



INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS

ISSN 2320-7345

DEVELOPMENT OF A PHARMACEUTICAL MANAGEMENT SOFTWARE: (FAMACXIA)

Tingir, T. James¹, Akuhwa T. David², Yilwatda M. Morkat³, Tsenongo N. Stephen⁴

¹²³Department of Electrical and Electronics Engineering, University of Agriculture, Makurdi,

⁴Bringfireh Technologies, Gboko, Benue State

¹Email: jaytinter@gmail.com. ²Email: akuhwagong@gmail.com, ¹Email: manmorkat@gmail.com. ²Email: ngutortsenongo@gmail.com.

Abstract: - The research involved in this work is the development of a pharmaceutical management application (famacxia). Famacxia is a software developed to cater for all the requirements of an organisation that deals in Pharmaceutical products. It has the Administrator and Staff Interfaces which are logged in. The Administrator Interface creates accounts for staff, accounts for stores, record of financial details and registers companies dealing with the Organisation while the Staff Interface is for handling accounts related to Customers, Products, Sales, Pricing, Purchasing among others. The application is run online and is used by Javascript based browsers. The Application was written in ASP.NET and the GUI application was written using C-sharp programming language and developed using the Visual Studio 2015 Environment.

Keywords: Pharmaceutical, ASP.NET, C-Sharp, Visual Studio. GUI

1. Introduction

The world of business, especially large organizations with many activities, keeping track of every activity usually poses a huge challenge. Inventories, sales, accounts, products, prices for products etc. become very difficult to track. In the Pharmaceutical Industry, there are several categories of products, several brands and different customers for a different product. The pricing for each of these products vary and becomes very difficult to keep track. The Software Famacxia is aimed at keeping track of all the products, inventories, accounts, sales, purchases and pricing for products in the Pharmaceutical Industry. Famacxia development structure has two interfaces: Administrator and User. The administrator interface handles registration processes; that is, staff registration, company registration, registration of stores, and financials. The User interface handles customer, products, sales, pricing and purchasing among others.

Many systems have been developed for managing pharmaceutical industries. Onuiri et al (2016) designed an Online Pharmaceutical Management system whose main aim was to provide a platform to help with drug regulation, as well as providing ease to all parties involved. The methodology used in the implementation of the software was the Incremental Model of System Development Life Cycle, which allowed room for scalability as time went on. The System used the JavaScript, for the webpage, PHP as a server-side scripting language to compute records and information, MySQL for database management, and AJAX to bring web application features to the system.

Samaan (2017) designed and implemented a pharmaceutical inventory database management system. The system was implemented by creating a database containing information about the stored medicines in the

inventory, customers making transactions with the pharmaceutical trading company (which owns the inventory), medical suppliers, employees, payments, etc. The database was connected to the main application using C sharp. The system was useful in managing inventory operations which include adding/Updating employees' information, preparing sale and purchase invoices, generating reports, adding/Updating customers and suppliers, tracking customer payments and checking expired medicines in order to be disposed.

Sylvester et al (2016) designed and modelled an Integrative computerized pharmacy inventory system that would had the capability to improve patient care by promoting medication dispensing accuracy and also allow pharmacists to spend more time on patient care activities. The structured system analysis and design methodology (SSADM) and UML (Unified Modelling Language) used for the model design. The interface between the front-end and the back-end is through SQL (in WAMPSEVER) and through an application program interface (API) defined in PHP. PHP was used as back-end for managing access structures, query evaluation and optimization, concurrency control and recovery. NetBeans IDE was used as front-end to develop forms, report-trail, and graphical user interface.

Liu (2015) designed an enterprise management system based on JavaEE work platform. The system was to aid the enterprise management personnel in a timely manner to provide pharmaceutical companies operating in the process of all kinds of information, improve managers in the management of user efficiency and level of employment, with strong adaptability and practicability. The research highlights the principle and structure of the system, proving it to be an optimal management system with flexible mechanism, high safety, and extendibility.

