A Real-Time USSD Food Ordering System for Restaurants

Odukoya Oluwatoyin H.
Obafemi Awolowo University Ile-Ife, Osun State
Nigeria

Aina Segun
Obafemi Awolowo University Ile-Ife, Osun State
Nigeria

Makinde Abass Adekunle
Obafemi Awolowo UniversityIle-Ife, Osun State
Nigeria

ABSTRACT
A majority of the students find it difficult to order food from restaurants. Many food ordering systems are either website-based or mobile application-based, which require a good smartphone, good internet access, and a sufficient storage device. However, the aforementioned phone requirements have made the system unusable because of the cost and the difficulty of getting good internet access. It was developed in such a way that, for the first time, users will register on the platform. For subsequent times, users will just go straight to placing their orders and the system will automatically detect the user's phone number. An admin dashboard system was developed in which the restaurant's staff can see the orders being made and can manage them using the platform. The objectives of this work have been achieved and, based on the results of this work, the eating of food on campus can be enhanced by using USSD applications to connect and facilitate communication between the restaurants and customers.

General Terms
Food ordering, USSD

Keywords
Smartphone, real-time food ordering, USSD

1. INTRODUCTION
A food ordering system is a method of placing an order for food using a restaurant's own website or mobile app, or through a website or app that serves multiple restaurants. Customers have the option of either having the food delivered or picked up (Wikipedia). Web-based solutions with interactive menus that allow clients to place food orders [2] are known as online meal ordering systems. All over the world, food delivery already accounts for £83 million, or one percent comes from the total food market, and including the 4 percent from restaurants and fast food chains. This figure is expected to rise to 3.5 percent in many mature countries over the next five years. By far the most common category is waiting for the restaurant to bring the food to the customer, but nearly three-quarters still use phones [6]. As a result of this astonishing rate of online ordering, there may still be some disadvantages or problems that need to be handled, such as website costs, infrastructure costs, security and fraud, privacy laws, computer ethics, advertising costs, and customer costs [7]. A food ordering system is an application that will help restaurants optimize and have control of their restaurants. For the waiters, it is making life easier because they don't have to go to the kitchen and give the orders to the chef. From the management point of view, the manager will be able to control the restaurant by having all the reports and keep record of all their customers. Some disadvantages or concerns, such as website expenses, infrastructure costs, security and fraud, privacy laws, computer ethics, advertising costs, and customer costs, may still need to be addressed at this astounding rate of online ordering[7].

1.1 A USSD Food Ordering System
USSD (Unstructured Supplementary Service Data) is a GSM communication technique that allows a mobile phone to transmit messages to a network-based application server [5]. It is a device and SIM-independent messaging service that is seven times faster than SMS and is extremely cost-effective. USSD can support interactive menu-based applications that allow both speech and data transmission at the same time [10]. Mobile chatting, m-commerce, pre-paid balance enquiries, callback services, software upgrades, and mobile banking services all use USSD [10]. News, weather, movie information, sports updates, currency updates, stock market reports, telephone directory and yellow pages are examples of 'pull' based services, while voting/polling and emergency information services are examples of 'push' based services. It is also utilized for marketing purposes and (ii) 'push' based services that include voting/polling and emergency information services. It is also used for advertising, where businesses get listed on menu based USSD systems in order to promote their services.

USSD has several advantages: it is faster, with average response times of 2 seconds; it is supported by all GSM phones; it is phone and SIM card independent; users do not need to type messages or remember short codes to access services; its menu-based interaction allows network operators to offer self-care applications for Value-Added Service (VAS);which helps network operators improve Average Revenue Per User (ARPU) and operates even when consumers are traveling, helps network operators increase ARPU. USSD, on the other hand, preserves resources allotted for the duration of transactions, resulting in higher traffic on the MSC's and Home Location Register's communication channels (HLR). Furthermore, sent messages are not saved on the device for future use [10].

