



Graphic Arts Institute

Department of Computer

Internship Report on Web Development with Python Django

Submitted to:

Sajib Bhawal

Lecturer

Department of CST

Submitted by:

MD Rifatuzzaman Riffat

Roll : 452048

Reg : 1502063844

.....
Lecturer's Signature

Django E-commerce Web Application

Table of Contents

Chapter 1	(1)
Introduction	1
1.1 Overview	1
1.2 Motivation	1
1.3 Objective	1
Chapter 2	(2)
Background Study Chapter	2
Chapter 3	(3-4)
Requirement Specification	3
3.1 Hardware Configuration	3
3.2 Software Configuration	3
3.3 Context Diagram and E-commerce Case Diagram	4
3.4 Relationship Diagram	4
Chapter 4	(5-10)
DESIGN SPECIFICATION	5
4.1 Introduction to Django	5
4.2 Customer Interface	8
4.3 Admin Interface	9
4.4 Project Planning and Scheduling	10
Chapter 5	(11)
Implementing and Testing	11
Chapter 6	(12-19)
Result	12
Chapter 7	(20-21)
Conclusion	20
Future Scope	21

Chapter 1

Introduction

1.1 Overview

Web Development with Python Django represents a powerful and streamlined approach to building dynamic web applications. Django's emphasis on rapid development, scalability, and security makes it a compelling framework for developers. With its clean syntax, modular design, and extensive community support, Django offers an efficient and effective solution for creating modern, feature-rich web applications. This report explores the key features, advantages, and real-world applications of Web Development with Python Django, providing insights into its significance in the ever-evolving field of web development.

1.2 Motivation

The motivation for focusing on Web Development with Python Django is driven by its proven effectiveness and efficiency in the realm of web development. Django's emphasis on rapid development, scalability through modularity, and built-in security features makes it an attractive choice. The developer-friendly syntax, strong community support, and versatile industry adoption further contribute to the framework's appeal. Choosing Django for web development aims to leverage its robust features, streamline the development process, and align with modern best practices in the dynamic landscape of web application creation.

1.3 Objective

The objective is to explore and understand the fundamental aspects of Web Development with Python Django. This includes gaining insights into the framework's rapid development capabilities, scalability features, and inherent security measures. The aim is to showcase how Python Django facilitates efficient and effective web application development, leveraging its clean syntax, modular structure, and the support of a vibrant developer community.

Chapter 2

Background Study

The project has been designed for Internet and Web Programming lab. So it has been needed huge amount of background studies. The project has been designed by HTML,CSS,Bootstrap,Python,Django framework and SQLite.

This document aims at defining overall software requirement for Ecommerce Website. Efforts have been made to define the requirements exhaustively. The final product will be having only features/functionalities mentioned in this document and assumptions for any additional functionality/feature should not be made by any of the parties involved in developing/testing/implementing/using this product.

This specification document describes the capabilities that will provide by the web based software application Ecommerce Website. It also state the various constraints by which the system will abide. The intended audience for this document are the development team, testing team and end users of the product.

The application will manage the information about various Ecommerce Website. The application will greatly simplify and speed up the preparation and management process. This report encapsulates not only the technical prowess of Web Development with Python Django but also the narrative of our journey. It is a story of challenges met, efficiencies gained, and a community embraced. Through sharing our experiences, we aim to contribute to the collective narrative of developers navigating the vast possibilities of web development with Python Django.

Chapter 3

Requirement Specification

Our project entails the development of a dynamic web application leveraging Python Django for seamless backend operations. The primary goal is to create an intuitive and scalable platform that addresses].

3.1 Hardware Configuration

Minimum Requirement

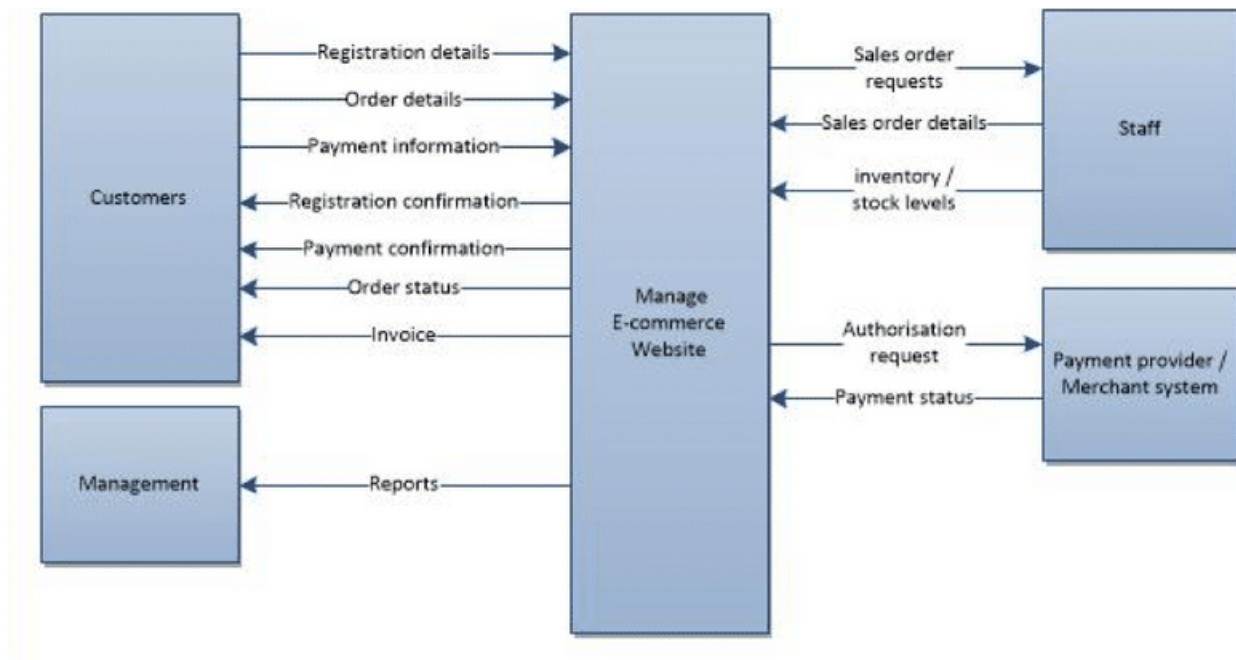
1. Dual Core Processor
2. 2GB RAM
3. 40 GB HDD
4. 1024*768 Resolution Color Monitor

