Four Weeks Industrial Training Project Report on

HexaClothing – E-Commerce Web Application Using Python & Django

Submitted in the partial fulfilment of the requirement for the award of degree of

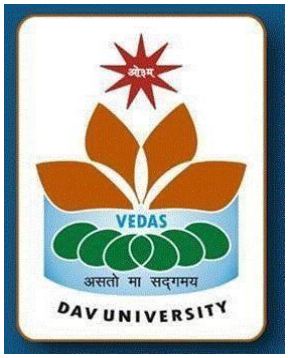
Bachelor of Technology

in

Computer Science and Engineering

Batch

(2023-2027)



Submitted to:

Dr. Ridhima

Submitted by:

Himanshu

12300428

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DAV UNIVERSITY

JALANDHAR -PUNJAB 144012

ABSTRACT

This report presents the development of **HexaClothing**, a fully functional e-commerce web application developed during my four-week industrial training using **Python, Django, HTML, CSS, JavaScript, and DbSql**.

The objective of the project was to understand real-world web development workflows, implement an MVC-based architecture, and build modules such as product catalog, cart handling, user authentication, payment integration, and admin management.

The training enhanced my understanding of backend development, database modeling, template rendering, CRUD operations, and deployment workflow. The project successfully meets the goals of responsiveness, usability, and structured data handling for a modern e-commerce platform.

CERTIFICATE

ACKNOWLEDGEMENT

I express my sincere gratitude to my Coordinator, **Dr. Ridhima**
,Department of Computer Science & Engineering, DAV University, for granting me the opportunity to undergo the 6-week industrial training. I am highly thankful for their support and continuous guidance throughout the preparation of this project and report.
I also extend my thanks to all the faculty members of the CSE Department for their constant encouragement and academic support.
Finally, I would like to thank the entire team at **TechCadd Computer Education**, Jalandhar Branch (Company/Institute Name) for providing me with valuable learning resources, hands-on experience, and support throughout my industrial training. Their guidance made this training highly productive and enriching.

DECLARATION

I,Himanshu , hereby declare that the training work titled **"HexaClothing – E-Commerce Web Application Using Python & Django"** is an original piece of work carried out by me during my industrial training under the guidance of **Mrs. Mamta**

To the best of my knowledge, this work has not been submitted to any other University or Institution for the award of any degree or diploma.

Student Name: Himanshu

Roll No: 12300428

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Overview of the Training

As part of the curriculum of B.Tech (CSE), students are required to undergo industrial training to gain practical exposure. I completed my **6-week industrial training** at **TechCadd Computer Education, Jalandhar Branch**, where I worked on developing an **e-commerce web application using Python and Django**.

1.2 Training Company

- **Name:** TechCadd Computer Education
- **Branch:** Jalandhar Branch
- **Domain:** Web Development / Software Development
- **Training Duration:** 6 Weeks
- **Training Period:** 25-06-2025 to 06-08-2025
- **Mode:** On-site

---

2. TITLE OF THE PROJECT

- **Project Title:**

  "HexaClothing – E-Commerce Web Application using Python & Django"

- **Project Description:**

  The project involved the design and development of a **full-stack e-commerce platform** using **Python, Django, HTML, CSS, Bootstrap, and SQLite (db.sqlite3)**. The system provides features such as **product browsing, user authentication, cart management, and secure checkout**, with emphasis on performance and usability.

3. Objectives

1. Training Area Objectives

- To acquire in-depth practical knowledge of **full-stack web development** using **Python and Django framework**.
- To gain exposure to **frontend–backend integration** in a real-world e-commerce environment.
- To understand how **database-driven web applications** are designed, developed, and maintained.
- To learn the implementation of **secure authentication and session-based access control**.
- To study the architecture and workflow involved in **commercial e-commerce platforms**.

---

2. Training Program Objectives

- To undergo structured industrial training in **Python programming, Django framework, HTML, CSS, Bootstrap, and SQLite (db.sqlite3)**.
- To understand the complete **Django development lifecycle**, including:
  - Project setup and configuration
  - App creation and modularization
  - URL routing and request handling
  - Model creation and database migration
  - Template rendering and static file management
- To learn implementation of **CRUD (Create, Read, Update, Delete) operations** using Django ORM.
- To gain practical exposure to **debugging, error handling, and performance optimization**.
- To understand the basics of **version control and code management using Git**.
- To learn concepts of **web application deployment, server configuration, and hosting basics**.
- To develop professional practices related to **coding standards, documentation, and project versioning**.

