NIPES

Advances in Engineering Design Technology

Journal homepage: www.nipesjournals.org.ng

Development of Client-Server Architecture for Agro-Information Systems and Services

A. Obayuwana[a], J. U. Okoye[b], S. Akinbohun [c]

augustine.obayuwana@uniben.edu, jokoye360@gmail.com, sylvester.akinbohun@uniben.edu

[a, b,c] Department of Computer Engineering, University of Benin, 1154, P.M.B, Benin City, Nigeria.

ARTICLE INFORMATION

Article history:
Received 11 August 2023
Revised 23 August 2023
Accepted 15 Sept 2023
Available online 5 Oct. 2023

Keywords: server, agro-info, systems.

https://doi.org/10.5281/zenodo.8410159

ABSTRACT

The agricultural industry in emerging nations like Nigeria faces challenges due to limited access to modern technology, infrastructure, and markets. To address these issues, organizations and researchers are using Information and Communication Technologies (ICT) and modern practices in agriculture, such as mobile apps for weather forecasts and market prices, precision agriculture with drones and GPS, blockchain for supply chain transparency, data analytics, online markets, and IoT sensors for environmental monitoring. This research proposes a three-tier architecture for an integrated agro-information system, comprising four software components: Public Front-End, Cloud API Server, Cloud Database Server, and Public Back-End Admin. The system focuses on agro-based information sharing and farm produce prices, and it was rigorously tested for functionality and performance.

1. Introduction

The architecture for most web applications is a Client-Server architecture consisting of a computer network in which the clients communicate with the server [1]. It is a distributed application structure that partitions tasks or workloads between the providers of the resources or services, called servers, and service requesters, called clients Distributed Application [2]. In simple terms, the client, who is typically the requester, submits a request for access to data, the server responds with an acknowledgment, and the client is then given access, and the request is made over the Transfer Control Protocol or Internet Protocol [3].

Client-server is a system that consists of three major components; the client application, the server, and the application and database logic. Depending on the type of client-server model (architecture) the client, server, and its database can reside in one computer system (location), or the client, server and its database can reside in different computer systems [4].

The physical layout of the client/ server network can be made up of three possibilities: the single client, multiple clients, and multiple and multiple servers [5]. The single server is where one server is connected to one client, the multiple clients, the single server is where multiple clients are connected to a single server, and finally, the multiple clients, and multiple servers where multiple clients are connected to multi-servers.

Client-server architecture has found relevance in the development of ICT solutions which includes software applications for various fields of endeavor, these fields include the banking industry [6], for instance, In the banking sector, a customer accesses online banking services using a web browser and

initiates a request to the bank web server, which allow the customer to log in credential which might have been stored in a database and web servers. In communication client-server model is used for sending and receiving electronic mail, users access the server via client portals like Gmail, Microsoft Outlook, yahoo, and a lot more [7]. Another application of client-server application is in the education sector, which is becoming popular for virtual classrooms where students can access their classes, assignments, and course materials [8].

Meanwhile, the application of technology in the agricultural sector is a new field that focuses on improving agricultural productivity [9]. With the application of ICT advancement, farmers and non-farmers can be provided with accurate and timely relevant information and services that facilitate a remunerative agricultural environment [10]. The development of agriculture has enabled the human population to grow many times larger than its original population [11]. Thus, over the years, agriculture has evolved. Modern-day agriculture includes cultivating lands for crop production, animal farming for consumption, raw materials, partial processing of farm products, preservation and storage of farm produce, and marketing of farm produce.

Agriculture has evolved to be the backbone of the economy as it provides raw materials to other sectors of the economy, such as cocoa for tea and coffee to beverage-producing industries, grains to the milling sector, natural latex for tyre production, sugar for sugar production, roots and tobacco for medical use, wood pulp for paper production, fruits, and oil seed for oil production [12]. The design architecture of AgriCom, a distributed multi-agent system for agricultural communication, is presented in [13]. The four agents used are the farmer, the buyer, the seller, and the instructor. The users of AgriCom can communicate via a GUI while the system is running in both manual and automated modes. Thus, twenty agents were used to test the system, and the proposed result was archived.