3.2 Software Configuration

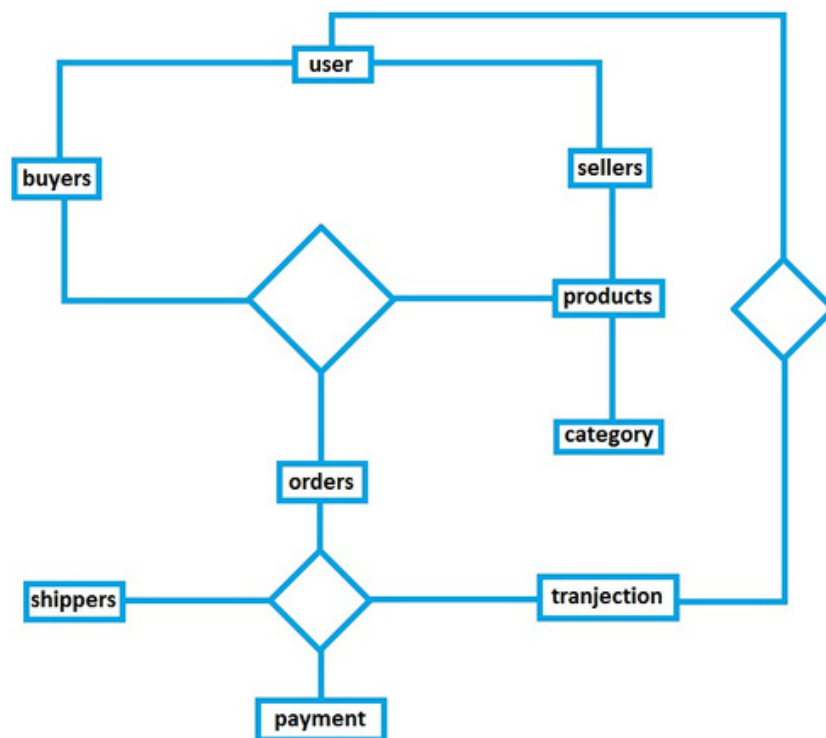
1. OS: Windows 10
2. Python 3 and Django
3. Vs Code, Sublime Text.

The software methodology followed in this project includes the object-oriented methodology and the application system development methodologies. The description of these methodologies is given below.

3.3 Context Diagram and E-commerce Case Diagram



3.4 Relationship Diagram



Chapter 4

DESIGN SPECIFICATION

4.1 Introduction to Django

Django, a high-level Python web framework, was first released in July 2005 by Adrian Holovaty and Simon Willison while working at the Lawrence Journal-World newspaper. The framework was created to streamline the development of web applications and alleviate common challenges faced by developers.

Backend Django

1. Models:

Django uses models to define the structure of the application's database. Models are Python classes that represent database tables, and Django's Object-Relational Mapping (ORM) system allows developers to interact with the database using Python code instead of raw SQL queries.

```
from django.db import models

class Product(models.Model):
    name = models.CharField(max_length=100)
    price = models.DecimalField(max_digits=10, decimal_places=2)
```

2. Views:

Views in Django handle the processing of requests and the generation of responses. Views are Python functions or classes that receive input from the user, interact with the database if necessary, and return a response, often in the form of rendered HTML templates.


```
from django.shortcuts import render
from .models import Product

def product_list(request):
    products = Product.objects.all()
    return render(request, 'products/product_list.html', {'products': products})
```

3. URL Routing:

URL routing maps URLs to specific views in Django. The `urls.py` file in each Django app defines the URL patterns and the corresponding views to be called when a particular URL is accessed.

```
from django.urls import path
from .views import product_list

urlpatterns = [
    path('products/', product_list, name='product_list'),
]
```

4. Middleware:

Middleware components in Django process requests and responses globally before reaching the views. They handle tasks such as authentication, security, and request/response modification.