---

3. Tasks Assigned During Training

- Requirement analysis and functional understanding of the **HexaClothing e-commerce system**.
- Designing the **database schema** for users, products, categories, carts, and orders.
- Developing the **frontend interface** using HTML, CSS, and Bootstrap for:
  - Home page

- Product listing
- Product detail pages
- Login and registration pages
- Cart and checkout pages

- Implementing **backend functionality** using Django for:
  - User authentication and authorization
  - Product management (add, update, delete, view)
  - Category-wise product filtering
  - Shopping cart operations
  - Order placement and order history
- Integrating the **SQLite (db.sqlite3) database** with Django using ORM.
- Implementing **session management and security controls**.
- Performing **unit testing and functional testing** of all modules.
- Debugging runtime errors and logical flaws in the application.
- Preparing **documentation of system design, flow, and implementation**.

---

4. Professional Skill Development Objectives

- To improve **logical thinking and problem-solving skills** through real project development.
- To develop **team coordination and communication skills** in a training environment.
- To follow **industrial coding discipline and structured development practices**.
- To enhance confidence in **presenting and defending a technical project during viva**.

4. Steps to Achieve Objectives

To successfully accomplish the defined training and project objectives, the following systematic steps were followed during the training period:

---

1. Requirement Analysis and Problem Understanding

- Studied the basic working of **e-commerce systems** and online shopping workflows.
- Analyzed user requirements such as product browsing, cart management, user login, and order processing.
- Identified functional and non-functional requirements of the HexaClothing application.
- Prepared a clear understanding of input, process, and output of each module.

---

2. Learning of Core Technologies

- Learned **Python programming fundamentals** required for backend development.
- Studied the **Django framework architecture**, including:
  - Models
  - Views
  - Templates
  - URLs
- Gained practical knowledge of **HTML, CSS, and Bootstrap** for frontend design.
- Understood **SQLite (db.sqlite3) database concepts**, tables, keys, and relationships.

---

3. System Design and Architecture Planning

- Designed the **overall architecture** of the e-commerce system.
- Created the **database schema** for users, products, categories, cart, and orders.
- Planned the **module-wise structure** of the project.
- Designed basic **UI wireframes** for major pages such as home, product listing, login, and cart.

---

4. Environment Setup

- Installed and configured:
  - Python
  - Django framework
  - SQLite (db.sqlite3) database
  - Code editor (VS Code)
- Created Django project and application structure.
- Configured database connectivity between **Django and SQLite (db.sqlite3)**.

---

5. Frontend Development

- Designed user interface using **HTML, CSS, and Bootstrap**.
- Developed responsive web pages such as:
  - Home page
  - Product catalog
  - Product detail page
  - Login and registration pages
  - Cart and checkout pages
- Integrated frontend with Django templates.

---

6. Backend Development

- Created Django models for all database tables.
- Implemented **CRUD operations** using Django ORM.
- Developed backend logic for:
  - User authentication and authorization
  - Product management
  - Cart operations
  - Order placement
  - Session handling
- Implemented **form validations and error handling**.

---

7. Database Integration

- Implemented database migrations.
- Connected Django application with **SQLite (db.sqlite3) database**.
- Stored and retrieved user, product, cart, and order data securely.
- Ensured data consistency and integrity using relational design.

---

8. Testing and Debugging

- Performed **unit testing** for individual modules.
- Conducted **functional testing** for complete application flow.
- Identified and removed:
  - Runtime errors
  - Logical errors
  - Data handling issues
- Tested system performance under different input conditions.

---

9. Version Control and Code Management

- Used **Git** for:
  - Source code management
  - Tracking modifications
  - Maintaining project versions
- Followed systematic coding standards and modular structure.

---

10. Deployment and Execution

- Configured project for local/server execution.
- Tested application in real browser environment.
- Verified all features after deployment.

---

11. Documentation and Reporting

- Prepared technical documentation including:
  - System design
  - Database design
  - Module descriptions

- Flow diagrams
- Documented training activities and project progress.
- Prepared final project report for academic submission.

---

12. Evaluation and Final Submission

- Performed complete system validation.
- Corrected residual errors based on feedback.
- Prepared for **final demonstration and viva presentation**.

5. Coding and Implementation

The coding and implementation phase involved converting the system design into a fully functional web application using **Python and the Django framework**. This phase focused on developing the frontend interface, backend logic, database connectivity, and integrating all modules into a single operational system.

