1/12/2025

***GROUP 2,***

***UNIVERSITY OF GHANA-LEGON,***

***DEPARTMENT OF COMPUTER SCIENCE,***

***BSC. INFORMATION TECHNOLOGY,***

***E-BUSINESS ARCHITECTURES(DCIT209)***

***12TH JANUARY 2025***

**SEMESTER PROJECT DOCUMENTATION**

A FULL-FUNCTIONAL E-COMMERCE WEBSITE FOR A MONOPOLISTIC MARKET (NUDUDU RESTAURANT)

**GROUP MEMBERS:**

1.GYASI KWADWO AMOS-22245133

2.NTREH MEONOTHAI NII-22146400

3.AGBOVIE OFOTSU-KOBLA EMMANUEL -22175047

4.KUEWU PAMELA-22048845

5.KANYIRI ELLIPIO-22019535

6.TEPKER KWAKU JOSEPH-22014263

7.ADJEI JOEL-22044650

8.APRAKU ROONEY TAWIAH-22223752

9.ADAMS MICHELLE AFUA-22039724

10.NABIIRIBONG DAVID-11296202

11.PURITY ABENA KYEI-11215590

12. VICAL AGHEC DIVINE-22250179

**OVERVIEW**

The project has been designed within a monopolistic market framework. The website showcases a restaurant and food delivery service named 'Nududu Restaurant,' providing a comprehensive range of food and services exclusively from this establishment. This project is such that, the restaurant, which is monopolistic in this context, have the ability to display their menu and other services they render on the app where any other user of the website, being an authenticated user or non- authenticated user can view the displayed menu with their corresponding prices. Meanwhile, Any user that wants to place and order for a product must be an authenticated user. This was implemented for easy identification of users in terms of delivery. This will also help trace payments in cases where there`s an interrupted payments methods.

**GROUP COMPOSITION**

Our group is composed of 11 dedicated members. To enhance our effectiveness and encourage full participation, each member has been assigned specific tasks. Under the leadership of Project Manager Agbovie Ofotsu-Kobla Emmanuel, responsibilities have been distributed among the team members, as detailed in the table below.

|  |  |  |
| --- | --- | --- |
| INDEX | NAME | ASSIGNED TASK/TASKS |
| 1 | AGBOVIE OFOTSU-KOBLA EMMANUEL | Project Manager, Tester |
| 2 | KUEWU PAMELA | Database Designer/Analyst, Tester |
| 3 | NABIIRIBONG DAVID | Database Designer/Analyst |
| 4 | TEPKER KWAKU JOSEPH | Frontend Developer |
| 5 | KANYIRI ELLIPIO | Frontend Developer |
| 6 | APRAKU ROONEY TAWIAH | UI/UX Designer |
| 7 | ADAMS MICHELLE AFUA | Frontend Developer |
| 8 | PURITY ABENA KYEI | UI/UX Designer/Tester |
| 9 | NTREH MEONOTHAI NII | System Analyst, Tester |
| 10 | ADJEI JOEL | Frontend Developer, System Analyst |
| 11 | GYASI KWADWO AMOS | Group Leader, Backend Developer |
| 12 | VICAL AGHE DIVINE | UI/UX, Tester |

A group of people posing for a photo

Description automatically generated

**MARKET TYPE:**

The market type chosen for this project is the Monopolistic type. This website showcases an environment controlled by the producer, in this case, a restaurant that seeks multiple buyers. Buyers can access the menu provided by the restaurant, but the prices are solely determined by the restaurant administrator. Buyers have no influence over pricing. The producer has the ability to adjust prices or change the quantity produced to impact profits. Discounts may be offered on certain purchases at specific times. Since the power resides with the seller (the restaurant), both value and authority remain within the market.

**TECHNOLOGIES USED**

To develop a fully functional e-commerce website tailored to our selected market, we employed a range of advanced technologies. For the database design, we utilized DRAW.IO, an effective tool for visualizing database schemas, known for its user-friendly interface and ergonomic features. The actual implementation of the database was carried out with SQLite3, a decision made due to its simplicity and compatibility with the other technologies integrated into the system.