2. LITERATURE REVIEW
The authors in [4] developed a system that provides a view of current food information (category, name, image, price, description, etc.) on the website and in the Android app. These two systems allow customers to place meal orders. This application provides a set of actions for restaurant administrators to add, update, delete, and query information about food, meal orders, and personnel. The system is not real-time, which means the restaurant admin has to refresh the website every time to see the new order. This can be improved by using Firebase real-time database, which allows the restaurant admin to be notified in real-time as new orders are made. [8] Carried out market analysis in South Ostrobothnia to identify whether a home delivery system is required or not.

Another goal of this thesis is to determine whether or not it will be well received. If not, why not? If so, how and when will you do it? Qualitative and quantitative research approaches were used in the study. Quantitative research yielded numerical results, whereas qualitative research yielded in-depth interviews and thoughts. The questionnaires were kept simple to make them easy to understand and reply to. Short questionnaires were utilized in quantitative research. Interviews were held with those who were interested and willing to participate. They had already been given the questionnaires prior to the interview dates. The scope of the analysis was restricted to South Ostrobothnia.

In universities, the most common phenomenon is that, owing to variable timetables and study pressure during the examination process, it could be more difficult for students to decide when and where to have their lunch. With the increasing popularity of food delivery in colleges and universities, the traditional telephone order of food has inconvenienced both the customers and the food delivery store. [11] Explores and develops new takeaway apps that are easier and more object-oriented than existing apps. The system can be improved by making it real time. The system does not include in app purchase, which is very important in the food delivery system. It rather depends on other payment platforms like Apple Pay. In [1], an online food ordering system that uses Internet of Things technology is proposed. It uses wireless technology and Android devices to order food easily from restaurants. Food is ordered, transactions are kept track of, and there is a way for users to give feedback and rate the food they have ordered. The authors in [11] proposed an android based order placement system for restaurants. This was an improvement over the paper based ordering system and point of sale system. In this system, customers can place orders from home, and it also has extra features like the ability to report and analyze data. In [9], the authors proposed an online food ordering application that has a Global Positioning System (GPS) option to give the customer the ability to see restaurants near their area. The user sign up for the app, chooses food from a menu card, and places an order through an Android app.

3. PROPOSED SYSTEM

The system architecture of The USSD food ordering system adopts a client server side architecture where the mobile device allows users to dial a service code, which serves as the client side of the system, and send requests to the server API where the business logic is defined and written in Node.js, a web framework of JavaScript (programming language). Figure 1 below shows a picture of the system architecture, which is called a system architecture diagram.

The system architecture contains the following components listed below.

1) The mobile phone which is on the system client side, in this project, is a third party sandbox simulator, which is used to make the USSD request. This is the main interaction interface between the users and the system; it takes input from the user and sends it to the backend server side to process.

2) The API Service for USSD Is a form of micro-service, a RESTful API that handles the core logic of the system, connects the database and all other micro services to the client side.

3) Mailing micro-service this is another micro-service on the server side. There are a variety of mailing services to accomplish this task. In the course of this project, a third party library written in Node.js called Nodemailer aids the USSD system by sending a mail receipt to the customer after a successful order.

4) Real-time Helper (Pusher) This is a way to fix the problem of data transmission from the client side of the customer to the admin dashboard, which is also a client side, taking too long or failing.

5) Cloud Database A cloud version is adopted for customer data persistence. The data consists of the basic information of the customer alongside with the order details. A cloud database is a database that runs on a cloud computing platform and provides as-a-service access to the data. The database's scalability and availability are ensured by database services. The underlying software stack is hidden from the user thanks to database services (Wikipedia).

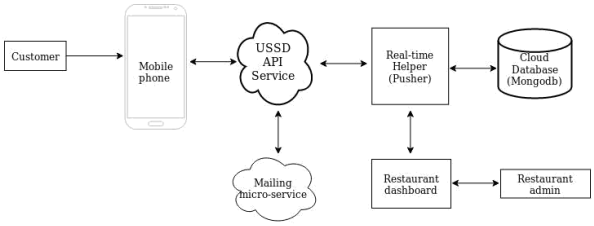

Figure 1 System Architecture

3.1 Use Case Diagram

The use case diagram shown in Figure 2 shows various activities that can be performed by the two categories of users of the system.