5. Django Admin:

Django provides a powerful admin interface for managing data in the database. By registering models in the admin.py file, developers can easily perform CRUD operations through the admin interface.

```
from django.contrib import admin
from .models import Product

admin.site.register(Product)
```

6. Databases:

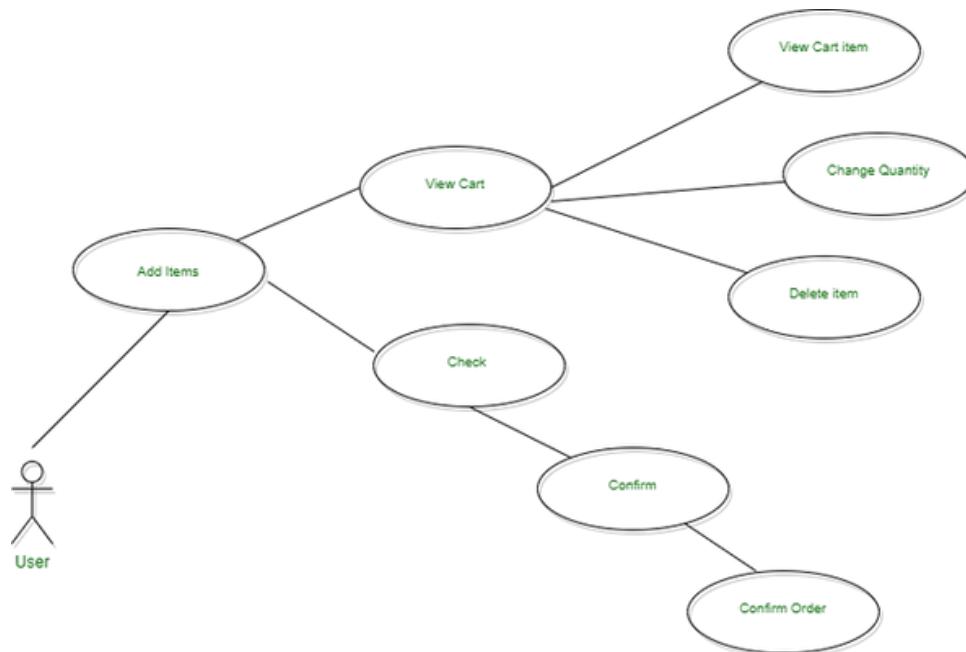
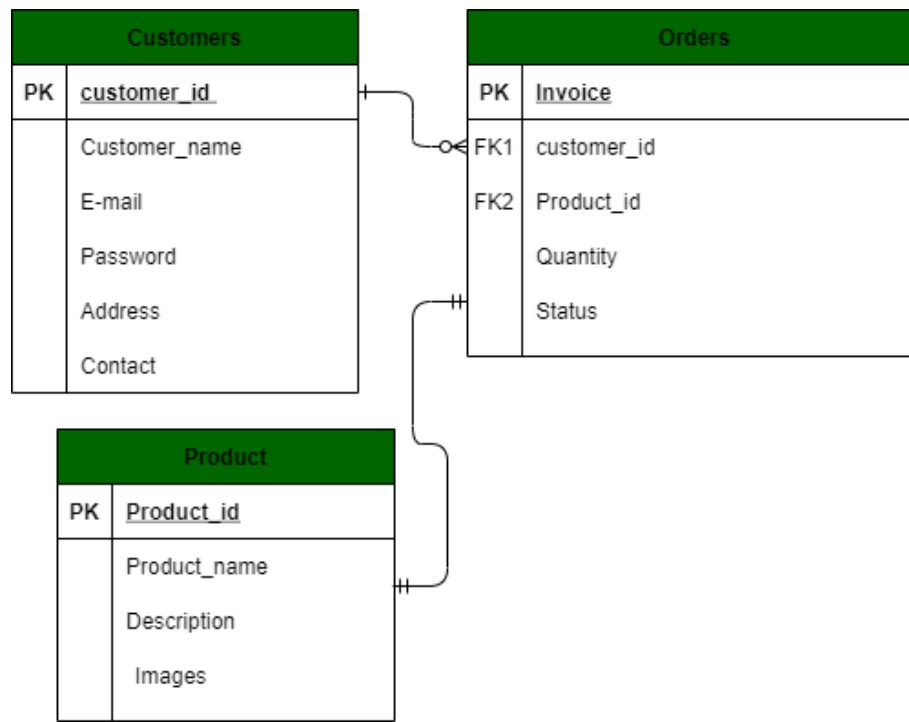
Django supports various databases, and the database configuration is specified in the settings.py file. The ORM system abstracts database interactions, allowing developers to work with models rather than writing raw SQL queries.