5.1 Development Environment

The following tools and technologies were used for coding and implementation:

- **Programming Language:** Python
- **Framework:** Django
- **Frontend Technologies:** HTML, CSS, Bootstrap
- **Database:** SQLite (db.sqlite3)
- **Code Editor:** Visual Studio Code
- **Web Browser:** Google Chrome
- **Operating System:** Windows

The Django environment was set up using a virtual environment to manage project dependencies efficiently.

5.2 Project Structure

The Django project was organized using a modular architecture to ensure scalability and maintainability.

- **Project Folder:** Main configuration and settings
- **App Folder:** Business logic and core functionalities
- **Templates Folder:** HTML pages
- **Static Folder:** CSS, JavaScript, and images
- **Media Folder:** Uploaded product images
- **Models:** Database table definitions
- **Views:** Backend logic
- **URLs:** URL routing

This structure helped in separating concerns and maintaining clean code organization.

5.3 Database Design and Implementation

The database was designed using **SQLite (db.sqlite3)** and integrated with Django using ORM (Object Relational Mapping).

Major tables implemented:

- User Table
- Product Table
- Category Table
- Cart Table
- Order Table

Each table was created as a Django model with appropriate fields, data types, primary keys, and foreign key relationships. Database migrations were used to apply schema changes to the SQLite (db.sqlite3) database.

5.4 Frontend Implementation

The frontend of the HexaClothing website was developed using **HTML, CSS, and Bootstrap** for responsive design.

Implemented user interfaces include:

- Home Page
- Product Listing Page
- Product Detail Page
- User Registration and Login Pages
- Cart Page
- Checkout Page
- Order Confirmation Page

Django templates were used to dynamically display data fetched from the database. Bootstrap ensured cross-device compatibility and consistent UI styling.

5.5 Backend Implementation

The backend logic was implemented using Django views and Python scripts.

Key functionalities coded:

- User authentication and authorization system
- Product management (add, edit, delete, view)
- Category-wise product display

- Shopping cart operations
- Order placement and management
- Session handling and security validation

Django's built-in authentication system was used to manage user accounts securely.

5.6 URL Routing and Navigation

URL routing was handled using Django's URL configuration system. All application URLs were mapped to their respective views to enable smooth navigation between different modules such as home, products, cart, login, and checkout.

5.7 Form Handling and Validation

User input was handled through Django forms. Server-side validation was implemented to ensure:

- Mandatory fields are filled
- Correct data types are entered
- Secure password handling
- Prevention of invalid or malicious inputs

Error messages were displayed to users for incorrect form submissions.

5.8 Image Upload and Media Handling

Product images were uploaded through the admin panel and stored in the media directory. Django media settings were configured to display product images on the website dynamically.

5.9 Security Implementation

Basic security measures were implemented including:

- Password encryption using Django's authentication system
- Session-based authentication
- User access control for admin and customers
- CSRF protection for form submissions

These measures helped in protecting user data and system integrity.

5.10 Testing During Implementation

Testing was performed continuously during development to ensure:

- Correct module integration
- Accurate database operations
- Proper UI responsiveness
- Error-free transaction flow

Both unit testing and functional testing methods were applied during this phase.

5.11 Final Integration

After individual module testing, all components were integrated into a complete system. The frontend, backend, and database were synchronized to ensure proper data flow across the application.

The final system was tested under real-time conditions for:

- User registration
- Product browsing
- Cart operations
- Order placement

All detected issues were debugged and resolved.

6. Conclusions and Recommendations

6.1 Conclusion

The successful completion of the **HexaClothing – E-Commerce Web Application using Python and Django** achieved all the defined training and technical objectives. The project provided extensive hands-on exposure to **full-stack web application development**, covering frontend design, backend logic, database integration, and system testing.

Through this project, a complete e-commerce workflow was implemented including **user authentication, product browsing, cart management, checkout process, and order handling**.

The application was developed using **Django framework**, which enabled rapid development, secure authentication, and efficient database handling using ORM.

This industrial training significantly enhanced practical understanding of:

- Real-world software development lifecycle
- Modular coding practices
- Database-driven web applications
- Error handling and debugging techniques
- Version control and structured development

The project not only strengthened technical competence but also improved professional skills such as **problem-solving ability, documentation, time management, and software development discipline**. The successful implementation of HexaClothing demonstrates the effective application of theoretical knowledge into a practical and functional system.

6.2 Recommendations

Although the current version of the HexaClothing application fulfills basic e-commerce requirements, several enhancements can be implemented to make the system more scalable, secure, and commercially usable:

1. **Online Payment Gateway Integration**

   Integration of payment methods such as UPI, debit/credit cards, and net banking will enable real-time online transactions.