The use cases are described as follows.

1. **Dial USSD number:** This is the first interaction point between the user and the system. This usecase allows a user to enter the USSD code, which will then initiate the system, and this use case will include another functionality of the other use case called "Register customer."
2. **Register Customer:** This use case allows users to provide registration information to the **USSD API.** The informationcollected at this stage includes name, email, and phone number, which the USSD API will automatically detect the user's phone number from the device, and all this information taken from the customer will be used to register the user at the back end server side and persist them to the database.
3. **View the menu list:** This use case allows customers to view the available food items.
4. **Enter your order:** This use case allows the customer to enter the desired order into the system.
5. **Enter the delivery address**: This use case allows the customer to enter the delivery address,that is, where they want the food to be delivered.
6. **Place your order here**: This use case will now take the user's entries and process the orderwith them, which consists of sub processes such as validating the details, persisting them to the database, and finally sending them to the admin dashboard with the aid of the third party module pusher.
7. **Send mail:** This use case performs the mail messaging action, in which the system will send the receipt message to the customer after a successful transaction.

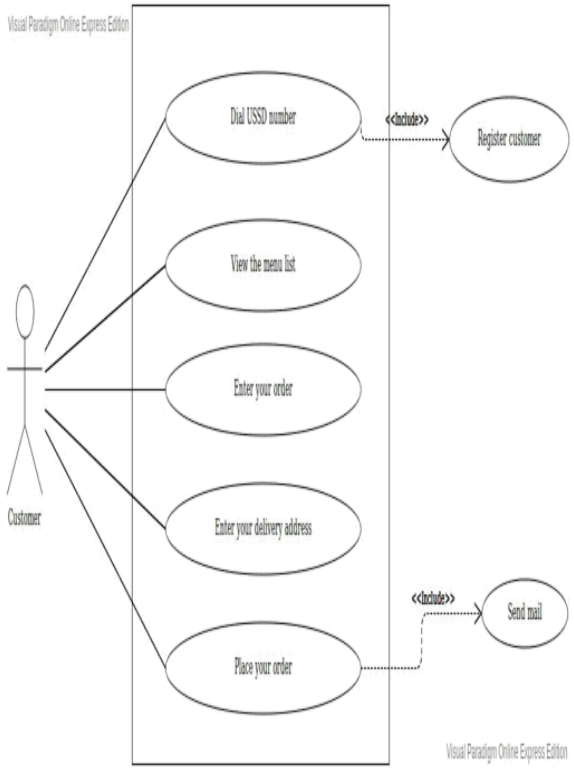

Figure 2 Customer use case diagram for food Ordering System

3.1 Activity Diagram

Another essential diagram in UML for describing the dynamic characteristics of the system is the Activity diagram. An activity diagram is essentially a flowchart that depicts the movement of information from one action to the next. The action can be described as a system operation. From one action to the next, the control flow is depicted. This flow can be sequential, branching, or running at the same time. Different elements such as fork, join, and so on are used in activity diagrams to cope with various sorts of flow control.

The activity diagram for this system is shown below in Figure 3: It shows the Flow of Operations For making orders by customer, processing orders by the system, and viewing the orders as well as closing the orders by the admin.

3.2 Class Diagram

Diagram of a Class in the Unified Modeling Language, a class diagram depicts the relationships and source code dependencies between classes (UML). A class defines the methods and variables in an object, which is a specific entity in a program or the unit of code that represents that entity in this context. In all types of object-oriented programming, class diagrams are useful (OOP). The concept has been developed as OOP modeling paradigms have progressed over the years. The classes are organized into groups that share common properties in a class diagram. A class diagram is similar to a flowchart in which classes are represented as boxes with three rectangles inside each box. The top rectangle holds the class's name; the middle rectangle contains the class's properties; and the bottom rectangle contains the class's methods, commonly known as operations. The boxes are connected by lines with arrows at one or both ends. Figure 4 illustrates this. These lines define the relationships between the classes, commonly known as associations. The class diagram's purpose can be summarized as follows: a) Analysis and design of an application's static view. b) It represents a system's responsibilities. c) It serves as the foundation for component and deployment diagrams.