For the UI/UX design, we chose FIGMA, a free design software that provides a wide array of tools for creating user-cantered interfaces. The implementation of the UI/UX designs was executed using HTML, CSS, and JavaScript. To optimize the handling of HTTP requests, we selected React.js as our framework.

On the backend, we opted for Python Flask, a lightweight microframework based on the Python programming language. This choice was influenced by its status as one of the most widely utilized Python frameworks for backend development and its extensive library support for functionalities such as user authentication.

For research and troubleshooting, Stack Overflow served as a critical resource. This platform facilitates interaction between experienced developers and newcomers, providing valuable insights on programming and technical matters. Additionally, resources such as YouTube, Google, and Amazon assisted in broadening our understanding of the chosen market and enhancing the project's overall development.

**WEBSITE FEATURES:**

***HOMEPAGE***

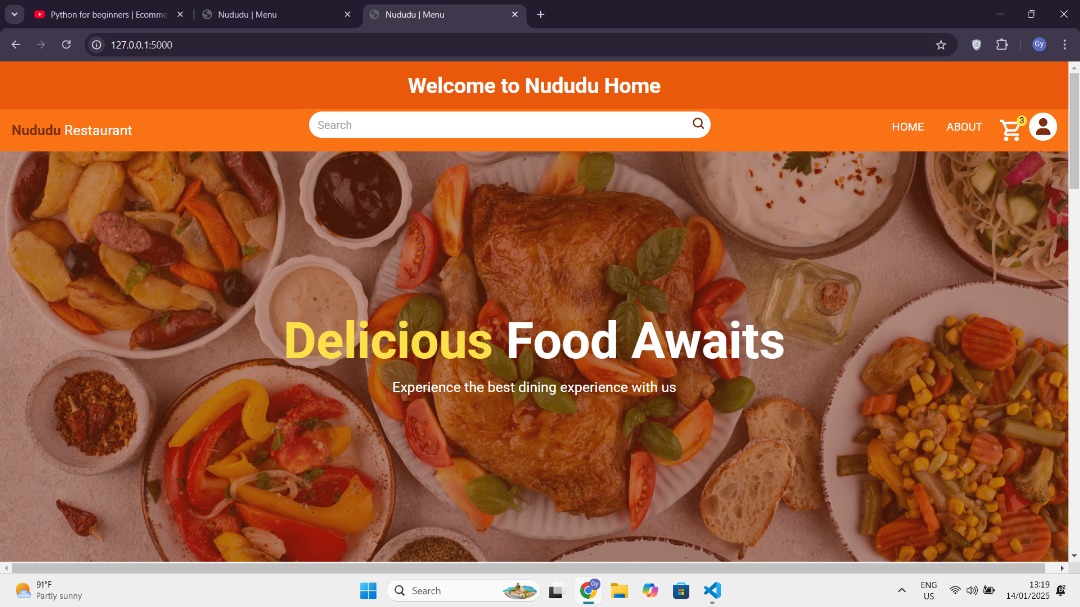
The Home page displays a welcome message to all visited users. It also has a navigation bar that helps users to explore other features of the website, a hero section with cunning words that serves as a hook and advertisement to users. On the navigation bar are the links that leads to the about page, sign In page, cart page, the user profile page, and a search bar. However, if the user is not authenticated, the cart and the user profile link will not be visible. The design of the website is such that, potential customers must create an account/profile in order to make purchases or add food to cart. However, upon accessing the site one can view the HOME Page. The homepage displays warm, simple welcome message. At the top of the page on the nav bar is a search bar which allows users/customers to search for available foods. On the navbar, there are the ‘HOME’, ABOUT’, cart and Profile buttons respectively. 

Figure 2 home page

To place an order for food, the website requests the user to create an account. Clicking on the add to cart button sends the customer to an account creation page where a username, email address and password are requested. After successfully creating an account, the website pops up the message, ‘Account creation successful’.

A screenshot of a computer

Description automatically generated

Figure 3 User profile page

Users can then go ahead to make their selections and place their orders.