2. **Admin Dashboard Enhancement**

   A detailed admin panel with sales analytics, revenue tracking, and inventory status can be added for better business control.

3. **Order Tracking System**

   Real-time order tracking and status updates (confirmed, shipped, delivered) can improve user experience.

4. **Product Reviews and Ratings**

   Customer feedback through reviews and ratings will increase user trust and product transparency.

5. **Wishlist and Recommendation System**

   Advanced features such as wishlists and AI-based product recommendations can enhance user engagement.

6. **Improved Security Features**

   Implementation of OTP verification, CAPTCHA, and advanced encryption will strengthen system security.

7. **Mobile Application Development**

   A mobile-friendly application using React Native or Flutter can be developed for wider accessibility.

8. **Cloud Deployment and Scalability**

   Deploying the application on cloud platforms will improve scalability, reliability, and performance under high traffic.

9. **Multi-Vendor Support**

   The platform can be extended to support multiple vendors with independent dashboards and inventory management.

10. **Performance Optimization**

    Database query optimization and caching mechanisms can be implemented to improve system response time

7. Appendices

The appendices section contains the **supporting material and technical references** related to the HexaClothing E-Commerce Web Application. These documents provide additional details that support the design, development, and implementation of the project.

Appendix A: Hardware Requirements

The minimum hardware specifications required to develop and run the application are as follows:

- **Processor:** Intel Core i3 or higher
- **RAM:** 4 GB minimum (8 GB recommended)
- **Hard Disk:** Minimum 20 GB free space
- **Display:** 1366 × 768 resolution or higher
- **Keyboard & Mouse:** Standard input devices
- **Internet Connection:** Required for installation and updates

---

Appendix B: Software Requirements

The software tools and platforms used in the project are:

- **Operating System:** Windows 10 / Windows 11
- **Programming Language:** Python 3.x
- **Web Framework:** Django
- **Frontend Technologies:** HTML, CSS, Bootstrap
- **Database:** SQLite (db.sqlite3)
- **IDE / Editor:** Visual Studio Code
- **Web Browser:** Google Chrome
- **Version Control:** Git

Appendix C: Project Modules

The major modules implemented in the project include:

- User Registration and Login Module
- Admin Management Module
- Product Management Module
- Shopping Cart Module

- Order and Checkout Module
- Session and Security Management Module

Each module works independently and is integrated through Django's URL routing and views.

Appendix D: Sample Code Structure

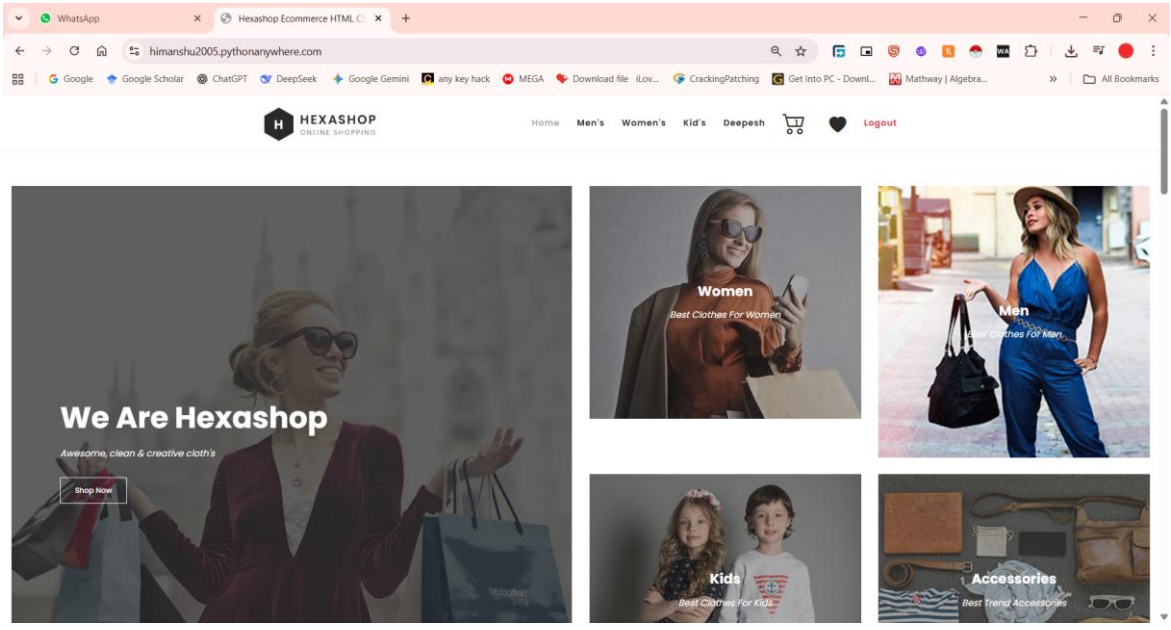
The project follows the standard Django directory structure:

- settings.py – Project configuration
- urls.py – URL routing
- models.py – Database models
- views.py – Business logic
- templates/ – Frontend HTML files
- static/ – CSS, JavaScript, images
- media/ – Uploaded product images

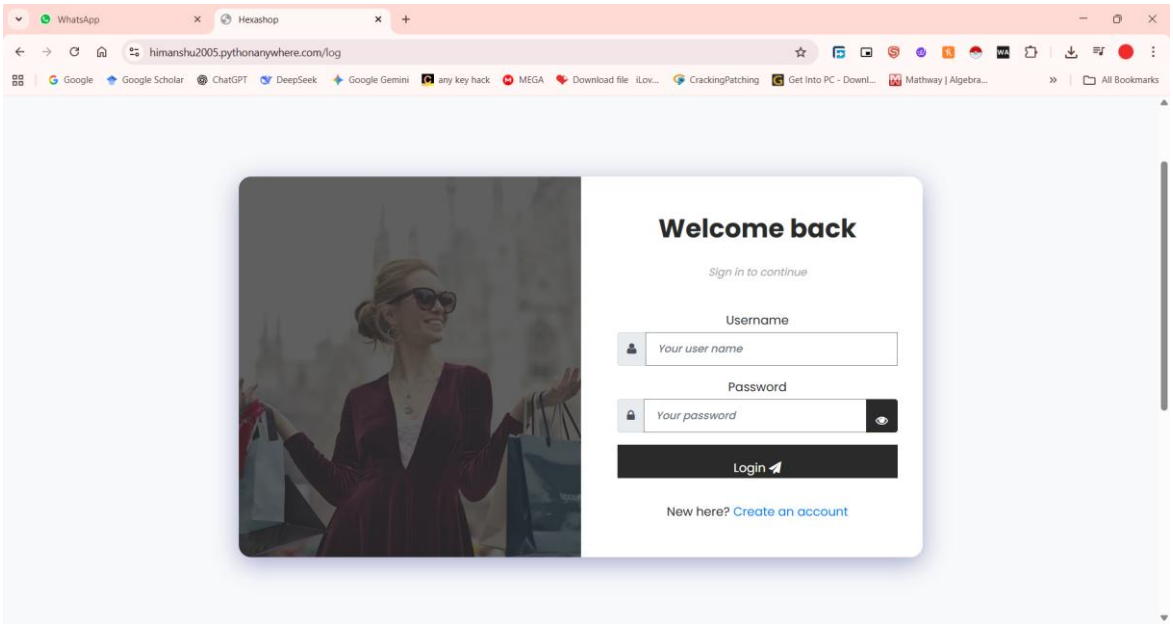
Appendix E: Screenshots List (If Included in Report)

Screenshots included in the project documentation:

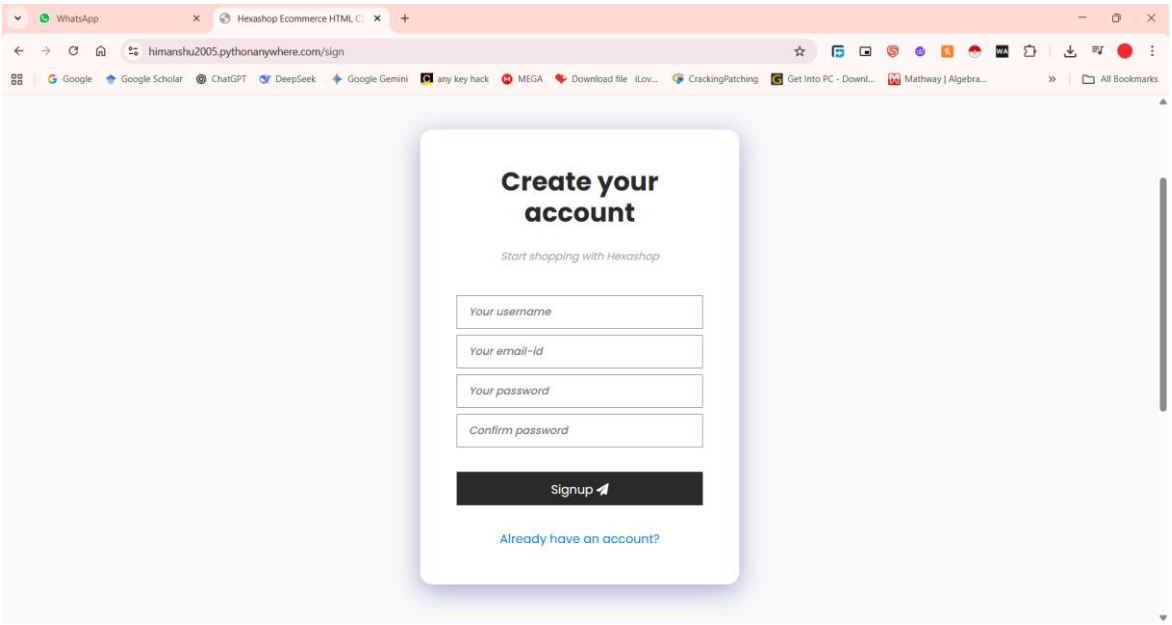
- Home Page



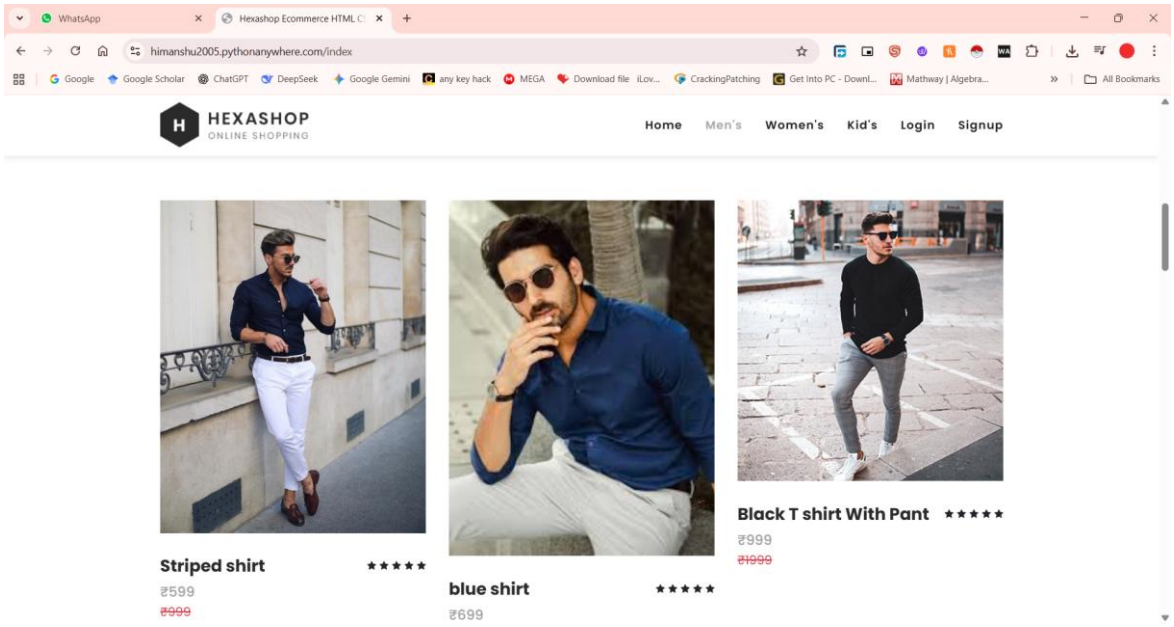
- User Login Page



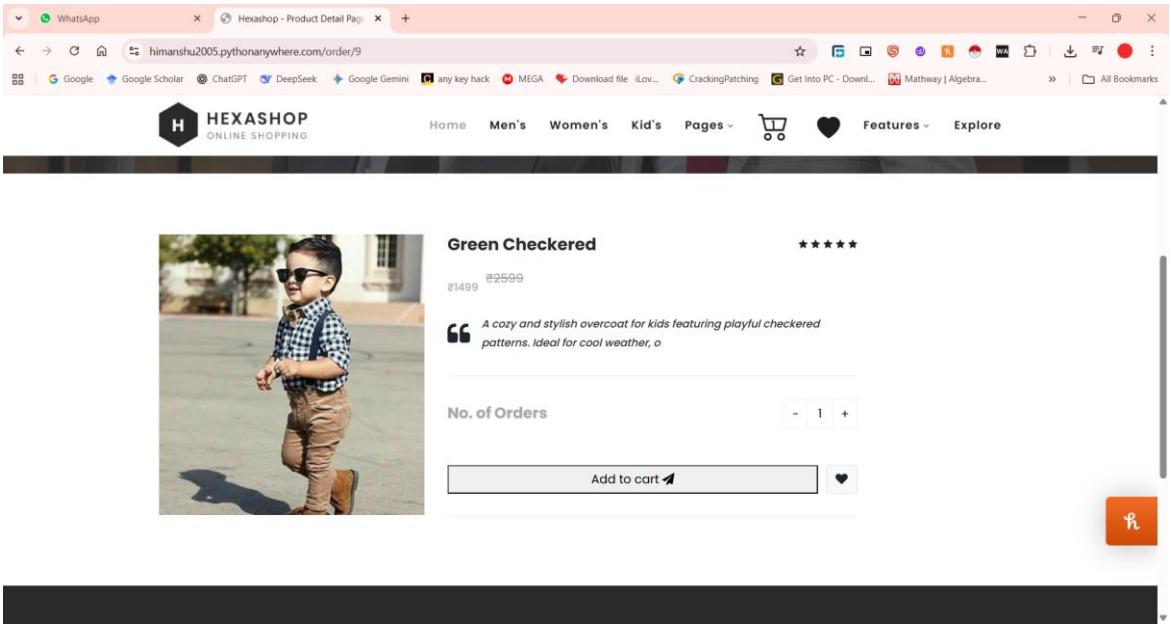
- Registration Page



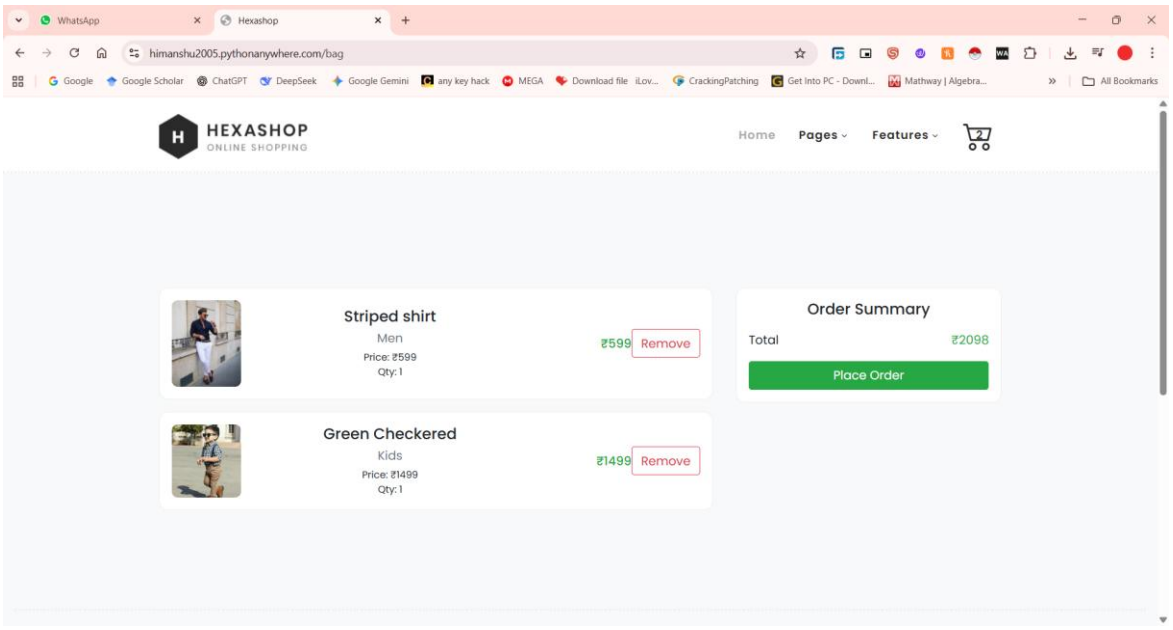