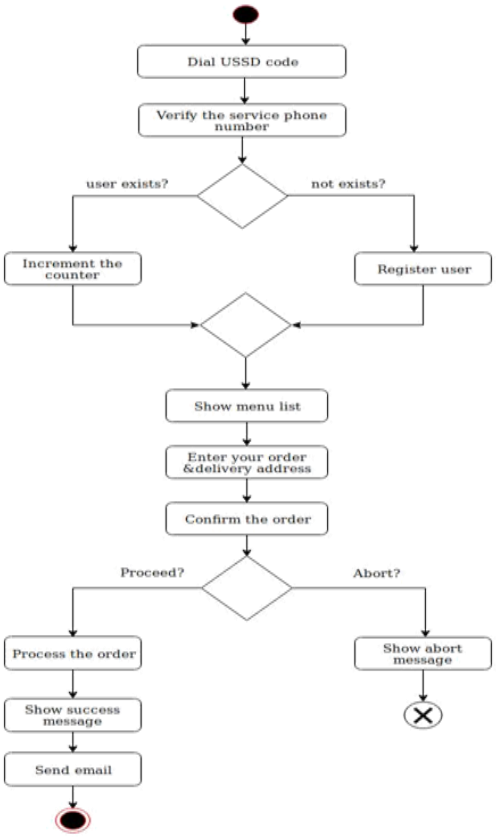

Figure 3 Activity Diagram for food Ordering System

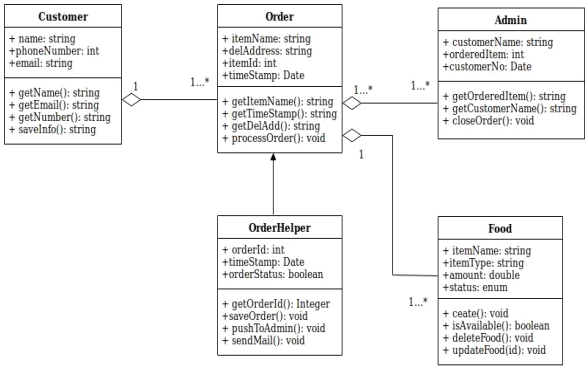

Figure 4 Class Diagram for food Ordering System

4. SYSTEM IMPLEMENTATION

The flow of this real-time food ordering system is made up of five parts that work together to get the job done.

These are:

1. The Customer Authentication Module
2. 'Making Order Request
3. Mailing Service
4. Rendering of Orders on Dashboard
5. System Deployment

4.1 The Customer Authentication Module

The first phase of the application, after dialing the USSD service code, will automatically detect the user's mobile phone number. Then customer account registration takes place for the first time.Subsequent dialing will not require user account registration. The user account registration requires the user to enter his/her name and email. The API, which is the logic of the system written in Nodejs, will take the request (name and email) made by the user and validate it before saving the data into the Mongo database. This system usedthe cloud version of Mongo, which is referred to as Mlab. Figures 5 to 7 show the customer authentication screens.

