Grenze International Journal of Engineering and Technology, July Issue

A Low-Cost Implementation of an IoT based Food Ordering System

Arthav Mane[1], Janhavi Bhopale[2], Raksha Jain[3] and Priya Chimurkar[4]
[1-4] Department of Electronics Engineering, Sardar Patel Institute of Technology, Mumbai, India
Email: {arthav.mane, janhavi.bhopale, raksha.jain, priya.chimurkar}@spit.ac.in

***Abstract*—This paper proposes the use of an IoT-based food ordering system that would be a very actionable and profitable system in the coming age of digitization. This system functions to automate and simplify the entire process of ordering food in public places such as restaurants, food courts, etc. wherein it would aid in reducing the load on manpower and fastening the service at all times, whilst making the entire process cost-effective for the restaurants. In a nutshell, the system constitutes various on-table devices that connect each table to a monitor display in the kitchen via WiFi. A device, with TFT touch display mounted on it and connected to the NodeMCU, on each table enables the customer to place the order conveniently which is further sent across on a web server using the inbuilt ESP8266 WiFi module on the NodeMCU which is displayed on the monitor installed in the kitchen.**

***Index Terms*— IoT, NodeMCU, restaurant, automation, Web Server, Food Ordering.**

I. INTRODUCTION

Technology and improved lifestyle plays a very important role in the development of the world. With the increasing digitization in every field, it is necessary for all domains to adapt to this change, in order to sustain and survive in this era of immense competition all around. The present methods of food ordering at busy restaurants are inefficient, prone to human error, and with the increasing cost of living, dependent on expensive manpower. If we consider the current usual system of food ordering in restaurants, an attendant typically spends about a minute at a table where he either stays at the table and writes the first phase of the order or gives them more time to come to a decision. The next visit of the attendant at the table happens either after being called several times or a follow up after sometime. Furthermore, in metropolitan cities, several occasions arise where the waiting list for getting a table in a restaurant is long and the customers can only order their food once they have obtained a table.

An IoT based food ordering system would be a very useful creation in the coming age of digitization, where it would reduce the load on manpower and provide instant services at all times at the customers' convenience while also providing an interactive experience to the user with the help of a graphical interface [1]. Overall, the idea is to develop a system of numerous server-connected devices in order to automate the entire process of ordering food in a restaurant, food court, or any other public eatery. While doing so, the focus is also on considering the financial boundaries making it economically and technically efficient.

By understanding the current methodologies and techniques of food ordering at restaurants, the major

Grenze ID: 01.GIJET.6.2.11_1

objectives of this innovation is to create a system that enhances the overall user experience by providing an interactive graphical user interface on the device. It is also aimed at contributing towards cost efficiency of small-scale as well as large-scale food outlets to lead them towards automation.

II. Prior Related Work

Over the years, the restaurant food ordering systems have been confined to the involvement of an attendant utilizing the pen-paper method for taking down the order. And in this coming age of digitization, this traditional method of food ordering undoubtedly leaves room for an opportunity for improvement and digitization of the whole process.

The idea of automating the entire food ordering system while attempting to enhance user experience initially involved the use of computer screens [1] on each table which wasn't very cost-effective for most of the restaurants. Similarly, numerous attempts were made to develop completely software-based means for ordering food that heavily involved the use of tablets or smartphones, either belonging to the user or provided on every table [2][3][4]. However, again, these methods weren't cost-effective. On-table devices involving various components such as Advanced RISC Machine (ARM) microcontrollers, Liquid Crystal Display (LCD) and Zigbee were also proposed which rendered the device to be bulky while increasing the total cost of the system [5][6]. Furthermore, Ref. [7] and Ref. [8] combined the use of microcontrollers and smartphones to deliver an automated food ordering system. Another food ordering system that was proposed was in Ref. [9] involving the use of bluetooth modules in the on-table devices which restricted the connectivity and coverage area of the system on account of the limited bluetooth range. Keeping in mind all these work samples, an attempt at coming up with a compact and low-cost implementation of a similar food ordering system has been made.

III. System Overview

The basic idea behind implementing the proposed food ordering system is to make the current ordering system much more efficient and automated with the use of current technologies such as the Internet of Things (IoT). The system comprises numerous on-table devices and a monitor screen that is installed in the kitchen. The on-table device is the one on which the user views the menu, makes a selection of the food items that are to be ordered and sends it over to the server after confirming the order. The server is then loaded in the kitchen via the fixed IP address of the server and the entire order is displayed on the kitchen's monitor screen. The block diagram depicting the components and the basic idea of one on-table device of the system has been shown in Fig. 1.