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / "db.sqlite3",
    }
}
```

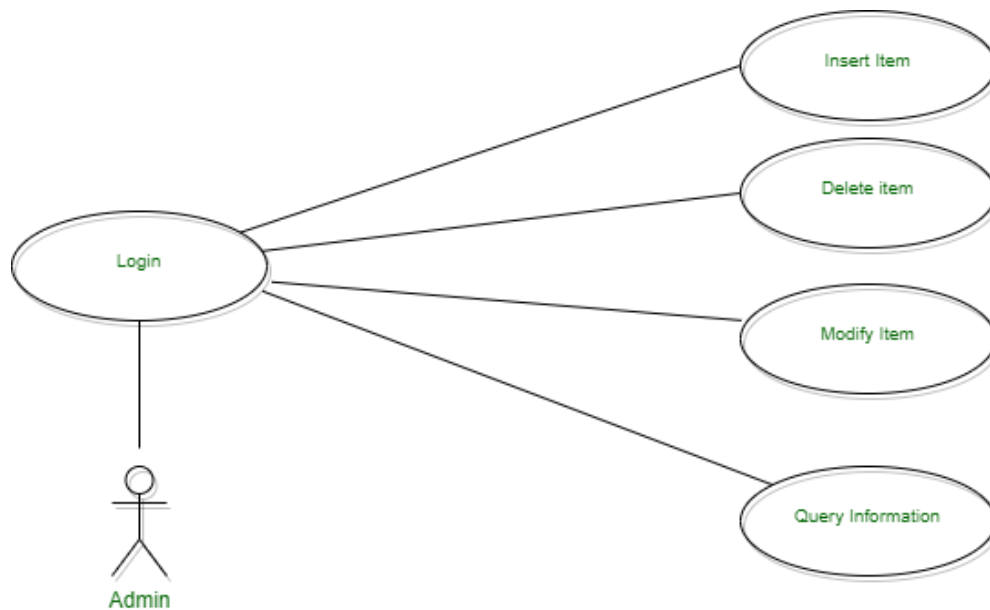
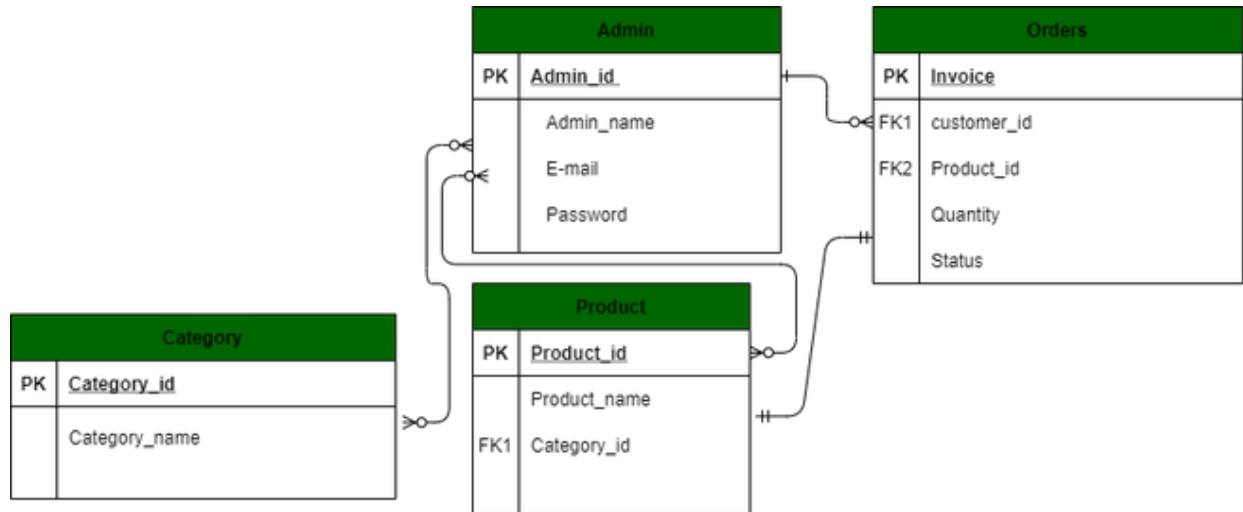
7. Authentication and Authorization:

Django includes built-in authentication and authorization systems. Developers can easily integrate user authentication and control access to views or resources based on user roles and permissions.

4.2 Customer Interfacer



4.3 Admin Interface



4.4 Project Planning and Scheduling

	Task	Start Date	Duration	End Date	
	----- -	-----	-----	-----	
	Project Initiation	2023	5 days	2023	
	Database Design	2023	10 days	2023	
	Models and ORM Implementation	2023	14 days	2023	
	API Design and Development	2023	21 days	2023	
	Authentication System	2023	10 days	2023	
	Product Management API	2023	14 days	2023	
	Shopping Cart API Integration	2023	7 days	2023	
	Order Processing API	2023	14 days	2023	
	Payment Gateway Integration	2023	14 days	2023	
	Backend Testing and Debugging	2023	14 days	2023	
	Documentation and Code Review	2023	7 days	2023	
	Deployment and Optimization	2023	7 days	2023	
	Post-Launch Support and Review	2023	7 days	2023	

Chapter 5

IMPLEMENTING AND TESTING

5.1 Implementing of Database

Testing

Testing is a process of executing a program with the indents of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. This a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

Testing Objectives:

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error
3. A successful test is one that uncovers an undiscovered error