2. System Analysis and Methodology

The Methodology employed in the design of this system was the Incremental Model system where the software is designed and tested step by step

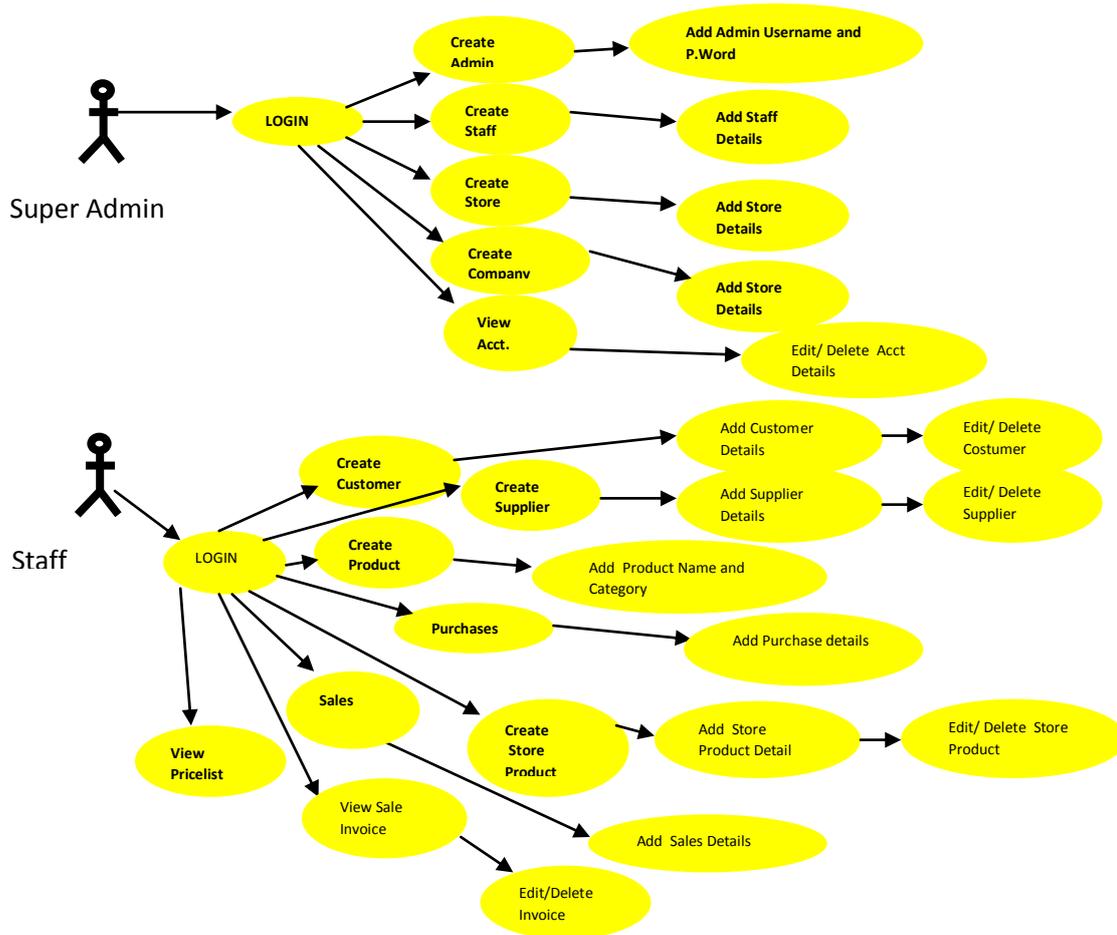


Figure 1: Use-Case Diagram

2.1 Software requirements.

1. The Software is compatible with Windows (2000, XP, Vista, 7, 8 and 10), Mac OS and Linux Operating Systems.
2. Java Script enabled Browsers.

2.2 Hardware requirements

1. Volatile Memory (Random Access Memory, RAM) of at least 512 MB.
2. Processor Speed of at least 500 MHz.
3. Storage Memory of at least 5 GB

2.3 System implementation and structure

In software development, implementation is refers to realization of the objective or algorithm as a program, through coding and deployment of software.