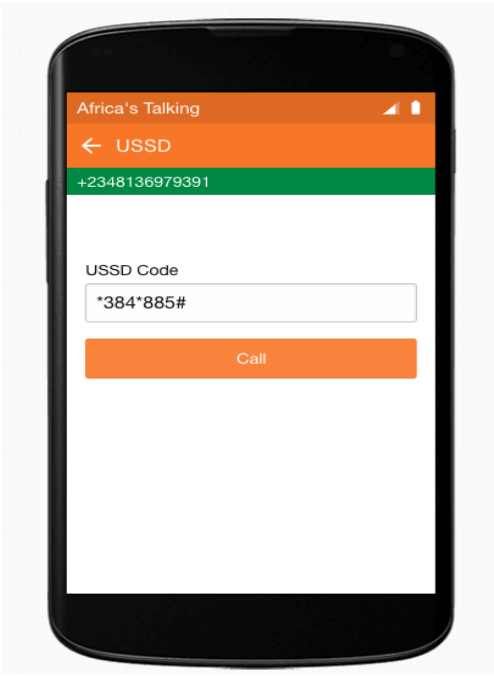

Figure 5. USSD Dialing code page

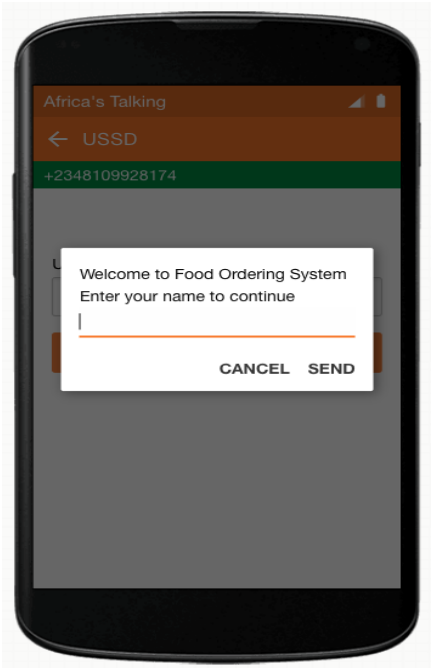

Figure 6. Customer Authentication page 1

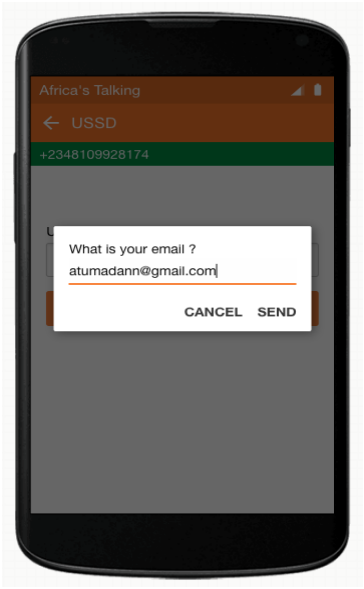

Figure 7 Customer Authentication page 2

After a successful registration of the user, the user proceeds to the ordering of items. This phase is similar to the initial process using the same technology. An event will be listening on to take the user's response and save it to the database, but at this stage, the third party service Pusher will take the data gotten from the user's response and get it pushed to the client's side, which is the admin dashboard. This establishes the real-time service, so that the admin can see the effect of the order in real time, even before the persistence of data.

4.2 Mailing Service

At the point of saving the order, the system will initiate the mailing service embedded in the back end logic. The system will take some user's information such as email and name for mail sending. This mail is just a confirmation or a receipt that the order has been made successfully. A sample e-mail generated is shown in figure 8.

Figure 8.Sample Sent Mail

4.3 Rendering of Orders on Dashboard

The Pusher will append the newly made order to the dashboard and the database which is mongodb will be in support after the effect of Pusher, because Pusher can only hold data temporarily, after the admin window refresh, the data will no longer be displayed on the client, but to prevent

this, the persistent data will come from the database and admin will not notice the effect of this Pusher data loss.