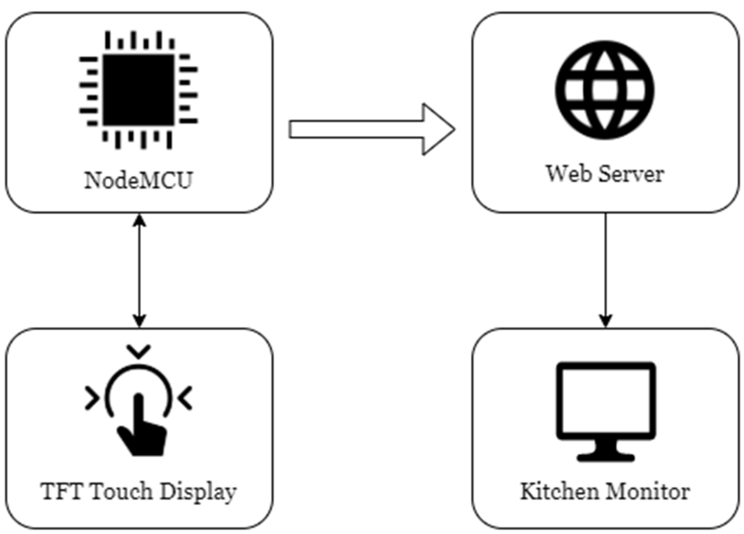

Figure 1. Block diagram of the system

IV. Implementation and Methodology

A. Components Used

- TFT LCD Display: A TFT (Thin Film Transistor) LCD display has been used in the on-table device. The display contains a touch panel which provides a user-friendly interface to the customer. Equipped with a touch screen display of 480x320 pixels and ILI9488 LCD driver IC [10], the TFT display supports Serial Peripheral Interface (SPI) and a power IC with an SD card socket. This device was found to be ideal for the project due to its dimensions and resolution. It fitted perfectly well, ergonomically while enhancing the systems user experience.
- NodeMCU: NodeMCU Dev Kit v1.0 (Version 2), having the on-board ESP8266 [11] WiFi System on Chip (SoC) from Espressif and an on-module flash-based SPIFFS file system, has been used in every on-table device. This single-board microcontroller is the cheapest WiFi enabled development kit available in the market and thus, is a perfect choice to use in the system for every device.
- Arduino IDE: Arduino IDE (Integrated Development Environment) Kit, a software that enables assisted code editing, compiling and debugging, has been used to program the NodeMCU and develop the TFT Graphical User Interface (GUI).

B. Connection and Programming

The entire experimental setup constitutes the connection between a TFT touch-screen display, NodeMCU, an external memory module (if required) and a 5V battery. The connection between the NodeMCU and the TFT has been established using the SPI connection. Two different connections are required for the NodeMCU to be successfully interfaced with the TFT. One SPI connection is between the NodeMCU and the ILI9488 driver IC on the TFT display, and the other connection, between the NodeMCU and the XPT2046 [12] touch-panel IC. The NodeMCU development board has been powered by connecting a 5V battery to its voltage supply pins. The TFT touch-screen display is powered by the NodeMCU by connecting their voltage supply pins to each other in the SPI interface connection. An external memory card is also connected with the TFT display to store additional files. For applications that require larger memory, an external memory module can be connected to the NodeMCU as well.

The NodeMCU has been programmed using the Arduino IDE. For easy SPI interfacing and extended functionality, the Bodmer's TFT_eSPI [13] library is used. The library by default works with ILI9341 driver IC, but the library is specifically configured to work with the ILI9488 LCD driver IC and the XPT2046 touch-panel driver IC, by us. All the NodeMCUs used for the devices have been programmed to start a WiFi server with a specific IP address to avoid the loss of connection between the device and the kitchen server. The graphical user interface for the TFT touch-screen LCD display is also programmed on the Arduino IDE. The device has been programmed to take in the order from the user via the touch display and send it across onto the server via NodeMCU. The programming was done in accordance to the flowchart shown in Fig. 2.

