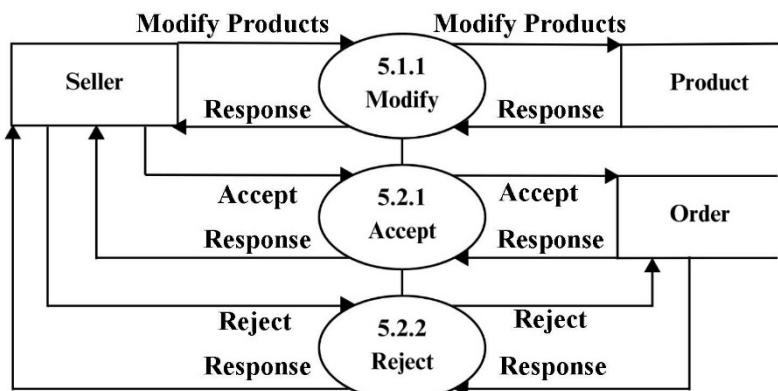
3. External Interactions (Models on the Right Side)

- **Product** : Stores product details.
- **Category** : Stores product categories.
- **Order** : Stores order information.

❖ Key Insights:

- This **DFD models the seller's ability to retrieve data** regarding their products, categories, and orders.
- **Efficient data retrieval is crucial for sellers** to manage their inventory and track sales.
- The **response flow ensures that sellers receive up-to-date information** when viewing different records.

Seller Level 5.1 DFD



❖Explanation of Seller Level 5.1 DFD

The **Seller Level 5.1 Data Flow Diagram (DFD)** represents how a seller interacts with the system to modify products and manage order approvals.

❖ Entities and Data Flow:

1. Seller (External Entity)

❖ The seller initiates three main actions:

- **Modify Product (5.1.1)**
- **Cancel Order (5.2.1)**
- **Delete Order (5.2.2)**

2. Processes (Circles in the DFD)

❖ 5.1.1 Modify Product

- The seller modifies product details (e.g., updating price, stock, description).
- The system updates the product and sends a response back.

❖ 5.2.1 Accept Order

- The seller accepts an order request.
- The system updates the order status and sends a response.

❖ 5.2.2 Reject Order

- The seller rejects an order request.
- The system updates the order status accordingly and sends a response.

3. External Interactions (Entities on the Right Side)

❖ **Product Database:** Stores and updates product details.

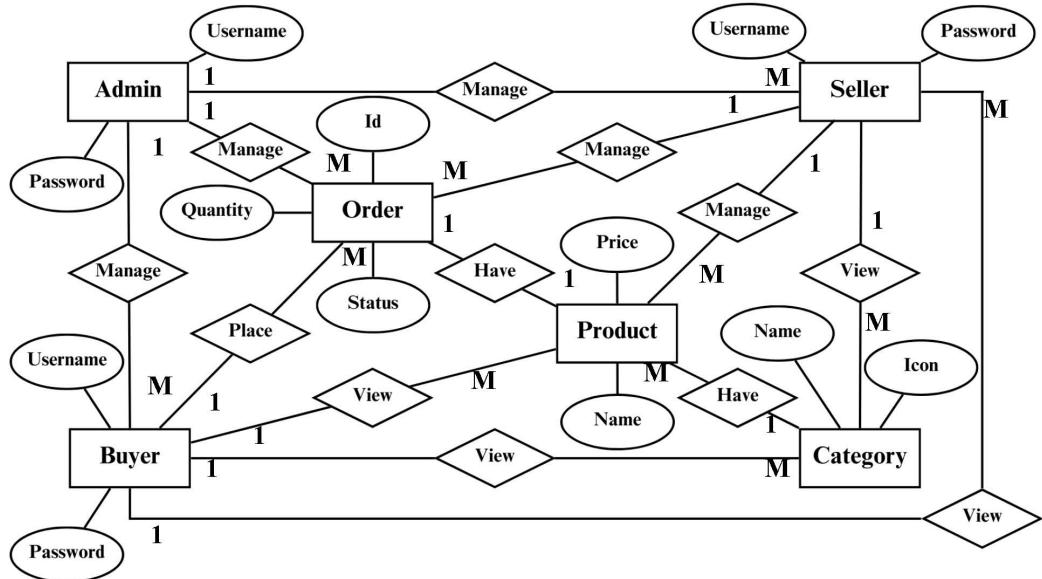
❖ **Order Database:** Manages order acceptance or rejection.