However, the agricultural sector in developing countries is experiencing some limitations in growth such as Nigeria, due to a lack of availability of information, proper market system, and modernization of its techniques [14]. Therefore, applying proper ICT technology and software applications can improve the reliability of information and the growth of the agricultural sector.

2. Methodology

This section contains all the methods and steps taken to achieve the goals and objectives of the project. Figure 1 is the flow chart of the methodology used in achieving the said objectives of this research work.

2.1 Identification of Major Agro-base Service in Nigeria

The major type of service model can be summarized in Table 1.

Table 1: showing the major agricultural sector

S/N	Type of Service Model	Authors/ Developer
1	Agricultural marketing	Chengwei Zhang and Weiqi Rao
2	Agricultural information	Susan P Naik
3	Agricultural Information	Ganesh S. Wepathak
4	Agricultural Marketing	SantoshG Karkhile
5	Agricultural information	Aniket Bhave
6	Agricultural information	Southern Living
7	Agricultural sales	Farm crowdy
8	Agricultural Information	The Spruce
9	Agricultural sales	Farmforte

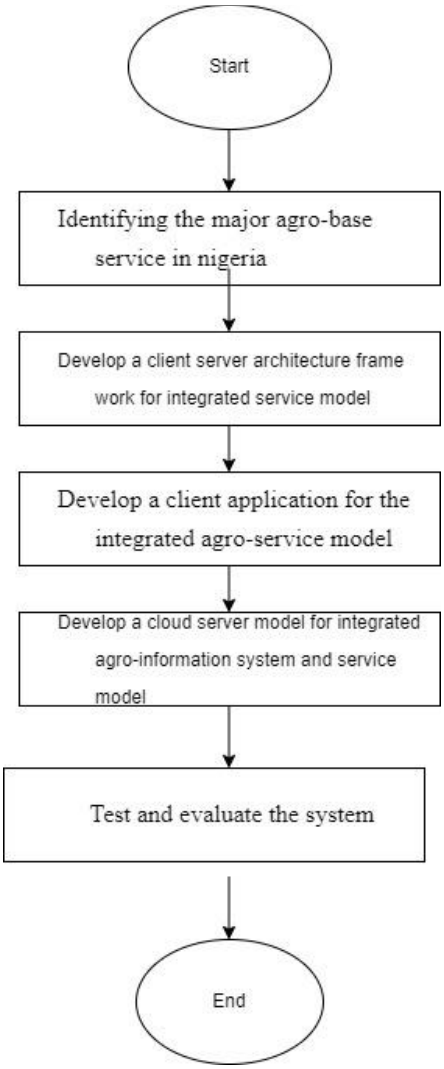

Figure 1: Method flow chat

2.2 Develop a client-server architectural framework for an integrated service model
This model is made up of basically 4 components: Public Front–End, Cloud API Server, Cloud Database Server, and Public Back-End Admin as shown in Figure 2.