The system runs on a number of languages and tools which make it fully functional and dynamic. These include:

1. ASP.NET is an open-source server-side web application framework designed for web development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, web applications and web services (Wikipedia, 2018a).
2. JavaScript is a lightweight, interpreted, object-oriented language with first-class functions, and is best known as the scripting language for Web pages, but it's used in many non-browser environments as well . It is a prototype-based, multi-paradigm scripting language that is dynamic, and supports object-oriented, imperative, and functional programming styles (MDN web docs, 2016). JavaScript is usually runs on the client side of the web, and was can be used to design how the program pages behave in the occurrence of an event.
3. MySQL is an open-source database management system owned by Swedish Company, Oracle Corporation. The system gets its name from the combination of the name of co-founder Michael Widenius's daughter, "My", and "SQL", the abbreviation for Structured Query Language. (Wikipedia, 2018b). MySQL is written in C and C++ and used on Most platforms including Windows, Mac, Linux amongst others.

2.4 Mode of operation.

The system has ASP.NET as its base which works with the other tools to bring to life Pharmaceutical Management System (Famacxia). The system is used online; hence JavaScript is necessary to improve interactions with the users of the system. The MySQL tool is useful in the system in creating the database which the system uses to store information for different functions and purposes. The Graphical User Interface (GUI) is the link between the user and the software system. The GUI of the system works with all the different phases in the system coming together to form a cohesive unit. The Use Case Diagram of the System is shown in Figure 1. The system has two phases: the Super Administrator and the Staff modules. The Login Page (Screenshot shown in Figure 3) gives access to the Administrator or Staff based on their login details. The Administrator has access to create admin, create staff, create company, create store and access to view the accounts of all transactions. The User has access to Create Products and Product Categories, Add Suppliers Details, Add Sales Details, Print Invoices, Change Prices, Check inventory on available Products, Add Customers Details and Purchasing Details. The structure of the Pharmaceutical Management System is shown in Figure 2.