❖ Key Insights:

- This **DFD models the seller's ability to manage product listings** and handle customer orders.
- **Order acceptance and rejection** are crucial for managing inventory and fulfilment.
- **Product modifications** ensure that the latest information is available to buyers.

Entity Relationship Diagram

Entity Relationship Diagram



❖Explanation of the Entity Relationship Diagram (ERD)

This **Entity Relationship Diagram (ERD)** represents the structure and relationships within an eCommerce system. It includes key entities such as **Admin**, **Seller**, **Buyer**, **Order**, **Product**, and **Category**, along with their attributes and associations.

❖Key Entities and Their Relationships:

1. Admin

- Attributes: Username, Password.
- Relationships:
 - **Manages Sellers** (One-to-Many, 1:M).
 - **Manages Buyers** (One-to-Many, 1:M).
 - **Manages Orders** (One-to-Many, 1:M).

2. Seller

- Attributes: Username, Password.
- Relationships:
 - **Manages Products** (One-to-Many, 1:M).
 - **Manages Orders** (One-to-Many, 1:M).
 - **Views Categories** (Many-to-Many, M:M).

3. Buyer

- Attributes: Username, Password.
- Relationships:
 - **Places Orders** (One-to-Many, 1:M).
 - **Views Products, Sellers and Categories** (One-to-Many, 1:M).

4. Order

- Attributes: ID, Quantity, Status.
- Relationships:
 - **Placed by Buyers** (Many-to-One, M:1).
 - **Managed by Admin** (Many-to-One, M:1).
 - **Has Products** (One-to-One, 1:1).

5. Product

- Attributes: Name, Price.
- Relationships:
 - **Managed by Sellers** (Many-to-One, M:1).
 - **Belongs to Categories** (Many-to-One, 1:1).

6. Category

- Attributes: Name, Icon.
- Relationships:
 - **Has Multiple Products** (One-to-Many, 1:M).
 - **Viewed by Sellers & Buyers** (One-to-Many, 1:M).

❖ Key Takeaways:

- Admins have control over sellers and orders but do not directly manage products.
- Sellers manage products and view categories.
- Buyers place orders and view available products, sellers and categories.
- Orders contain multiple products, and products belong to multiple categories.
- This structure ensures proper separation of roles between Admins, Sellers, and Buyers.

Data Dictionary

Default User Model

| Attributes | Data Type | Constraints | For Example |
|-------------------|----------------------|--------------------------------|--------------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| Username | Character Field(150) | Unique , Not Null | admin |
| Password | Character Field(128) | Not Null , PBKDF2 with SHA-256 | ‘S@hil0209’ |
| First Name | Character Field(150) | Not Null , Default=“” | ADMIN |
| Staff Status | Boolean Field | Not Null , Default=False | True |

Buyer Model

| Attributes | Data Type | Constraints | For Example |
|------------|---------------------|------------------------------|--------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| New_User | Foreign Key(Users) | Unique , Not Null | Users(2) |
| First_Name | Character Field(50) | Not Null | Sahil |
| Last_Name | Character Field(50) | Not Null | Shaikh |
| Email | Email Field(50) | Not Null | s@gmail.com |
| Date | Date Field | Not Null | 02-09-2003 |
| Gender | Character Field(10) | Not Null | Male |
| Aadhaar | Character Field(12) | Not Null | 111122223333 |
| Contact | Character Field(10) | Not Null | 9999988888 |
| Address | Text Field(400) | Not Null | Mehsana |

Seller Model

| Attributes | Data Type | Constraints | For Example |
|--------------------|---------------------|------------------------------|----------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| New_User | Foreign Key(Users) | Unique , Not Null | Users(3) |
| Seller_Name | Character Field(50) | Not Null | Samsung |
| Seller_Description | Character Field(50) | Not Null | Mobiles |
| Logo | Image Field | Upload To , Not Null | 'logo/1.png' |
| Banner | Image Field | Upload To , Not Null | 'banner/1.png' |
| GSTIN | Character Field(15) | Not Null | 111122223333 |
| Email | Email Field(100) | Not Null | s@gmail.com |
| Contact | Character Field(10) | Not Null | 9999988888 |
| Address | Text Field(400) | Not Null | Mehsana |

Category Model

| Attributes | Data Type | Constraints | For Example |
|-----------------|---------------------|---------------------------------|--------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| Category_Name | Character Field(50) | Not Null | Mobiles |
| Category_Desc | Character Field(50) | Not Null | Phones |
| Category_Icon | Image Field | Upload To , Not Null | 'icon/1.png' |
| Category_Banner | Image Field | Upload To , Not Null | 'ban/1.png' |