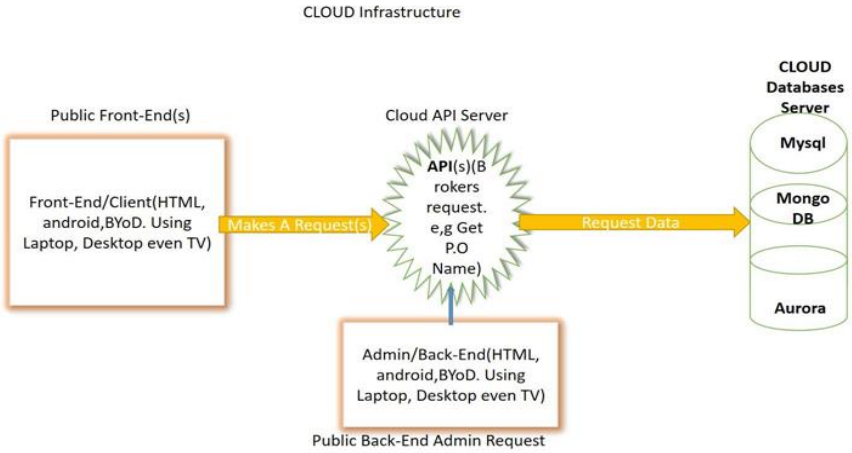

Figure 2: showing the framework for the client-server application

2.3 Development of Client Application for The Integrated Agro-Services Model.

The client application in this work is the user interface, which was programmed using HTML, CSS, and Javascript for their different capability and characteristics. The user interacts directly with this interface, which can be accessed either using a computer system, smartphone, or any other digital device with a browser, Figure 2 shows the workflow of the client-server.

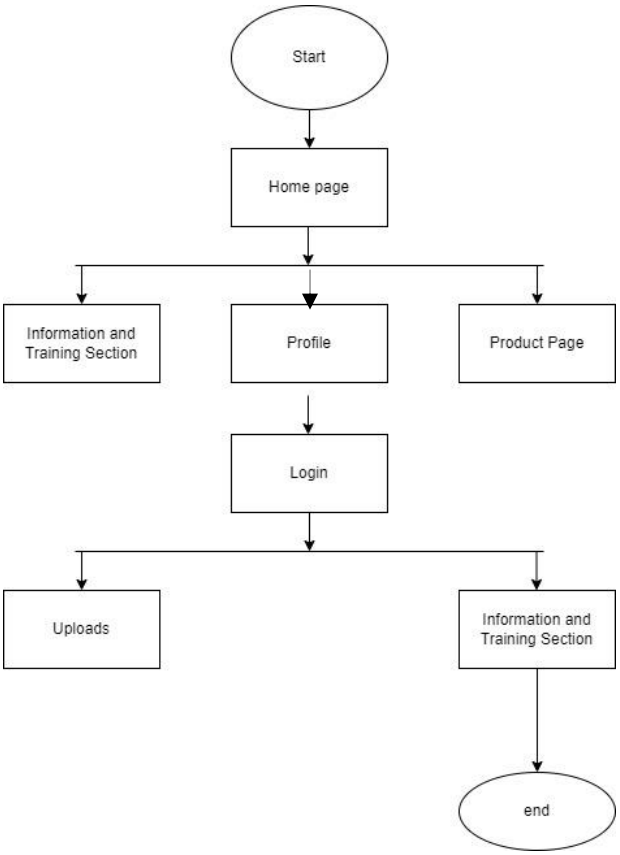

Figure 3: showing the workflow of the client serve

From the result of the comparison of other web applications done in 2.3, there is a need to incorporate the major independent services (sales and information) rendered by this web application into one web application. Hence, the client architecture of this project shall be divided into:

- A home pages
- An information section where information about plants animals and agricultural machines/ tools can be found
- A product page where users can upload products for sales
- A profile page that contains details of users.
- A login and sign-up page

2.4 The Server Application:

The server application is the part of the system that provides functionality to the client application. The server is simply a system that hosts all the software, frameworks, and databases required to run the client-base application. The server application was built on a Firebase using Javascript language to develop the backend of the client-server application. Firebase Authentication was used to enable users to sign into the app using one or more sign-in methods, including email address and password sign-in, and federated identity providers such as Google Sign-in. The server application makes use of Cloud services, as the cloud storage in Firebase allows the user to upload and share user-generated content.

The server application is further broken down into section

a) Database Management System
b) Backend program
c) Cloud Service provider
d) Administrative dashboard

2.4.1 Database Management System

It is a software system that allows users to define, create, manage, and control access to databases. All information inputted by a user is collected and stored in the database. This information includes the login details, the signup details, and their upload of products. The database system used in the project is an SQL database.