Testing Principles:

1. All tests should be traceable to end user requirements
2. Tests should be planned long before testing begins
3. Testing should begin on a small scale and progress towards testing in large
4. Exhaustive testing is not possible
5. To be most effective testing should be conducted by an independent third party. The primary objective for test case design is to derive a set of tests that has the highest livelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are
 - White box testing.
 - Black box testing.

Chapter 6

Result

After going through all the phase of System Development Life Cycle of this project, the portal is designed successfully.

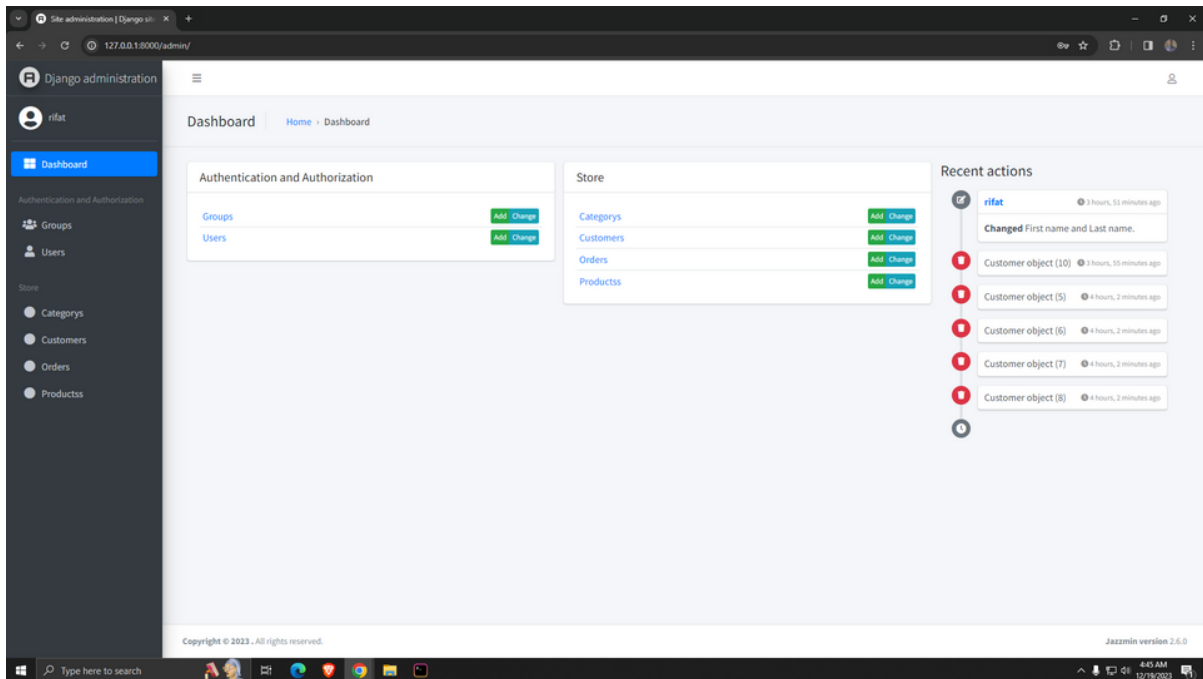
The result of the proposed project is demonstrated here by some screen shots as images.

6.1 For Admin

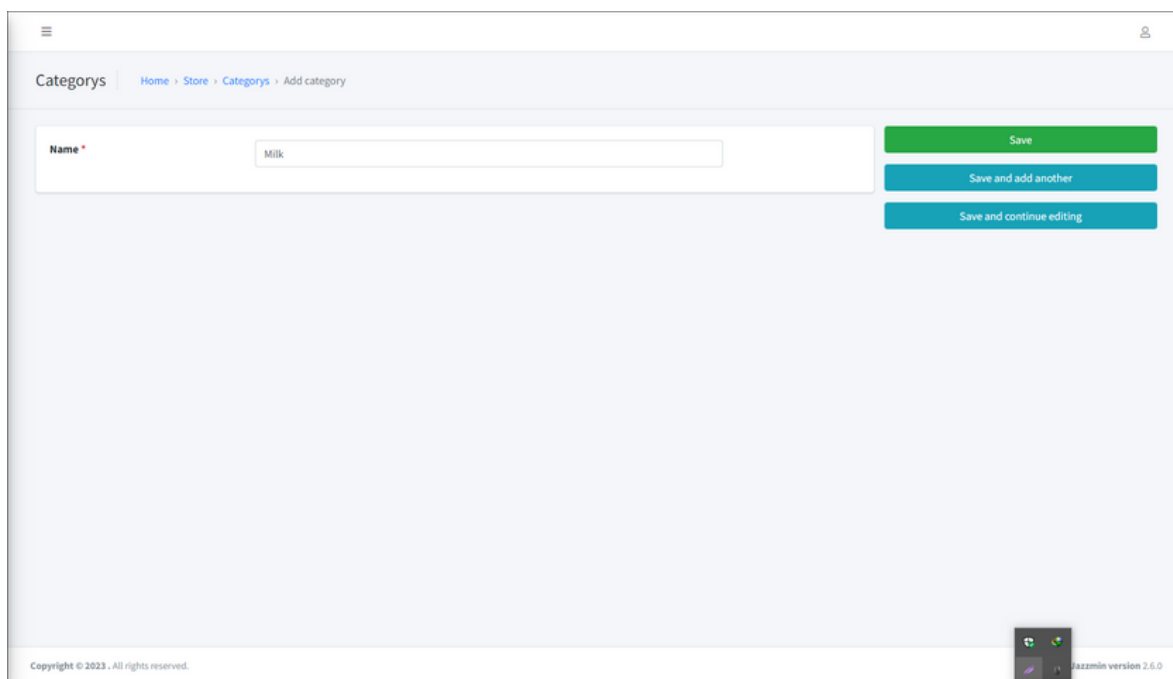


A login form titled "Welcome" is displayed. It features two input fields: "Username" with a user icon on the right, and "Password" with a lock icon on the right. Below these fields is a blue "Login" button.

After login , enter the admin panel is as follows:



From admin panel one can select an individual panel to insert page is as follows:



Customers

Home > Store > Customers > Add customer

First name *

Rifatuzzaman

Last name *

Rifat

Phone *

1234567890

Email *

rifat@gmail.com

Password *

rifat1122

Save

Save and add another

Save and continue editing

Copyright © 2023 . All rights reserved. Jazzmin version 2.6.0

Orders

Home > Store > Orders > Add order