Product Model

| Attributes | Data Type | Constraints | For Example |
|----------------|-----------------------|------------------------------|---------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| Seller_Name | Foreign Key(Seller) | Unique , Not Null | Seller(1) |
| Product_Name | Character Field(50) | Not Null | Jio Phone |
| Product_Desc | Character Field(50) | Not Null | It is a Phone |
| Product_Price | Integer Field | Not Null | 1000 |
| Product_Img | Image Field | Upload To , Not Null | ‘Prods/1.png’ |
| Product_Status | Character Field(50) | Choices , Default , Not Null | In Stock |
| Category_Name | Foreign Key(Category) | Unique , Not Null | Category(1) |

Order Model

| Attributes | Data Type | Constraints | For Example |
|------------------|----------------------|------------------------------|--------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| Buyer_ID | Foreign Key(Buyer) | Unique , Not Null | Buyer(1) |
| Seller_ID | Foreign Key(Seller) | Unique , Not Null | Seller(1) |
| Product_ID | Foreign Key(Product) | Unique , Not Null | Product(1) |
| Product_Quantity | Integer Field | Not Null | 5 |
| Product_Total | Integer Field | Not Null | 5 |
| Order_Status | Character Field(50) | Choices , Default , Not Null | Order Placed |

Contact Request Model

| Attributes | Data Type | Constraints | For Example |
|------------|----------------------|------------------------------|-----------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| Email | Email Field(50) | Not Null | sahil@gmail.com |
| Purpose | Character Field(500) | Not Null | Appointment |

Feedback Model

| Attributes | Data Type | Constraints | For Example |
|------------|----------------------|------------------------------|-----------------|
| ID | Integer Auto Field | Primary Key , Auto Increment | 1 |
| Email | Email Field(50) | Not Null | sahil@gmail.com |
| Feedback | Character Field(500) | Not Null | Good Work |

Technologies Used from Django Framework

1. Template Management and Template Inheritance –

Django framework provides in itself the feature of template management and also provides support for inheritance of one template into another.

2. Static Files and Media Files – Django framework also provides its efficient method of managing static files and media files by creating root directories.

3. Django Localhost Server – Django framework provides its own local server for testing and hosting the website for development and debugging.

4. Default User Model – Django framework provides a default user model which used to store data of all type of users.

5. Django Authentication – Django framework provides predefined function of Authenticate, Login and Logout which can be used to perform authentication of a user , it's login and logout activities.

6. Dynamic Url Routing – Django framework provides support to create and use dynamic urls for routing to webpages.

7. CSRF Token – CSRF stands for Cross Site Request Forgery , Django framework in all its forms uses a csrf token to ensure the integrity of the data passing via forms.

8. Django ORM – ORM stands for Object-Relational Mapping, Django framework uses object relational mapping to map each object member of a class as a field in the database. It provides easy, fast and efficient controls. We can fetch records faster.

9. Django Admin Interface – Django framework provides a default admin interface which can be used to create admins and super-admins. Admins are those with staff privileges and super-admins have all privileges of all tables.

10. Easy Integration with Other Python Modules – Django itself being a python module, it allows seamless integration with other python modules such as pillow and csv for better functionality in the overall projects.

Guest Side

Input Design

About Us

**Jaspreet Singh****Role : Frontend and Backend**[View on Github](#)**Shaikh Sahil****Role : Backend and Database**[View on Github](#)**Raval Viraj****Role : Database and Frontend**[View on Github](#)

Contact Us

Email :

Purpose :

[Submit](#)

Feedback

Email :

Feedback :

[Submit](#)

NEWEST

[About Us](#)[Login As Buyer](#)[Login As Seller](#)[Register As Buyer](#)[Register As Seller](#)

Buyer Login

Username:

Password:

[Login](#)

© 2025 All rights reserved.