2.4.2 Backend program

A backend program that connects the database with the front end. The Program was written using c#

2.4.3 Cloud Service Provider

This is a hosting service, a type of internet hosting service that hosts your website. This project was hosted on AWS.

2.5 Administrative Dashboards

The admin dashboard gives admins direct access to key Encompass tools to make community management faster and more convenient. The dashboard serves as an administrative home page that gives you access to the most important components of your community.

The admin dashboard was created by linking the database to an API, giving the admin access to the web application with added functionality like maintenance.

2.6. Use of case diagram

Figure 4 describes the use cases of the application, and how the user can navigate through the contents of the platform right from the user interface to the application products.

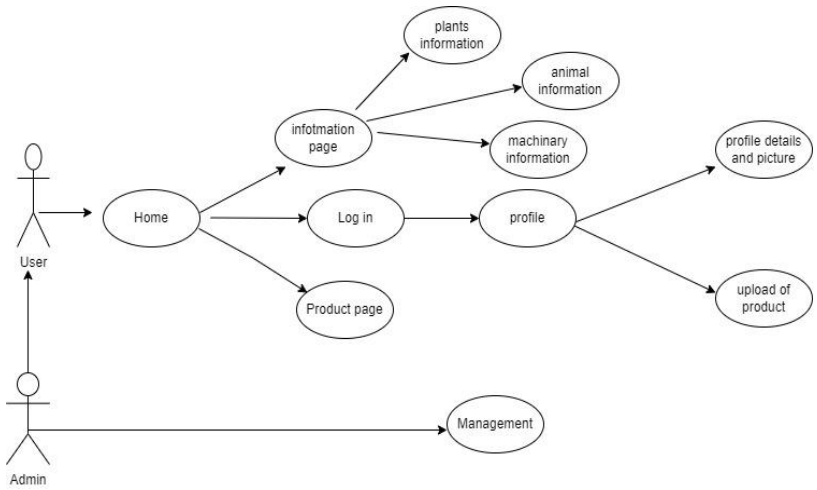

Figure 4: Case Diagram

2.9 Use Case Description

Table 2.: User Login use case description

Title:	**Client-server model (User/public use)**
Description:	The user attempts to use the Web Application
Actor:	All user
Pre-conditions:	The system must be connected to a Network.
Post-conditions:	The web application loads as per specification and user information.
Scenario:	i. The user opens the web application via a web browser ii. User access the information page iii. The user purchases a product iv. The user attempts to upload a product v. Prompt to sign up vi. The user signs up and log-in vii. User uploads product viii. User log-out
Extensions:	i. Invalid username and password show an error message ii. Limited service without sign-up

Table 3.: Admin Login use case description

Title:	Client-server model (Admin use)
Description:	The user attempts to maintain the Web Application
Actor:	Admin
Pre-conditions:	The system must be connected to a Network.
Scenario:	Admin logs into the application Attempt to monitor the application Finds an inappropriate upload Takes the inappropriate upload down

2.8 System Architecture Process

The application would be developed using the waterfall model software process. The waterfall model takes the fundamental process activities of specification, development, validation, and evolution and represents them as separate process phases. There are five stages of the waterfall model which are the requirement definition, system and software design, implementation and testing units, integration, system testing, and finally maintenance.

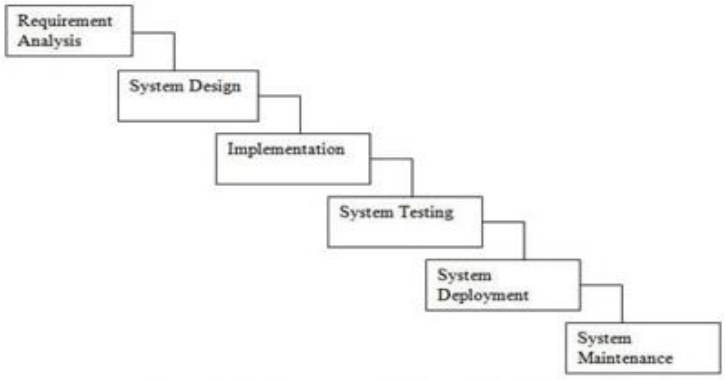

Figure 5: Waterfall Model

2.9 Requirement analysis

Specification and analysis of a system requirement is a key phase in system development, it involves identifying the functionality, constraints, and goals of the system. Requirement specification focuses on what the software should do, rather than how it should be done. Some of the system requirement for this project includes:'

[1] A sign-up page
[2] A pool of agricultural information
[3] A buying and selling section of agricultural produce for traders.
[4] A stable system
[5] The system should be easy to maintain and test.