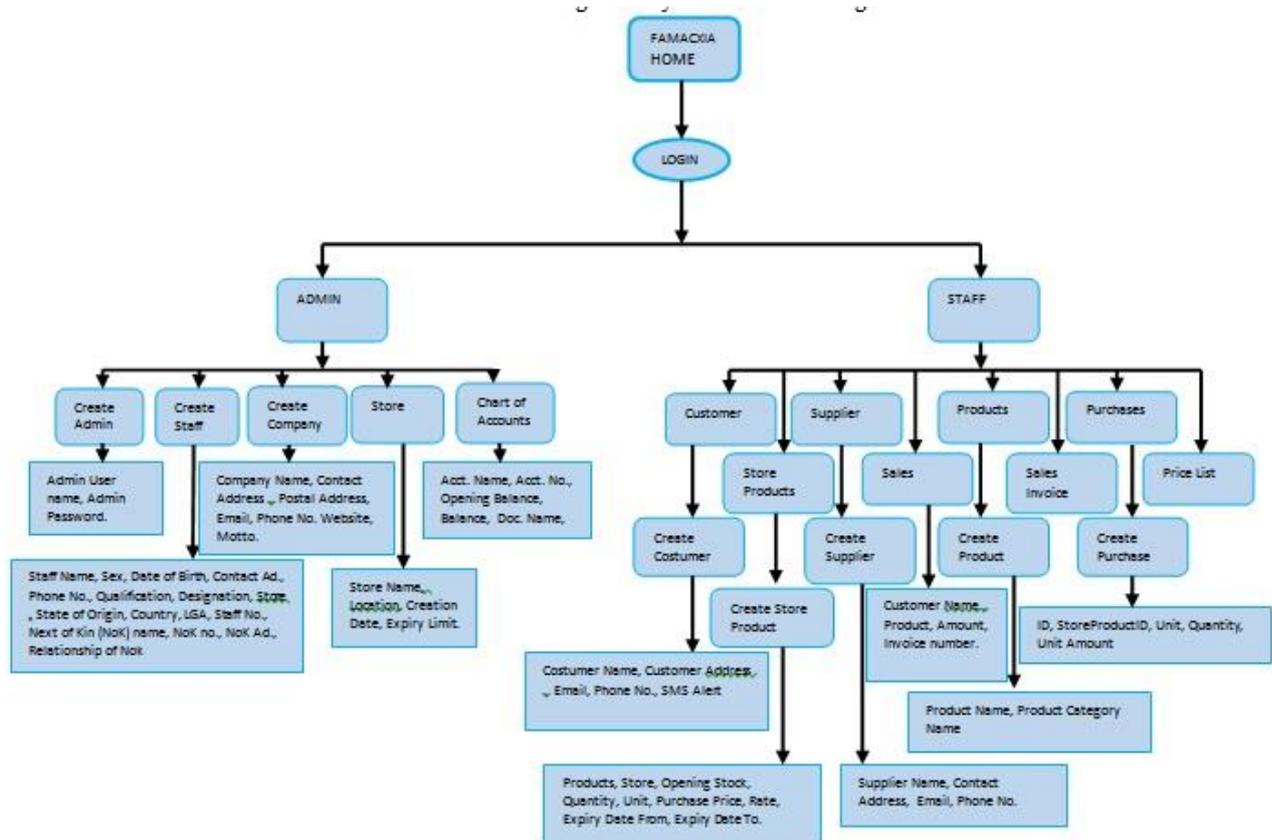


Figure 2 : Structure of Famacxia

2.4.1 Administrator module.

The Administrator Module has five sections; Create Admin, Create Staff, Create Company, Store and Chart of Accounts.

- Create Admin: The system requests for Admin details which include Admin username and password
- Create Staff: The staff details are required to be inputted to complete the creation. The detail required include Staff Name, Sex, Date of Birth, Contact Address, Phone Number, Qualification, Designation, Store , State of Origin, Country, Local Government Area, Staff Number, Next of Kin name, Next of Kin Phone Number, Next of Kin Address and Relationship of Next of Kin.
- Create Company: Details of the company to be registered are inputted. The details are Company Name, Contact Address , Postal Address, Email, Phone Number. Website and Motto.
- Store : To add a store to the system, details like Store Name, Location, Creation Date, Expiry Limit are required.
- Chart of Accounts: the Administrator has access to Input, Edit and Delete entries in the company account.

A screenshot of the Administrator interface is shown in Figure 4.

2.4.2 Staff module.

The Staff Module (screenshot shown in Figure 5) has eight sections: Customer, Supplier, Products, Purchases, Store Products, Sales, Sale Invoices and Price list.

- Customer: The staff inputs the required customer details in the required sections. The details required for the registration include Customer Name, Customer Address , Email, Phone Number, and a check box for SMS Alert. The screenshot for the Create Customer Window is shown in Figure 6.
- Supplier: The supplier details required are Supplier Name, Contact Address, Email, Phone Number.
- Products: The products window requires the following details: Product Name, Product Category Name
- Purchase: the Staff is required to input Purchase details like ID, StoreProductID, Unit, Quantity, Unit Amount

- Store Product: This takes into consideration the inventories in the store. The detail involved includes Products, Store, Opening Stock, Quantity, Unit, Purchase Price, Rate, and Expiry Date From, Expiry Date To.
- Sales: the Sales Window has details such as Customer Name, Product, Amount, Invoice number.
- Sales Invoices: the staff using the application can view sales invoice generated from the details inputted in the sales window. A screenshot of the sales invoice is shown in Figure 7
- Price List: This windows gives the prices of all the products handled by the organization. The Screenshot of the price list is shown in Figure 8.