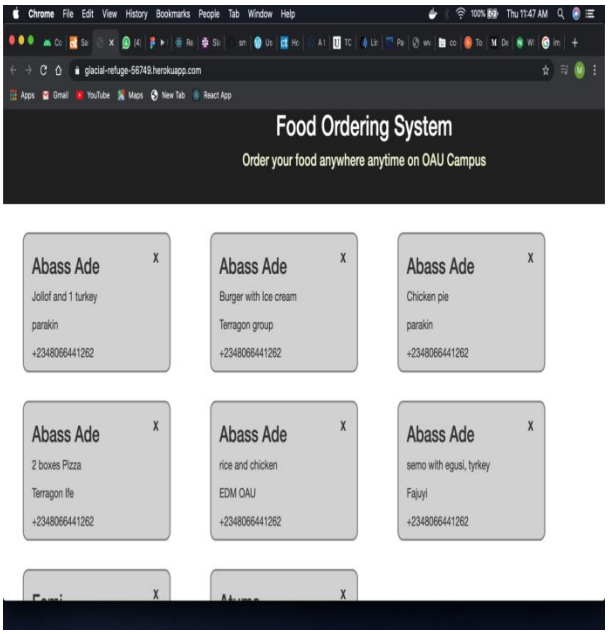

Figure 8. Admin Dashboard of the Food Ordering System

4.4 System Deployment

Because of the cost, the USSD part of the project is being deployed on Africastalking Sandbox, this sandbox is an online simulator as the sample is shown on figure 5, 6,7 and 9 and 10 below, it allows us to test the service without the proper paid hosting, and the API and the admin dashboard is hosted on Heroku. It is a CICD tool used for app deployment

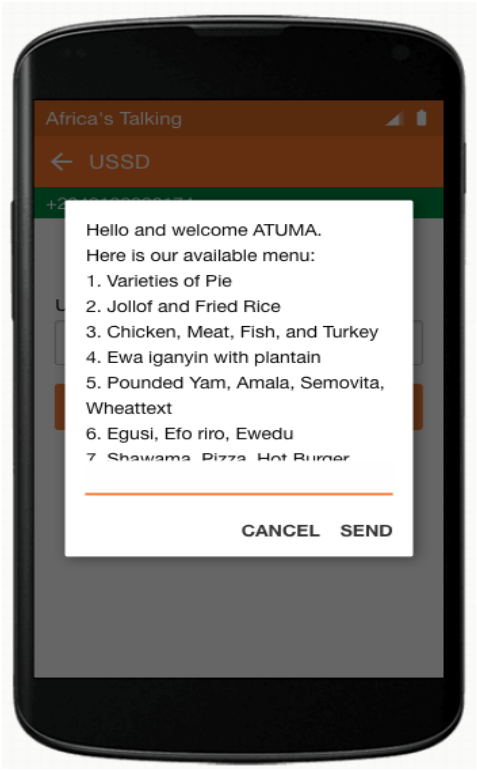

Figure9. Food Ordering page 1

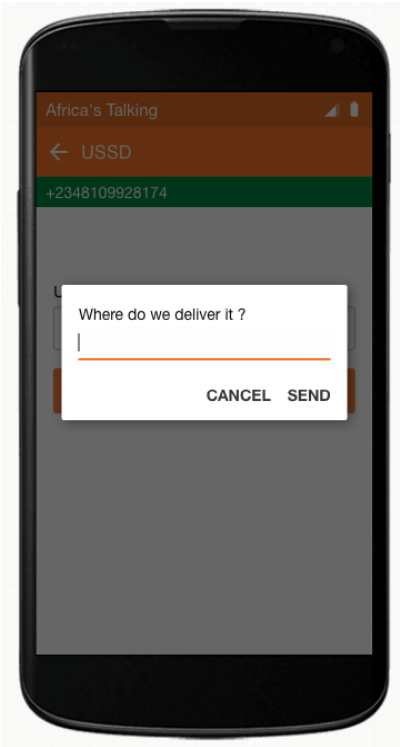

Figure10. Food Ordering page 2

5. EVALUATION

According to [13], web applications are often evaluated based on learnability, ease of use, effectiveness, efficiency, and satisfaction of users, comprehensibility, error frequency, and other usability criteria depending on the nature of the application. Some of the criteria mentioned were used to evaluate the food ordering mobile application's performance and usability. The criteria used are highlighted below.

a)Ease of Use: This refers to how easy it is to use the Application

b) Learnability: This refers to how quickly the user learned master the application

c) Effectiveness: This is the degree to which the application enhances the customer and the restaurant owners in ordering and processing orders.

d) Error: This was used to assess how often users encountered errors while trying to perform various tasks with the application. Errors include crashes, etc.

The evaluation of the system was done by asking people to register on the platform by dialing *384*885# on the sandbox used as a test for the USSD. Six people registered as customers. They were all asked to use the USSD code to perform system functionality available on the platform. The evaluators were interviewed and asked to evaluate the system based on the selected criteria after a few days. This is shown in Table 1.