2.10 System Analysis and Design

This is a way of identifying and describing the fundamental software architecture. It gathers, interprets, and examines a system component in other to achieve its underlying objectives.

2.11 System Implementation

System implementation is a collection of procedures used to achieve the specified system design requirement. System Requirement and Software

- Visual studio code
- A web browser
- 64-bit Windows 10 operating system
- RAM 4GB or higher

2.13 System testing

System testing is the process of ensuring that the functionality of the entire system works following the specifications in the system requirement documents. The projects shall be tested in the following ways:

- Functionality Test: This test is to ensure that the functionality of the program is adhering to all the requirements specification documents.
- Performance Test: This testing would be carried out to ensure the system carries out its functionality under various conditions.

Security Test: This is to confirm that the program is secure.

3. Results and Discussion

In this section, the test result for the development of a client-server application for agro-based service models in Nigeria is presented this includes: the identification of agro-service models in Nigeria, server model of multi-er architecture for agro-based services, client application for integrated agro-information system and services.

3.2 Identification of Agro-service Models in Nigeria

Figure 6 shows the two types of agro-service models in Nigeria, agro-service web applications in Nigeria can be deduced as:

i. few web applications render services specifically to agricultural-produced
ii. These web applications are dedicated to offering one service either rendering information about agricultural produce or sales of the produce
iii. There is a need for an integration of these services.

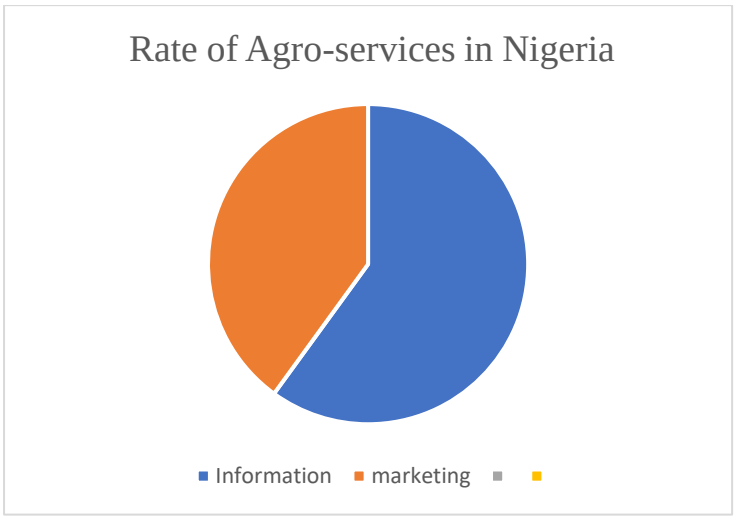

Figure 6: showing the ratio of agricultural services

4.3 Client Application for Integrated Agro-information System and Services.

The client application which the user can interact with composed of the following pages: home page, information page, profile page, product page, log in, and register page

4.3.1 A home page:

Figure 7 shows the home page that loads first when a user runs the application. It contains a navigation bar that links other pages for registered users, a sign-up page for new users, an information page, and a product page.

Figure 7: Home page

i. Information Page

Figure 8 shows an information section where information about plants, animals and agricultural machines/ tools can be found with updated information about them.