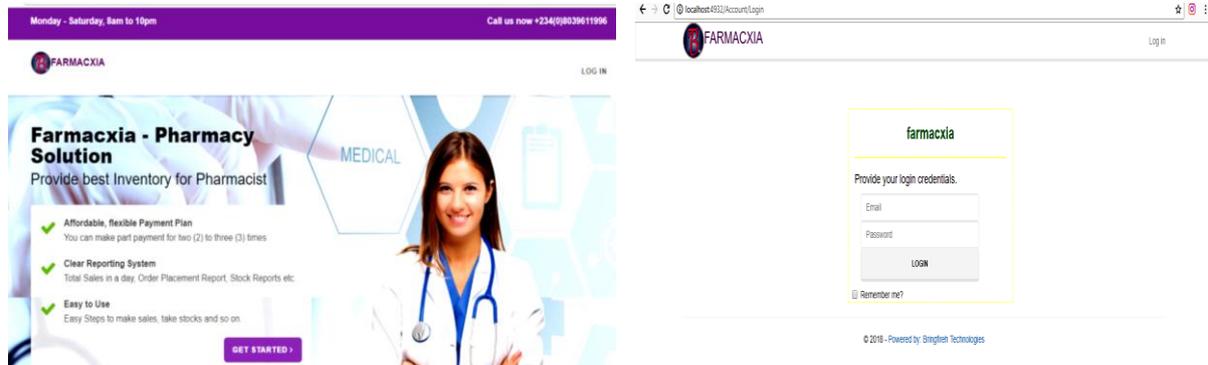


Figure 3: Famacxia Home Page and Login Page.

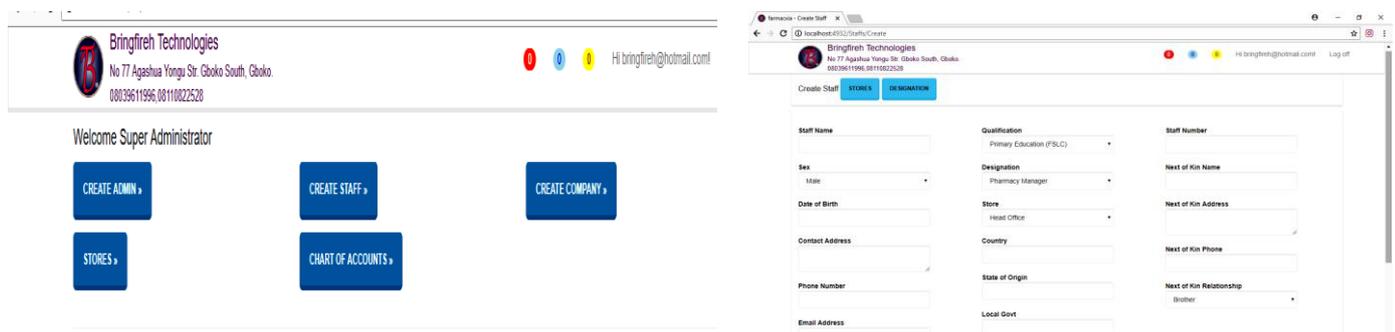


Figure 4: Administrator module interface with the "Create Staff" window

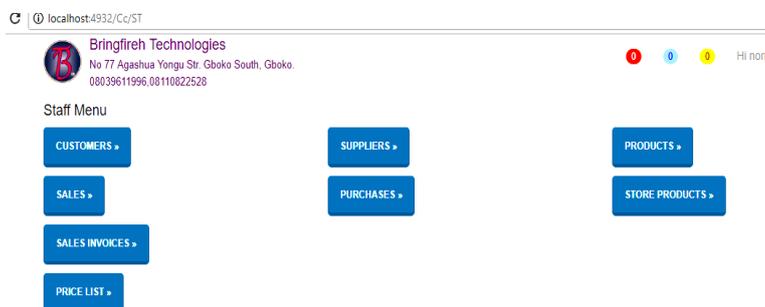


Figure 5: Staff module interface.

Figure 6: "Create Costumer" window.

Invoice Number	Invoice Type	Amount	Invoice Date
0e5f973e-7968-45	Credit Sales	0.00	08-May-2018

Figure 7: Sales invoice.

Product Name	Unit	Rate (NGN)
Tabs - Ciprofloxacin 500mg	Card	250.00
IV - Ciprofloxacin 500mg	bottle	250.00
Cream - Furbact A	Tube	350.00
INJ - Gentamicin	bottle	250.00

Figure 8: Screenshot of pricelist of products.

3. Software Testing and Results

Testing is done at two stages namely:-

1. Testing in Units, where the modules are tested one by one before integrating them This is helps in straightening out faults before complexities arise when integrating.
2. System Testing, where the integrated system is tested to see if it conforms with requirements and to check if the objectives in mind when designing the system have been met.