Seller Login

Username:

Password:

Login

Buyer Registration

| | |
|-------------------|---|
| Create Username: | <input type="text"/> |
| Create Password: | <input type="password"/> |
| Confirm Password: | <input type="password"/> |
| First Name: | <input type="text"/> |
| Last Name: | <input type="text"/> |
| Date of Birth: | <input type="text"/> dd-mm-yyyy <input type="button" value=""/> |
| Gender: | <input type="button" value="Male"/> <input type="button" value=""/> |
| Aadhaar Number: | <input type="text"/> |
| Email: | <input type="text"/> |
| Contact: | <input type="text"/> |
| Address: | <input type="text"/> |

Seller Registration

| | |
|---------------------|---|
| Create Username: | <input type="text"/> |
| Create Password: | <input type="password"/> |
| Confirm Password: | <input type="password"/> |
| Seller Name: | <input type="text"/> |
| Seller Description: | <input type="text"/> |
| Seller Logo: | <input type="file"/> Choose File No file chosen |
| Seller Banner: | <input type="file"/> Choose File No file chosen |
| GSTIN Number: | <input type="text"/> |
| Email: | <input type="text"/> |
| Contact: | <input type="text"/> |
| Address: | <input type="text"/> |

Register

Buyer Side

Input Design

All Products**Samsung Smart Fridge**

Price : Rs. 100000

[View Product](#)[Add To Cart](#)**Tomatoes 10 kg**

Price : Rs. 500

[View Product](#)[Add To Cart](#)**Rolex Silver Watch**

Price : Rs. 51000

[View Product](#)[Add To Cart](#)**Gold Coin**

Price : Rs. 210000

[View Product](#)[Add To Cart](#)

Load Product**Samsung Galaxy S24 Ultra**[Add To Cart](#)**Product Details****Product Description :** It is a Smartphone**Product Price :** Rs. 125000**Product Category :** Mobiles**Seller Details****Sold By :** seller_samsung**Thanks for Visiting Us**

All Shops



Samsung

We Sell Consumer Electronics and Mobiles

[View Shop](#)

Reliance

We Sell Mobiles and Groceries

[View Shop](#)

Samsung



Samsung

We Sell Consumer Electronics and Mobiles

Our Details

GSTIN Number: 9999988888777777

Contact Us :

Email : samsung@gmail.com

Phone Number : 9999988888

Address : South Korea

Thanks for Visiting Us

Our Products



Samsung Smart Fridge

Price : Rs. 100000

[View Product](#)

[Add To Cart](#)



Samsung Washing Machine

Price : Rs. 95000

[View Product](#)

[Add To Cart](#)



Samsung Smart TV

Price : Rs. 105000

[View Product](#)

[Add To Cart](#)



Samsung Roomba

Price : Rs. 90000

[View Product](#)

[Add To Cart](#)

All Categories



Mobiles

For Example Smartphone, Feature Phone , Telephones

[View Category](#)

Groceries

For Example Soaps , Shampoos , Fruits

[View Category](#)

MOBILES



Mobiles

For Example Smartphone, Feature Phone ,
Telephones

Our Products



Samsung Galaxy S24 Ultra

Price : Rs. 125000

[View Product](#)

[Add To Cart](#)



Samsung Galaxy Z Flip

Price : Rs. 120000

[View Product](#)

[Add To Cart](#)



Samsung Galaxy Z Fold

Price : Rs. 130000

[View Product](#)

[Add To Cart](#)



Samsung Galaxy Tab

Price : Rs. 115000

[View Product](#)

[Add To Cart](#)

My Cart

| Sno. | Product Image | Product | Price | Quantity | Total |
|-------------|---|--------------------------|--------|---|--------|
| 1 |  | Samsung Smart TV | 105000 | 1 | 105000 |
| 2 |  | Samsung Galaxy S24 Ultra | 125000 | 2 | 250000 |
| Grand Total | | 355000 | | Check Out | |

My Orders

Order Number : 75

Buyer Details

Username : buyer_sahil
Name : Sahil Shaikh
Email : hshaikh1345@gmail.com
Contact : 9999955555
Address : 1-7-51 Maniyar Vado ,
Kapasiya Bazar , Sidhpuri Gali Road

Order Details

Name : Samsung Smart TV
Price : 105000
Quantity : 1
Total : 105000
Status : Ordered Placed

Cancel

Order Number : 76

Buyer Details

Username : buyer_sahil
Name : Sahil Shaikh
Email : hshaikh1345@gmail.com
Contact : 9999955555
Address : 1-7-51 Maniyar Vado ,
Kapasiya Bazar , Sidhpuri Gali Road

Order Details

Name : Samsung Galaxy S24 Ultra
Price : 125000
Quantity : 2
Total : 250000
Status : Delivered

