**OLABISI ONABANJO UNIVERSITY**

**P.M.B 2002, AGO IWOYE, OGUN STATE**

**MEMBERS MATRIC NO**

1. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
2. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
3. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
4. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
5. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
6. **AGBIRIGBA CHIDERA VALENTINE**   **SCI/21/22/0145**
7. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
8. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
9. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
10. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
11. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
12. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
13. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**
14. **AGBIRIGBA CHIDERA VALENTINE SCI/21/22/0145**

**DEPARTMENT: COMPUTER SCIENCE**

**GROUP: 7**

**COURSE CODE: CMP 434**

**TOPIC: E-COMMERCE WESBITE**

**CMP 434 Group Project**

**Project Title: ShopSwift – A Simple E-commerce Website**

**1. Introduction**

ShopSwift is a basic e-commerce website developed using Django. It allows users to register, log in, browse products, add items to a shopping cart, and complete purchases through an integrated payment system. The platform also includes an admin dashboard to manage products and orders.

**2. Tools and Technologies Used**

* **Backend & Templating:** Django (Python)
* **Frontend:** HTML, CSS, JavaScript
* **Database:** SQLite (for development), PostgreSQL (for production)
* **Version Control:** Git & GitHub
* **Hosting:** Render
* **Payment Gateway:** Flutterwave
* **Design Tools:** Figma (for wireframes)

**3. Team Members and Contributions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sub Group** | **Names** | **Matric Numbers** | **Contribution** |
| 1 | John Doe | CMP/20/0001 | Designed the layout and UI using Bootstrap. |
| 2 | Mary James | CMP/20/0002 | Created Django templates for homepage, product pages, and cart. |
| 3 | Peter Adams | CMP/20/0003 | Built models for Users, Products, Orders, and integrated admin panel. |
| 4 | Linda George | CMP/20/0004 | Managed database setup and product uploads via Django admin. |
| 5 | Alex Chuka | CMP/20/0005 | Integrated Flutterwave payment into the checkout page. |
| 6 | Blessing Okoro | CMP/20/0006 | Developed cart and order summary views using Django views and sessions. |
| 7 | Ibrahim Musa | CMP/20/0007 | Tested views, forms, and payment responses. |

To ensure effective collaboration, our group of 14 was divided into **7 subgroups (pairs)**, each assigned a specific responsibility aligned with their strengths and interest areas.

**Group 1: UI/UX Design and Frontend Layout**

**Members:** John Doe, Mary James  
**Contribution:**  
This group was responsible for designing the layout and structure of the website. They created wireframes for all the pages (home, product listing, cart, and checkout) using Figma, ensuring the flow was intuitive and user-friendly. They also handled the styling of the pages using Bootstrap, focusing on responsiveness and clean design.

**Group 2: Backend Models and Database Design**

**Members:** Peter Adams, Linda George  
**Contribution:**  
They built the Django models to define the structure of the database. These models included Product, CartItem, Order, and UserProfile. They handled relationships between models (e.g., orders connected to users, products in carts) and ensured database integrity. They also ran migrations and tested data flow across the app.

**Group 3: HTML Template Development**

**Members:** Alex Chuka, Blessing Okoro  
**Contribution:**  
This group converted the UI designs into actual Django HTML templates. They created the reusable base.html and extended it in other pages like home.html, product\_detail.html, and cart.html. They ensured templates were properly linked with Django views and that data passed from the backend was correctly displayed.

**Group 4: Cart and Order View Logic**

**Members:** Ibrahim Musa, Grace Ade  
**Contribution:**  
They implemented the logic behind adding items to the cart, updating quantities, removing items, and placing orders. They wrote Django views and forms to manage user interactions with the cart. They also made sure cart data was stored in sessions and synchronized with the database at checkout.

**Group 5: Project Integration and URL Routing**

**Members:** Emmanuel Nduka, [Your Name]  
**Contribution:**  
This group oversaw the overall project structure and served as the bridge between different features. They set up URL configurations for each app and integrated all views and templates into a cohesive flow. They also managed settings for static files, security, and environment variables needed for deployment.

**Group 6: Admin Panel Setup and Content Management**

**Members:** Joy Umeh, David Obinna  
**Contribution:**  
They customized the Django admin interface to allow easy product and order management. They registered all relevant models and used Django's admin features to display product images, order details, and user information. They also populated the site with demo product entries for testing and presentation.

**Group 7: Payment Integration and Documentation**

**Members:** Vivian Eze, Samuel Dan  
**Contribution:**  
This group integrated Flutterwave as the payment processor, setting up test keys and implementing secure payment callbacks. They handled payment verification and ensured that successful payments led to proper order confirmation. Additionally, they compiled the documentation for the entire project, including team contributions, steps taken, and challenges faced.

**4. Implementation Steps**

**Step 1: Planning and Design**

* Drew wireframes for main pages: Home, Product Detail, Cart, Checkout, Admin Dashboard.
* Decided on site structure and features.

**Step 2: Project Setup**

* Created Django project and app using django-admin startproject and startapp.
* Set up static files and templates folders.
* Connected to the database.

**Step 3: Building the Backend**

* Defined models for Users, Products, Orders, and Cart Items.
* Used Django admin to manage products.
* Created forms for user registration, login, and checkout.

**Step 4: Frontend Development**

* Used HTML, CSS, and Bootstrap for page layouts.
* Created reusable base template (base.html) for consistency.
* Rendered product listings dynamically using Django template tags.

**Step 5: Cart and Checkout**

* Implemented shopping cart using Django sessions.
* Developed views for adding/removing items, and viewing cart summary.

**Step 6: Payment Integration**

* Integrated Flutterwave API in the checkout process.
* Handled transaction success and failure callbacks.

**Step 7: Testing and Debugging**

* Tested all forms, views, and payment process.
* Fixed bugs and adjusted styling for mobile responsiveness.

**Step 8: Deployment**

* Pushed project to GitHub.
* Deployed to Render.
* Set environment variables (e.g. SECRET\_KEY, database URL) for production.

**5. Challenges Faced**

* Handling sessions properly for cart functionality.
* Styling pages consistently using Bootstrap.
* Managing test payments with Flutterwave.
* Coordinating tasks remotely among group members.

**6. Conclusion**

This project gave us hands-on experience with full-stack Django development. We learned how to build and deploy a functional e-commerce platform using Django’s powerful templating system and backend tools.