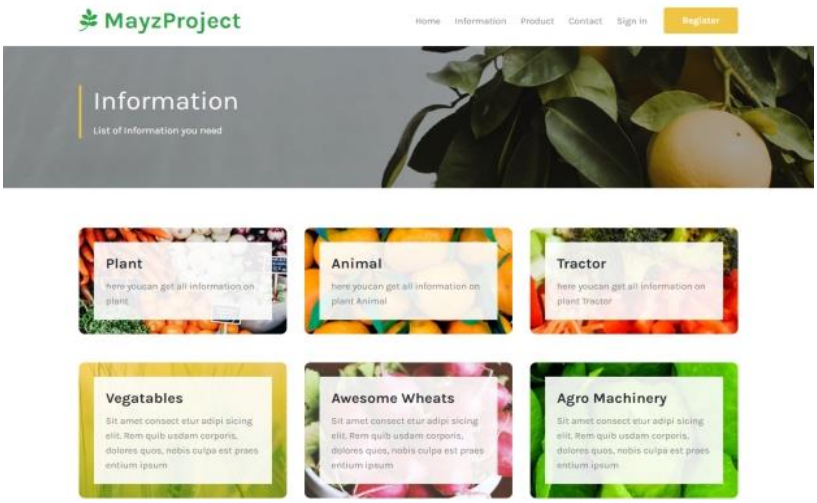

Figure 8: Information page

ii. **Product Page**

The **product page** showcases the product inventory of different farmers who registered with the website for the sales of farm products. It's a page that helps users (consumers) have easy access to farms' products and services. Figure 9 shows the product page developed for the project.

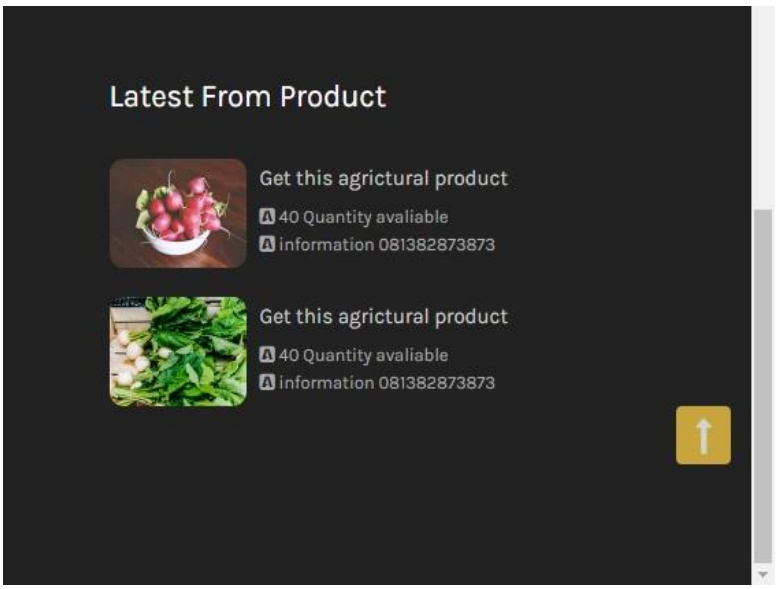

Figure 9: Product page

4.3.2. Profile Page

Figure 10: Profile page

Figure 10 shows the profile pages of different farmers or sellers who register with the site for the purpose sale of farm products and services, each profile is unique to each seller, contains the farmer's information, and enables the farmers or sellers to customize their profile, upload new products as well as managing products already uploaded. It also serves as a means by which users can manage their activities.