Table 1: Results of Evaluation

Evaluator	User type	Ease of Use	Learn ability	Effecti veness	Error
A	Restaurant admin	7	8	7	2
B	Restaurant admin	9	7	7	0
C	Restaurant admin	5	8	5	3
D	Customer	6	5	6	3
E	Customer	8	9	9	0
F	Customer	6	7	7	1
G	Customer	9	8	8	0
H	Customer	6	7	6	2
I	Customer	8	7	8	0
Average		7.11	7.33	7	1.2

5.1 Interpretation of Results

From the outcomes, the mean ratings for each category were calculated, and the results demonstrate that the mean scores for ease of use, learnability and efficacy are all higher than the national average(7.11,7.33, and 7 respectively).This means that the software and the system as a whole were successful in achieving their stated goals and objectives. The low mean error score (1.22) also indicates that the evaluators had little difficulty utilizing the application. It was discovered that assessors who had no "mistake" provided high scores on all other evaluation categories. Testers who encountered issues while using the app received lower scores on average. This means that by reducing errrors, learning ability, convenience of use and most importantly effectiveness can all be increased.

5.2 Conclusion

This work has been able to use USSD technologies to connect restaurant owners and customers together. It can be further improved by integrating a payment gateway that will allow you to pay as you order, also by building a system that helps the customers locate and choose nearby restaurants,sends an order notificationand handles the logistics of getting the food to the customer, by tracking the movement of delivery men. Based on the results of this work, eating food on campus can be enhanced by using USSD applications to connect and facilitate communication between the restaurants and customers.

6. ACKNOWLEDGMENTS

This Research was funded by the TETFUND Research Fund and Africa Centre of Excellence OAK-Park, Obafemi Awolowo University, Ile-Ife, Nigeria.

7. REFERENCES

[1] Adithya R., abhishek S., Salma P. (2017) Online Food Ordering System. International Journal of Computer Applications, 180(21) pp18-24.

[2] Ansel. D, and Dyer. C, (1999).A framework for Restaurant Information Technology.

[3] Bernard S.K. and Tanya G.M.(2018) An Android-based Order Placement System for Restaurants, International Journal of Computer Applications,180,18-24.

[4] Fan Yang, 2014, Mobile Food Ordering App, Vaasan Ammattikorkeakoulu University of Applied Sciences,Vol no1, page no. 6.

[5] Gupta, P.,(2010) End to End USSD Implementation, TATA Tele Service Ltd: India.

[6] Hirschberg,C.,Schumacher,T., Wrulich,M. and Rajko, A.(2016). The changing market for food delivery Mckinsey& Company High Tech. Available at http://www.mckinsey.com/industries/high-tech/our-insights/the-changing-market-for-food-delivery (accessed 25.02.2018)

[7] Nibusinessinfo.co.uk (2015) Advantages and disadvantages of online retail.[Online] Available at: http://www.nibusinessinfo.co.uk/content/advantages-and disadvantagesonline-retail [Accessed 26 Oct. 2017]

[8] Y.T.Pankaj K. (2014). Development of Home Delivery System in South Ostrobothnia, Degree programme

[9] Rohini, G., Sonali, J., Ashravanti, L. and Omkar, K. (2021) Online Tool Ordering Application. International Journal of Advanced Research in Computer and Communication Engineering, Volume10, Issue 4,pg 203-205.

[10] Sanganagouda, J.(2011) USSD-A Potential Communication Technology that can Ouster SMS Dependency. International Journal of Research and Reviews in Computer Science, 2(2):p2-10.

[11] Tanga, G. M. and Bernard, S.K.(2018) An Android based Order placement System. International Journal of Computer Applications. Volume 180, no21, pg 18-24.

[12] Wen. Y, Ying. X, Zhao, X and Li, M.(2017) An online food ordering system for people in University. School of Computer Science University of Birmingham, United Kingdom.

[13] Zhang, D., and Adipat, B.(2005). Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications. International Journal of Human-Computer Interaction, 18, 293-308.