***SIGN*** ***IN*** ***PAGE***

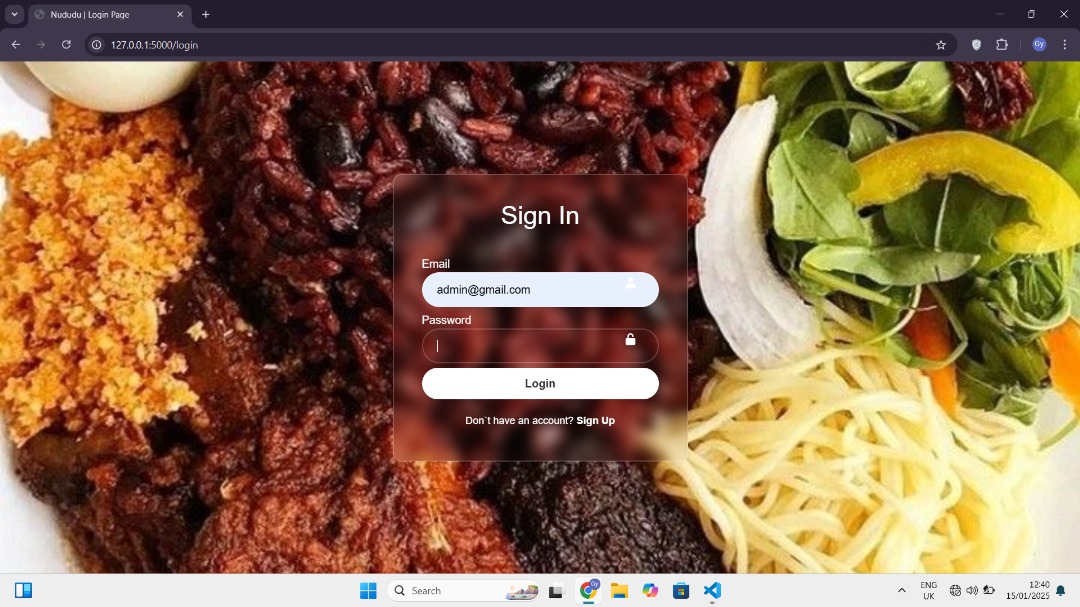


Figure 4 Sign In page

***ABOUT PAGE***

The ABOUT page gives an overview of Nududu Restaurant and its core values and stock-in-trade. This is where information about the restaurant is provided for the pursual of clients and customers. This page also allows for contact with management through social handles and also a WhatsApp contact.

A screenshot of a computer

Description automatically generated

Figure About page

The ‘CART’ button, as would be found on any online sales platform, allows users to view orders in their cart or basket, as well as add or delete orders in the cart. The total cost of orders placed is also calculated automatically and displayed to the user. The user can then proceed to make payment by clicking the ‘check out’ button. A white shopping cart with a yellow circle on an orange background

Description automatically generated

Figure Cart Icon

* Cart page

***MENU PAGE***A screenshot of a menu

Description automatically generated

***UPDATE PRODUCT PAGE***

The update products page allows administrators to add, edit or remove products from the list of available products.

A screenshot of a computer

Description automatically generated

* View customers page

***ADMIN DASHBOARD***

The admin dashboard enables admin to view, add(create), edit and remove employee and customer information.

A screenshot of a computer

Description automatically generated

**HOW TO RUN THE CODE**

Use the link; <https://github.com/GyasiAmosKwadwo/Web-App> to access the source code from Git Hub. In the Web-App repo, open the Full-Ecommerce Website and locate the run.py file to runt the source code. All the dependencies used for the project are preinstalled in the virtual environment. However, if any error occurs whiles running the code, use pip to install the dependencies again. In that case, use the following commands to install the dependencies.

* Pip install flask
* pip install flask-login
* pip install flask-wtf
* pip install werkzeug-security
* pip install intasend-API

The aforementioned commands can be run on the command prompt.

NB: some online resources were used for the frontend design, however, if some functionality are not working, try connecting to the internet.