Product *

Products object (57)

Customer *

Customer object (11)

Quantity *

1

Price *

500

Address

Bashundhara, Dhaka-1229

Phone

12345678901

Date *

2023-12-19

Today

Note: You are 6 hours ahead of server time.

Status

Save

Save and add another

Save and continue editing

Copyright © 2023 . All rights reserved. Jazzmin version 2.6.0

Page 14

Productss

[Home](#) > [Store](#) > [Productss](#) > Add products

Name *

milk

Price *

100

Category *

Women-Clothing

Description

good

Image *

Choose File

No file chosen

Save

Save and add another

Save and continue editing

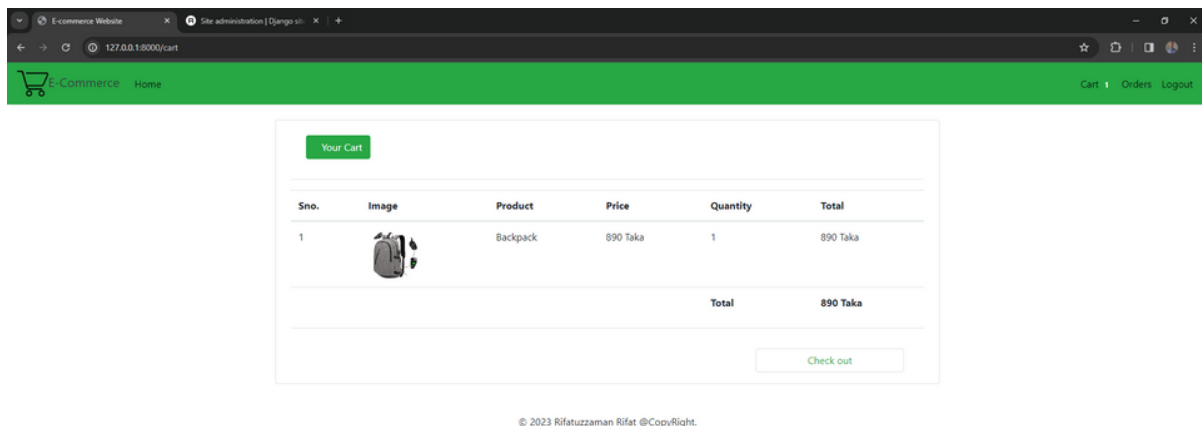
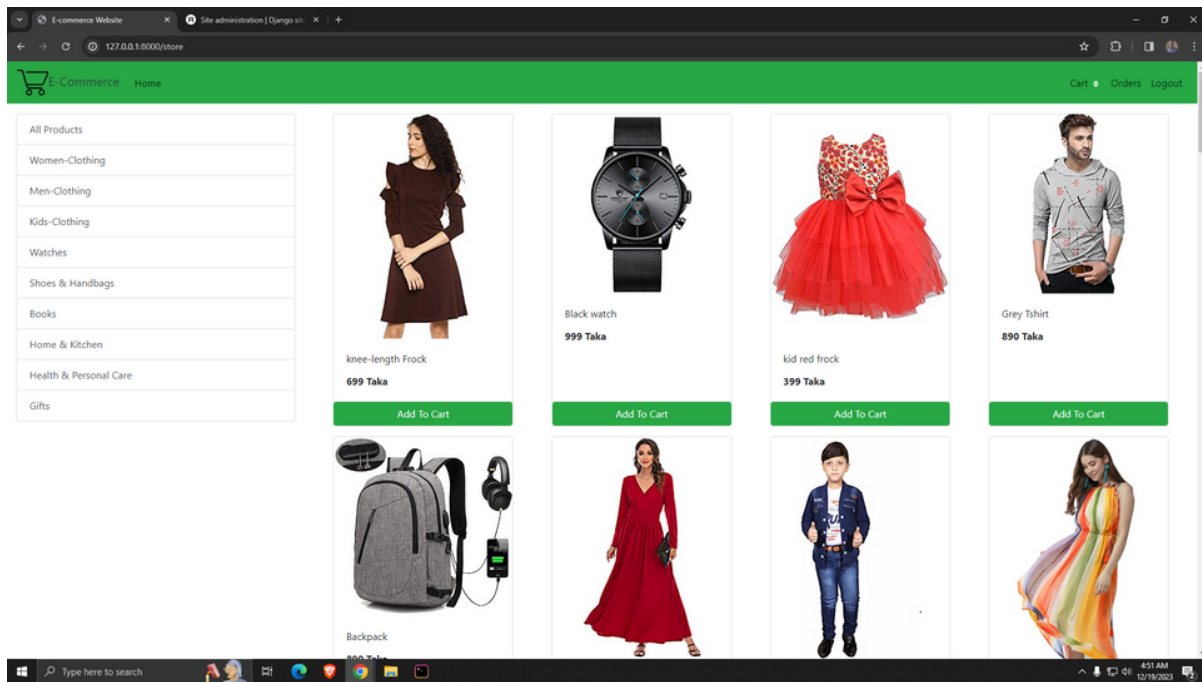
Copyright © 2023 . All rights reserved.

Jazzmin version 2.6.0

Page 16

5.2 For user

If an user enter this website can see the following page. From this webpage he can select various page for shopping




E-commerce Website

Site administration | Django 3.1

127.0.0.1:8000/orders





☆

 E-Commerce

Home


CartOrdersLogout

Your Orders

Sno.	Image	Product	Date	Price	Quantity	Total	Status
1		knee-length Frock	Dec. 19, 2023	699 Taka	1	699 Taka	Pending
2		Black watch	Dec. 19, 2023	999 Taka	1	999 Taka	Pending
3		kid red frock	Dec. 19, 2023	399 Taka	1	399 Taka	Pending
4		Grey Tshirt	Dec. 19, 2023	890 Taka	1	890 Taka	Pending

© 2023 Rifatuzzaman Rifat @CopyRight.

Type here to search

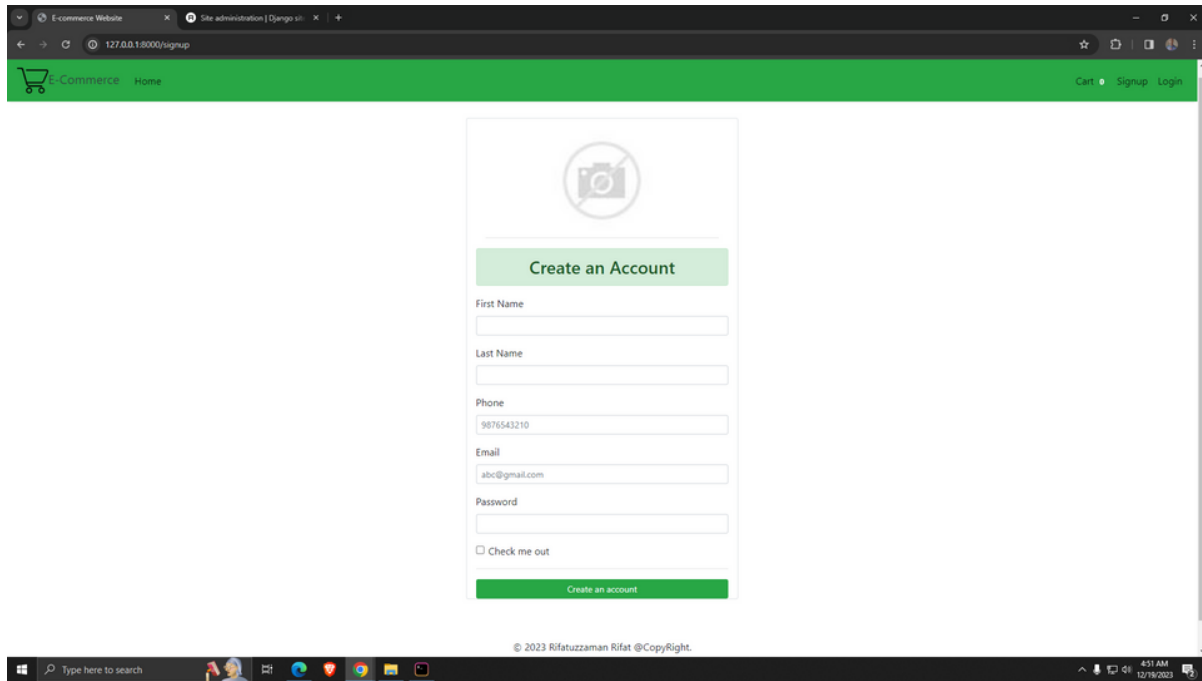


4:51 AM

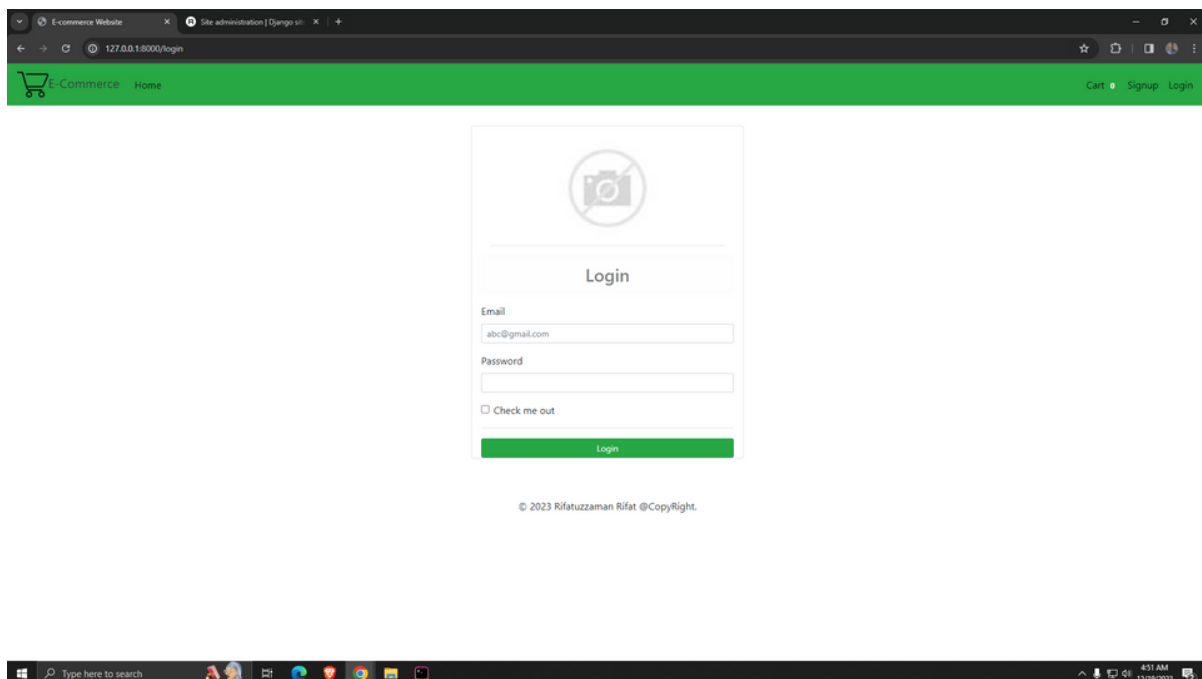
12/19/2023

5.2 For user login and signup

To create a new account and login user account :