i. A login and sign-up page

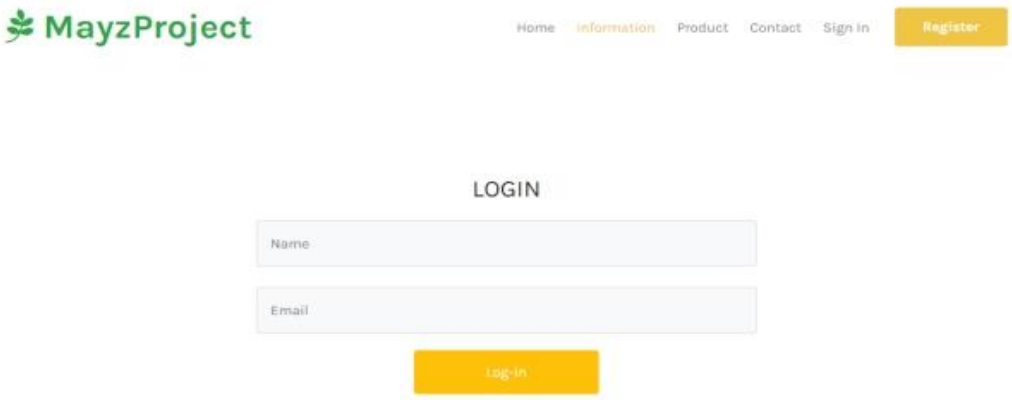

Figure 11: Login page

Figure 11 shows different users can log in to access a resource on the website, it uses decoupled authentication.

4.3.4. Server Model for the Integrated Agro-Information Services

i. Database

Figure 12: showing quarry to login

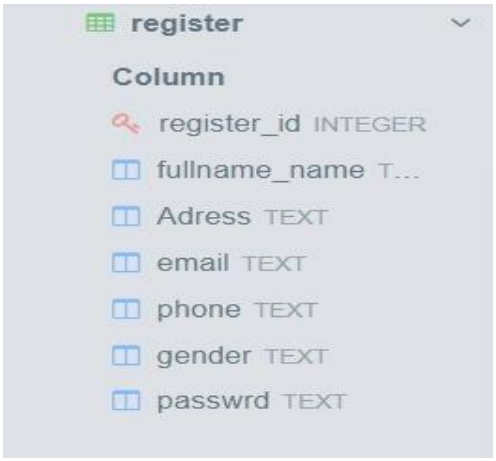

Figure 13: Showing the registration table

Figure 14: showing the registered database

ii. Administrative Dashboard

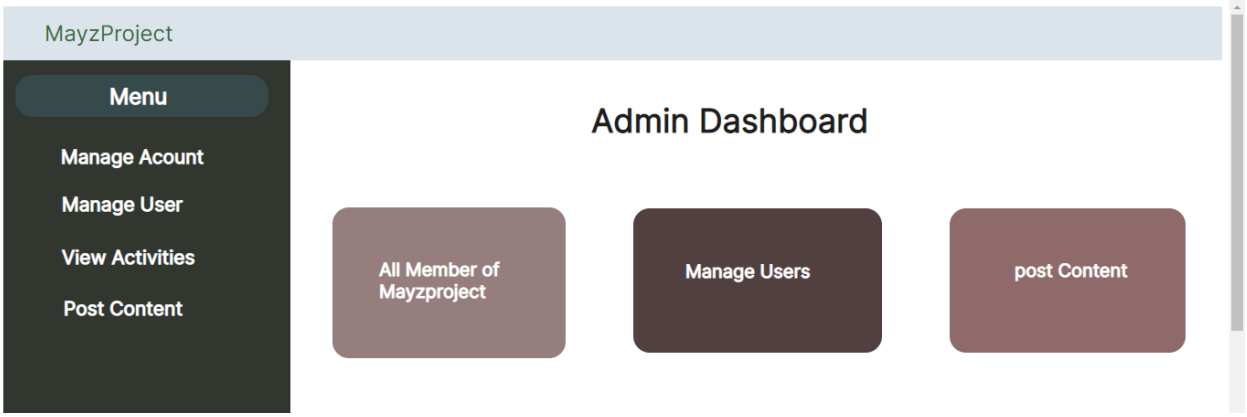

Figure 15: Administrative Dashboard

4.4. Testing and Evaluation of the Developed System

After all the requirements were gathered from the analysis phase, to ensure the functionality of the developed system, we carried out the testing phase on the web application using twelve (12) different agents. Thus, at the end of the project testing, the following was achieved:

Functionality Testing: The output we got when testing the system displayed that each functionality was working as they were specified in the requirements within the system's capabilities.