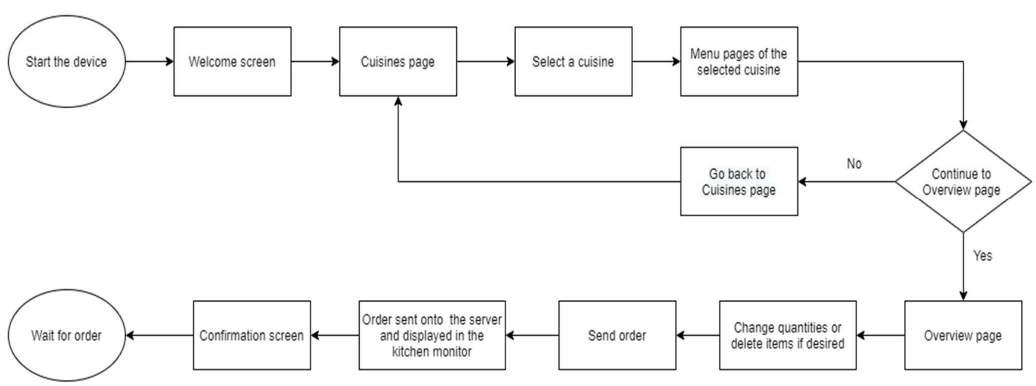

Figure 2. Flowchart of the ordering process in the system

C. User Interface

The graphical user interface has been provided via the TFT touch-screen LCD display [9]. The display acts as the first medium between the user and the menu of the restaurant and forms the first step in the order data acquisition. It is through this interface, that the user navigates through the menu and places the order. This entire process has been shown in Fig. 2.

On starting the device, the first page that is displayed is the welcome screen, after which the cuisines page is displayed as seen in Fig. 3. On selecting a cuisine, the user is directed to the first menu page of the selected cuisine. The display provides the functionality of navigating through the menu pages within the cuisine and also enables the user to go back and select another cuisine if desired. Once the user taps on a certain item, the touch coordinates are collected on sensing the touch and the item is selected. An overview page, as seen in Fig. 4, is displayed at the end of the selection process on which the user is shown the entire list of items and is provided with an option to send the order to the kitchen. It is, on this overview page, that the user can also alter the quantity of the items to be ordered. Once the user confirms the order to be sent to the kitchen, the order data is collected by the NodeMCU and sent on the web server. After the order is sent across and has been received by the web server, a confirmation screen is displayed thereby notifying the user of the successful transmission of the food order into the kitchen.

The order information, which is confirmed by the user to be sent, is collected by the NodeMCU and sent across to the kitchen on the web server created by the on-board ESP8266 WiFi module [14]. The web server is configured to be created with a fixed IP address. The web server, using the same IP address, is loaded in the kitchen and the order is seen via the monitor display that is installed in the kitchen.

V. TESTING

In order to test the program and tweak it to get a higher speed, accuracy, compatibility and efficiency, the system has been tried on various WiFi networks and servers with different speeds. This has been done to ensure that there is no bias in the connectivity to a server while developing the code. Different case scenarios such as prolonged waiting periods to place the order and orders with a large number of items, have also been tested. Quick navigation through the menu pages has been ensured by considering the screen refresh rate, considering the switching frequency of the panel and carrying out various reformations in the code. This led to a satisfaction that the problem of time being wasted due to the actual attendants, was tackled successfully. Validation of the results obtained has been done by checking if the correct order reaches the screen in the kitchen through the web server on the correct IP address every time without much delay.

Figure 3. Cuisines page

Figure 4. Overview Page

Furthermore, the validation of functioning of the WiFi module and the TFT LCD touch display has been also done, by working on various touch coordinates, for a better and more user friendly interface. The interface and front end of a device is just as important as a smooth back end processing; this ensures that the device is used by the larger market with no difficulty. The device was further developed to have a detailed interface and user friendly that has been made considering the needs and ergonomics of an individual. Furthermore, the working of the entire system has also been tested by configuring a few on-table devices to the same IP address and sending the order onto the web server. All the orders were displayed on the server while also mentioning a specific table ID given to every device, making it easier to recognize which table actually sent that particular order.

VI. RESULT

After extensive and multiple rounds of testing, the accuracy, efficiency and the overall user experience of the food ordering system has been improved to its maximum possible potential. Efficiency in terms of developing a very user friendly interface for ordering with multiple cuisine pages, minimum time required to change the various menu pages, displaying its overview in the end, establishing contact between the devices and the monitor display in the kitchen, and finally, sending the correct order to the kitchen successfully, has been achieved. The device was aimed to be low-cost and hence, as described earlier, the components used are quite inexpensive which makes the device cost less. Hence, it is quite affordable on a commercial scale.