Delete

Seller Side

Input Design

Listed Products

**Samsung Smart Fridge**

Price : Rs. 100000

[View Product](#)[Modify Product](#)**Samsung Washing Machine**

Price : Rs. 95000

[View Product](#)[Modify Product](#)**Samsung Smart TV**

Price : Rs. 105000

[View Product](#)[Modify Product](#)**Samsung Roomba**

Price : Rs. 90000

[View Product](#)[Modify Product](#)

My Categories



Mobiles

For Example Smartphone, Feature Phone , Telephones

[View Shop](#)

Consumer_Electronics

For Example TV , Fridge , Washing Machine

[View Shop](#)

Load Product**Samsung Smart TV**[Modify Product](#)**Product Details****Product Description :** It is a Smart TV**Product Price :** Rs. 105000**Product Category :** Consumer_Electronics**Seller Details****Sold By :** seller_samsung**Thanks for Visiting Us**

MOBILES



Mobiles

For Example Smartphone, Feature Phone ,
Telephones

Our Products



Samsung Galaxy S24 Ultra

Price : Rs. 125000

[View Product](#)

[Modify Product](#)



Samsung Galaxy Z Flip

Price : Rs. 120000

[View Product](#)

[Modify Product](#)



Samsung Galaxy Z Fold

Price : Rs. 130000

[View Product](#)

[Modify Product](#)



Samsung Galaxy Tab

Price : Rs. 115000

[View Product](#)

[Modify Product](#)

Product Registration

Seller Name:

Product Name:

Product Description:

Price:

Product Image:

 No file chosen

Category:

Register

Manage Products

| Sno. | Product Image | Product | Status | Set Status | Delete Product |
|------|---|-------------------------|--------------|------------------------------|------------------------|
| 1 |  | Samsung Smart Fridge | In Stock | Out of Stock | Delete |
| 2 |  | Samsung Washing Machine | Out of Stock | In Stock | Delete |

Reports



All Out of Stock Product Report

[Generate and Download Report In CSV Format](#)

All Accepted Orders Report

[Generate and Download Report In CSV Format](#)

Admin Side

Input Design

Django administration



Username:

Password:

Log in

Django administration

WELCOME, ADMIN. VIEW SITE / CHANGE PASSWORD / LOG OUT ☰

Site administration

AUTHENTICATION AND AUTHORIZATION

| | | |
|--------|-----------------------|------------------------|
| Groups | + Add | Change |
| Users | + Add | Change |

ECOMMERCE_WEBSITE_APPLICATION

| | | |
|------------------|-----------------------|------------------------|
| Buyers | + Add | Change |
| Categorys | + Add | Change |
| Contact_requests | + Add | Change |
| Feedbacks | + Add | Change |
| Orders | + Add | Change |
| Products | + Add | Change |
| Sellers | + Add | Change |

Recent actions

My actions

- [Order Id : 76
Order](#)
- [Rolex Diamond Watch
Product](#)
- [Premium Blue Jacket
product](#)
- [Traditional Kurta
product](#)
- [Golden Cufflinks
Product](#)
- [Rajshahi Dinning Set
product](#)
- [Non Stick Tawa
product](#)
- [Electric Toaster
product](#)
- [Heavy Necklace
product](#)
- [Analog Watch
product](#)

Future Enhancements

- 1. Payment Gateways** – Modern payment gateway systems such as bill desk and razor pay can be added to provide support for online transactions.
- 2. Add Sale Function** – Other functionalities such as creating a sale can be used to increase data traffic on the website.
- 3. Increase Filters** – The number of filters can be increased in the near future.
- 4. Review and Rating System** – A review system can be introduced to leave review on the purchased product by the buyer.
- 5. Profile Management** – Profile management will allow users(buyer & sellers) to manage their own profile records and details.
- 6. Increase Responsiveness** – The overall responsiveness of the website can be increased by using better animatronics and reactive CSS.
- 7. More Reports** – We can integrate other python modules like Matplotlib to make better info graphical and statistical reports

Bibliography

1. Django Reference – <https://docs.djangoproject.com/en/5.1/>

2. Learning – Geek for Geeks –

<https://www.geeksforgeeks.org/>

W3 Schools – <https://www.w3schools.com/django/index.php>

3. Debugging – Stack Overflow –

<https://stackoverflow.com/questions>

4. Other General Links – Google , YouTube , Wikipedia , GitHub , Udemy

5. General Utility Softwares – Canva , Word , Excel , PowerPoint

6. Chrome Extension – GoFullPage

7. Books Referred –



Python Basics.pdf



Django 4
Examples.pdf

Thank You