Performance Testing: This testing was carried out under various conditions, in terms of performance characteristics, which is also called compliance testing concerning performance. The testing ensures that the product meets the specified requirements, by checking the system when multiple users utilize the same application at a time, and then how it responds.

Security Testing: Confirms that the program can access authorized personnel and that authorized personnel can access the functions available to their security level. This testing makes sure that the system does not allow unauthorized access to data and resources. The purpose of security testing is to determine, how well a system protects against unauthorized internal or external access or wilful damage.

Usability Testing: To make sure that the system is easy to use, learn and operate

5.0. Conclusion

Farmers, buyers, sellers, and professional instructors are just a few of the important agricultural business stakeholders whose communication is successfully managed by multi-agent technology. This collaboration and cooperation is made possible via client-server architecture for agro-information systems and services. This work focused on the design and implementation of a client-server model for agricultural information and services. The design involved the detailed analysis and design of a client-server model for agricultural information and services using Object Oriented Analysis and Design using diagrams such as use case diagrams, and system architecture diagram models of various aspects of the system. The projects provide a pool of agricultural information and architecture for managing agricultural information and sales of agricultural products. The platform with improvements and efficiency as opposed to standalone web-based or mobile-based applications as proposed in [13] with a unified code-based, fast development, and rollout rate is the future of software development. Thus, agriculture and technology are the future of the economy; the application of modern farming techniques is paramount in the agricultural sector and this application would play a long role in achieving that goal.

References

[1] Crosbie, E., Erinoso, O., Perez, S., & Sebrié, E. M. (2023). Moving in the right direction: tobacco packaging and labeling in the Americas. Revista Panamericana de Salud Pública, 46, e196.

[2] Subhasha chandar, & Sanjay Kumar. (2009). An introduction to client-server computing. New Age International (p)limited publisher.

[3] Köksal Ö., & B. Tekinerdogan. (2018). The architecture design approach for IoT-based farm management information system.

[4] Obidike, N. A. (2011). Rural Farmers' Problems Accessing Agricultural Information A Case Study of Nsukka Local Government Area of Enugu State, Nigeria. Library Philosophy and Practice (e-journal). Retrieved from

[5] Mali, G. S. (2015). An approach of software engineering through middleware. The international journey of Computer Engineering & Technology.

[6] Mell, P., & Grance, T. (2011). The NIST Definition of Cloud Computing. US Department of Commerce.

[7] Song, L., & García-Valls, M. (2022). Improving the security of web servers in critical IoT systems through self-monitoring of vulnerabilities. Sensors, 22(13), 5004.

[8] Qodirov, F., Sirojev, N., & Negmatova, S. (2023). FEATURES OF THE ANDROID STUDIO SOFTWARE PACKAGE. Академические исследования в современной науке, 2(17), 130-146.

[9] Okediran O, Ganiyu R. A., & Badmus T. A. (2018). NmAgricultural framework for the agricultural services delivery. LAUTECH Journal of Engineering and Technology, 72-79.

[10] Jill S. Auburn, & Jayne T. MacLean. (2008). Development of an Information Network for sustainable agriculture. Journal of Agricultural & Food Information.

[11] Pawlak, K., & Kołodziejczak, M. (2020). The Role of Agriculture in Ensuring Food Security in Developing Countries: Considerations in the Context of the Problem of Sustainable Food Production. Sustainability, 12(13), 5488. MDPI AG. Retrieved from http://dx.doi.org/10.3390/su12135488

[12] Praburaj, L., Design, F., & Nadu, T. (2018). Role of agriculture in the economic development of a country. Shanlax International Journal of Commerce, 6(3), 1-5.

[13]Jayarathna, H. M. H. R., & Hettige, B. (2013, December). AgriCom: A communication platform for the agriculture sector. In 2013 IEEE 8th International Conference on Industrial and Information Systems (pp. 439-444). IEEE.