- Product Listing Page



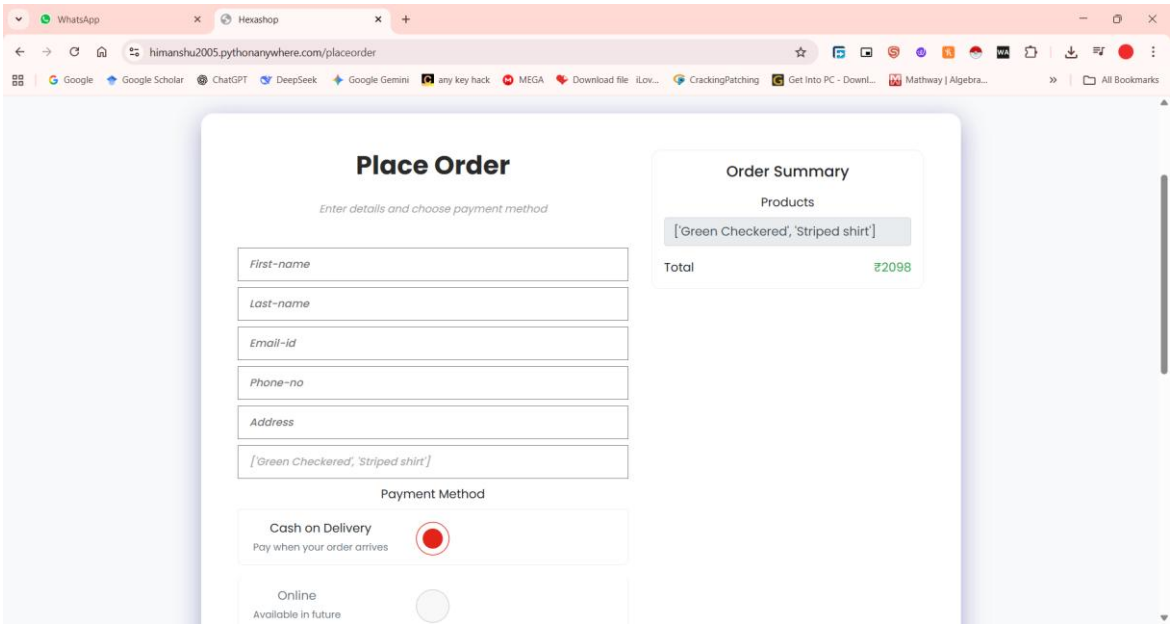
- Product Detail Page



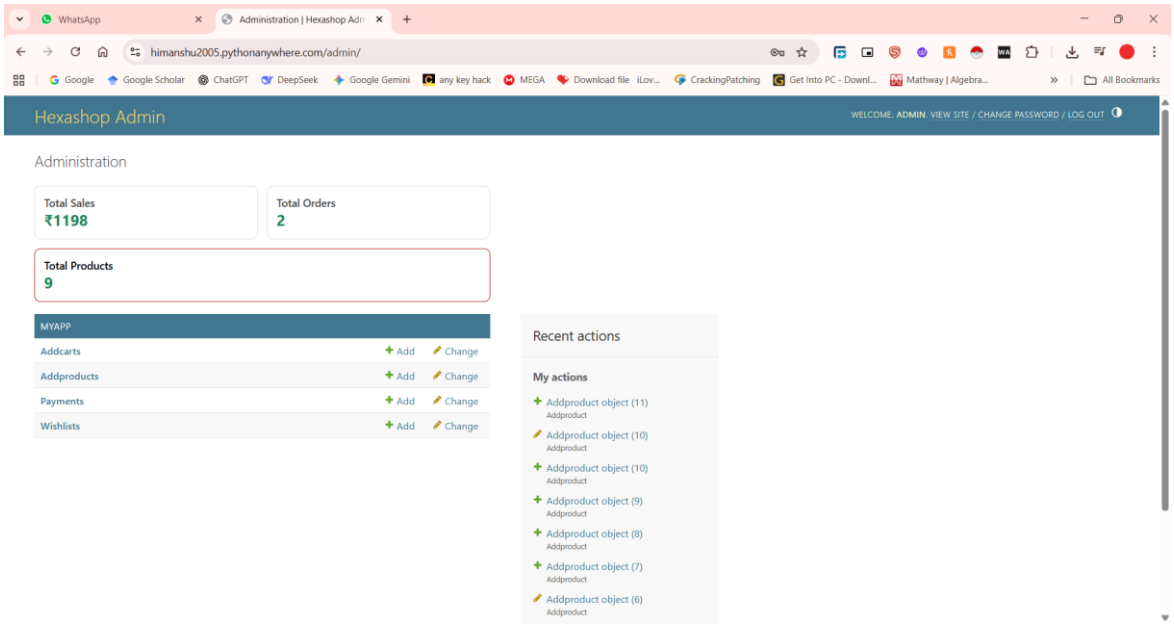
- Cart Page



- Checkout Page



- Admin Dashboard



These screenshots provide visual validation of the implemented system.

Appendix F: Bibliography / References

- Django Official Documentation
- Python Official Documentation
- SQLite (db.sqlite3) Documentation
- Bootstrap Documentation
- Online developer tutorials and reference materials