VII. CONCLUSION AND FUTURE SCOPE

The main objective was to create a highly cost-effective, compact, smart and efficient IoT based food ordering system, considering the shortcomings and the problems faced by the current ordering system. With an ideology of building a wireless multiple-device system, the whole process involves coordinate calibration of the touch screen, optimization of the refresh rate of the screen, start-up of a web server with a specific IP address, fast transmission of order to the kitchen. The mentioned stages are found to take place in a hassle free manner without lagging of the device and software in the presence of a moderate strength WiFi connection. There is quite a room for improvements as far as this type of ordering system is concerned. A connection between the monitor inside the kitchen and the device can be added, for communicating messages to the customers regarding their order such as the estimated time of arrival of the items that have been ordered. Furthermore, a new section of special notes for the cook/chef can be introduced, wherein the customers can mention their special requirements. Finally, a summary of the bill can be shown on the device, once the customer is done. These are some features which have the ability to make this device even more efficient and user- friendly.

ACKNOWLEDGMENT

This research was supported by the Department of Electronics Engineering, Sardar Patel Institute of Technology (S.P.I.T.). We are also immensely grateful to the entire teaching and non-teaching staff of the Electronics Department, S.P.I.T., for their support and guidance during the course of this project. We thank them for their assistance and comments that greatly helped us in the entire process of implementing this project and improved this manuscript.

REFERENCES

[1] Y. Tan, K. Lee, Z. Khor, K. Goh, K. Tan, and B. Lew, "Automated Food Ordering System with Interactive User Interface Approach," *IEEE Conference on Robotics, Automation and Mechatronics*, Singapore, 2010, pp 482-485.

[2] S. Tanpure, P. Shidankar, and M. Joshi, "Automated Food Ordering System with Real-Time Customer Feedback," *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)*, vol. 3, issue 2, 2013.

[3] H. Saeed, A. Shouman, M. Elfar, M. Shabka, S. Majumdar, and C. Horng-Lung, "Near-Field Communication Sensors and Cloud-Based Smart Restaurant Management System," *IEEE 3rd World Forum on Internet of Things (WF-IoT)*, Reston, VA, 2016, pp. 686-691.

[4] Adithya R., A. Singh, S. Pathan, and V. Kanade, "Online Food Ordering System," *International Journal of Computer Applications*, vol. 180, no. 6, 2017.

[5] V. Swapna and M. F. A. Khan, "Design and Implementation of Ordering System For Restaurants," *International Journal of Engineering Research and Technology (IJERT)*, vol. 1, issue 10, 2012.

[6] H. Wabale, G. Honrao, U. Deshmukh, and M. Chavan, "Automatic Menu Ordering System Using Zigbee and Arm Processor," *Journal of Electrical and Electronic Systems*, vol. 6, issue 2, 2017.

[7] M. Jakhete and P. Mankar, "Implementation of Smart Restaurant with e- menu Card," *International Journal of Computer Applications*, vol. 119, no. 21, 2015.

[8] M. Nandre, D. Patil, K. Patil, and J. Suryavanshi, "IOT based Restaurant Automation System," *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, vol. 6, issue 10, 2018.

[9] A. Cotta, T. Naik, A. Dias, N. Sawant, and A. Raikar, “Smart Restaurant Menu Ordering System,” *International Journal of Science Technology and Engineering (IJSTE)*, vol. 2, issue 9, 2016.
[10] ILI9488 Driver, https://focuslcds.com/content/ILI9488.pdf, Accessed: October 2019.
[11] ESP8266 Wi-Fi Module, www.espressif.com/sites/default/files/ documentation/0a-esp8266ex datasheet en.pdf, Accessed: August 2019.
[12] XPT2046 Touch Screen Controller, https://ldm-systems.ru/f/doc/catalog/ HY-TFT-2,8/XPT2046.pdf, Accessed: October 2019.
[13] Bodmer’s TFT eSPI library, https://github.com/Bodmer/TFT eSPI, Ac- cessed: September 2019.
[14] Vamseekrishna A., P. Jyothshnasri, V. Aiswarya, and S. Sameer, “Smart Restaurants,” *International Journal of Engineering and Technology (IJET)*, vol. 7, pp. 54-57, 2018.