The screenshot shows a web browser window with the URL `127.0.0.1:8000/signup`. The page has a green header with a shopping cart icon, the text "E-Commerce", a "Home" link, and "Cart", "Signup", and "Login" links. The main content area features a "Create an Account" form with a camera icon placeholder. The form includes input fields for "First Name", "Last Name", "Phone" (with the value "9876543210"), "Email" (with the value "abc@gmail.com"), and "Password". There is a checkbox for "Check me out" and a green "Create an account" button. The footer contains the copyright notice "© 2023 Rifatuzzaman Rifat @CopyRight."



The screenshot shows a web browser window with the URL `127.0.0.1:8000/login`. The page has a green header with a shopping cart icon, the text "E-Commerce", a "Home" link, and "Cart", "Signup", and "Login" links. The main content area features a "Login" form with a camera icon placeholder. The form includes input fields for "Email" (with the value "abc@gmail.com") and "Password". There is a checkbox for "Check me out" and a green "Login" button. The footer contains the copyright notice "© 2023 Rifatuzzaman Rifat @CopyRight."

Chapter 7

Conclusion

In conclusion, the journey through web development with Python Django has been both insightful and rewarding. Django's robust framework, coupled with its emphasis on rapid development and security, has provided a solid foundation for creating dynamic and scalable web applications. The experience underscores the importance of a collaborative community, efficient development practices, and the continuous pursuit of innovation in the ever-evolving landscape of web development. As this chapter concludes, the knowledge gained sets the stage for future endeavors, fueled by a commitment to excellence in building powerful and secure web solutions.

To assist individuals in web development with Python Django, consider actively engaging in community forums and answering queries to troubleshoot problems. Share concise tutorials on platforms like Medium, providing practical insights for learners. Offer mentorship to guide newcomers through challenges and contribute to open-source projects such as Django, fixing bugs, or enhancing features. I love to do this and in future i will flow this path as my career.

Future Scope

In the realm of web development with Python Django, my personal future scope is geared towards specialization and versatility. I aim to deepen my expertise in full-stack development, mastering both front-end technologies and Django's powerful back-end capabilities. Exploring emerging trends like microservices architecture, serverless computing, and Progressive Web Applications (PWAs) aligns with my goal of staying at the forefront of industry advancements. Additionally, I aspire to contribute actively to the Django open source community, collaborate on innovative projects, and potentially explore entrepreneurial ventures. Embracing continuous learning, networking, and mentorship opportunities will be pivotal in navigating the dynamic landscape of web development.