3.1 Graphical user interface testing

The Graphical User Interface (GUI) was tested to ensure that the graphic component of the system encompasses all functions in the system contained by the different modules. In this system, the administrator and staff interfaces are tested to ensure that all links in their respective modules work according to their functions.

4. Conclusion

In the development of this Pharmaceutical Management System, the design was set up such that only authorized personnel can have access to the individual interfaces hence the inclusion of a login. The administrator's interface allows only the administrator to add Staff, Companies, Stores and other administrators to the system, It also gives the administrator access to view the detail of accounts from all transactions. The Staff can add Customers and other necessary details that involve day to day running of the organization. The system was tested and implemented successfully. All the objectives aimed at in the development of this software were achieved.

4.1 Recommendation

- More login options could be added so that only authorized personal could have access to specific windows.
- The Administrator interface should have a link to access and perform functions in the Staff interface without logging in as a staff.

REFERENCES

- [1] Liu W., 2015, Design and implementation of pharmaceutical enterprise management system
- [2] based on JavaEE, *Journal of Chemical and Pharmaceutical Research*, Vol. 7, Issue 3, pp. 417-421
- [3] MDN Web Docs (2016). What is JavaScript?. (Accessed May, 2018) [Retrieved from https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript]
- [4] Microsoft Visual Studio (c) Microsoft Corporation . Version 2015
- [5] Onuiri E. E., Oyebanji I. G., Fayehun S. A., Chukwujioko S., 2016, Online Pharmaceutical Management System, *European Scientific Journal* , Vol. 12, Issue 12, pp. 139-156
- [6] Samaan, S. S. (2017) Design and Implementation of a Pharmaceutical Inventory Database Management System, *Al-Kwharizmi Engineering Journal* Vol. 13 Issue 1. pp 118-128
- [7] Sylvester I. E., Adesola, W. A., Ofem A. O., Ackley P. A., 2016, Design of an Integrated Computerized Pharmacy Inventory Monitoring System (ICPIMS): A Case of University of Calabar Teaching Hospital (UCTH) *British Journal of Mathematics & Computer Science*, Vol. 15, Issue 5, pp. 1-13.
- [8] Wikipedia: The Free Encyclopedia, 2018a, ASP.NET. (accessed February, 2018) [retrieved from <https://en.wikipedia.org/wiki/ASP.NET>]
- [9] Wikipedia: The Free Encyclopedia , 2018b, MySQL. (accessed March, 2018) [retrieved from <https://en.wikipedia.org/wiki/MySQL#Deployment>]

About Authors

Tingir T. James - holds a B.Eng and M.Eng. in Electrical and Electronics Engineering (Specialized in Digital Electronics and Computers) from the University of Agriculture, Makurdi, Nigeria. He is currently a staff at the Department of Electrical & Electronics Engineering, University of Agriculture, Makurdi. His areas of interests include Embedded Systems, Opto-electronics and Software Development.

Akuhwa T. David - holds a B.Eng. in Electrical and Electronics Engineering from University of Maiduguri, Nigeria. He is currently undergoing an M.Eng (Specialization in Digital Electronics and Computers) at the University of Agriculture, Makurdi. . He is also a staff at the Department of Electrical & Electronics Engineering, University of Agriculture, Makurdi His areas of interests include Power Electronics, Embedded Systems and Software Development.

Yilwatda M. Morkat - holds a B.Eng and M.Eng. in Electrical and Electronics Engineering (Specialized in Digital Electronics and Computers) from the University of Agriculture, Makurdi, Nigeria. He is currently a staff at the Department of Electrical & Electronics Engineering, University of Agriculture, Makurdi. His areas of interests include Embedded Systems, Intelligent Systems and Computers.

Tsenongo N. Stephen - holds a B.Sc. in Chemistry from University of Agriculture, Makurdi, Nigeria. He is the founder and CEO of Bringfireh Technologies, an organization that is specialized in software development and has certifications in several programming languages such as MySQL, Java, C++. His interests include Database Applications and Software Development.