TELCOMATICS, Vol.7 No. 1, June 2022, pp.8-13
e-ISSN : 2541-5867
DOI: 10.37253/telcomatics.v7i1.6797

Prototype of Automatic Cafe Management System (ACMS) Based on Internet of Things (IOT)

Sabariman*, Erris Fernandy
Faculty of Industrial Technology, Universitas Internasional Batam, Batam, Indonesia
*e-mail: *sabariman@uib.ac.id*

ABSTRACT

The COVID-19 Pandemic changes people's lifestyle nowadays. Physical interactions between person have lessened due to laws passed by central governments concerning its limitations. It's no surprise this impacted businesses that needs face-to-face meetings, such as cafés or restaurants. Based on such facts, prototype for Automatic Café Management System (ACMS) based on Internet of Things (IOT) is designed. ACMS allows customers to order food through web-based system that can be accessed with their gadgets. The system is equipped with a line follower robot to act as waiter/waitress. With such system, physical contacts in café or restaurant settings can be reduced significantly.
Keywords: Cafe, ACMS, Web-based, IoT

ABSTRAK

Pandemi COVID-19 mengubah gaya hidup masyarakat saat ini. Interaksi fisik antara orang-orang telah berkurang karena undang-undang yang disahkan oleh pemerintah pusat mengenai batasannya. Tidak heran jika hal ini berdampak pada bisnis yang membutuhkan pertemuan tatap muka, seperti kafe atau restoran. Berdasarkan fakta tersebut, dirancang prototipe Sistem Manajemen Kafe Otomatis (ACMS) berbasis Internet of Things (IOT). ACMS memungkinkan pelanggan untuk memesan makanan melalui sistem berbasis web yang dapat diakses dengan gadget mereka. Sistem ini dilengkapi dengan robot line follower untuk bertindak sebagai waiter/waitress. Dengan sistem seperti itu, kontak fisik di kafe atau restoran bisa dikurangi secara signifikan.

I. INTRODUCTION

The COVID-19 Pandemic has changed multiple ways people interact. Limitations of physical interaction due Health Protocols enforced by goverments across the globe had shaken the normal way of life significantly. For some businesses, the change in lifestyle may be beneficial, but more often than not, it is damaging. One of the easiest example of this is the food and hospitality industry. The industry itself inherently requires above average human contact, and being limited in that regards surely would affect it negatively.

According to a study published by Ashima Bansal and Rajiv Kumar Sardhana[1], during the COVID-19 pandemic, food and hospitality industry across the globe has been hit hard economically. This includes but not limited to restaurants, cafes, and hotels. From the same study, the authors pointed out that the biggest factor for the regression of the industry is physical contact limitation [2].

To help limit the interaction between humans in such settings while maintaining functionality, an understanding of the interaction itself is needed. In most restaurants, physical contacts can be divided to between patrons, between staffs, and between both. Logically speaking, the staffs would experience more human contacts, especially the waiters/waitresses, as they have to face multiple customers each day. Therefore, it's in the best interest to limit the interaction between staffs, especially waiters/waitresses, and patrons.

e-ISSN : 2541-5867
DOI: 10.37253/telcomatics.v7i1.6797

Based on such facts, a Prototype of Automatic Cafe Management System is devised. The aim of this system is to reduce physical contacts between humans in cafe and/or restaurant settings as much as possible. The system achieves this by offloading most work done by waiter/waitress to a web-based system designed to take customers' orders, show tickets to kitchen staff, and provide billing automatically to the cashier, and a line follower robot to bring the food to the customers.

II. Literature Review

In a paper titled "Automatic Cafe Management" authored by Abbas et al[3], the authors designed a automatic cafe system using what they called *electronic food bar* as an instrument to order. The device consists of a 16x2 character LCD display and a keypad. The customers are to use the device to place an order which will be sent via bluetooth to an identical device held by the kitchen staff. The authors also designed an Arduino-based, three-wheeled line follower prototype robot. The navigation method chosen for the robot is a simple decision table to tell the robot which direction to turn. The resulting system is relatively cheap and very simple to create, and thus is economically viable. That being said, the *electronic food bar* is not sufficient for this day and age, and the robot itself suffers some stability issues due to the three-wheeled design.

From a journal written by Kamruzzaman[4] titled "Design and Implementation of A Robotic Technique Based Waiter", the author designed a system intended to deliver drinks to guests. Instead of a custom device, the author opted for an Android application to place an order. The order will then be sent to the robot via bluetooth. The robot in question is an Arduino-based robot with similar configuration as before, but is also equipped with weight sensor to detect whether the robot is carrying an item or not. As for movement, the method chosen is *mapping*. Using a rotary encoder, the robot can determine the distance it has traveled. Using a set of pre-programmed distance values and turning directions, the robot can arrive at their destination. The result of this research is a system that is very easy to use due to the usage of an Android app instead of a custom hardware, and also very easy to implement due to the movement method chosen. The main drawback of this system is its poor scalability. The usage of app limits the access to only people with the apps installed, and thus can be a hassle. The *mapping* method chosen also doesn't scale well as the environment gets bigger and number of destinations rise.

In a journal titled "Restaurant Serving Robot with Double Line Sensors Following Approach" authored by V. Thanh et al [5], the authors designed a working food delivery robot for restaurant settings. The robot designed is a four-wheeled line following robot with two line sensors array, with one of them in front and the other in the back. Coupled with the PID controller implemented by the authors, the result is a mobile robot with high degree of stability, perfect for delivering food and drinks to customers in a restaurant and/or cafe settings.

III. Research Methods

The method that will be used in this journal are literature study and experiment. Literature study consists of studying past journals to discover the best possible combination of parameters in order to address the problem efficiently. Experiments then are conducted by building a new system that fits under said parameters in hope of creating a suitable or better solution to the problem.

The system can be divided into two parts: web-based system and robot. The web-based system consists of web pages hosted on the internet and a small database to handle customers' orders, creating tickets, and writing receipts. The robot is a prototype Line Following robot with WiFi capabilities.

The web-based system has four main workload: taking orders from customers, create tickets for kitchen staffs, call the robot, and write receipts for cashier. The customers will access the "order page" to place their orders. The orders are then sent into a database, which will be read by "kitchen page" in order to create food tickets for the kitchen staffs to work on. Once the food has finished cooking, the kitchen staffs will mark the ticket as done, and the system will automatically prepare the robot for delivery. After the robot has finished the

TELCOMATICS, Vol.7 No. 1, June 2022, pp.8-13
e-ISSN : 2541-5867
DOI: 10.37253/telcomatics.v7i1.6797

delivery, the ticket's status gets updated in the database, and then will be read by "cashier page" to write appropriate digital receipt. After payment has been completed, the receipt can be marked as done, and a full transaction has been completed.

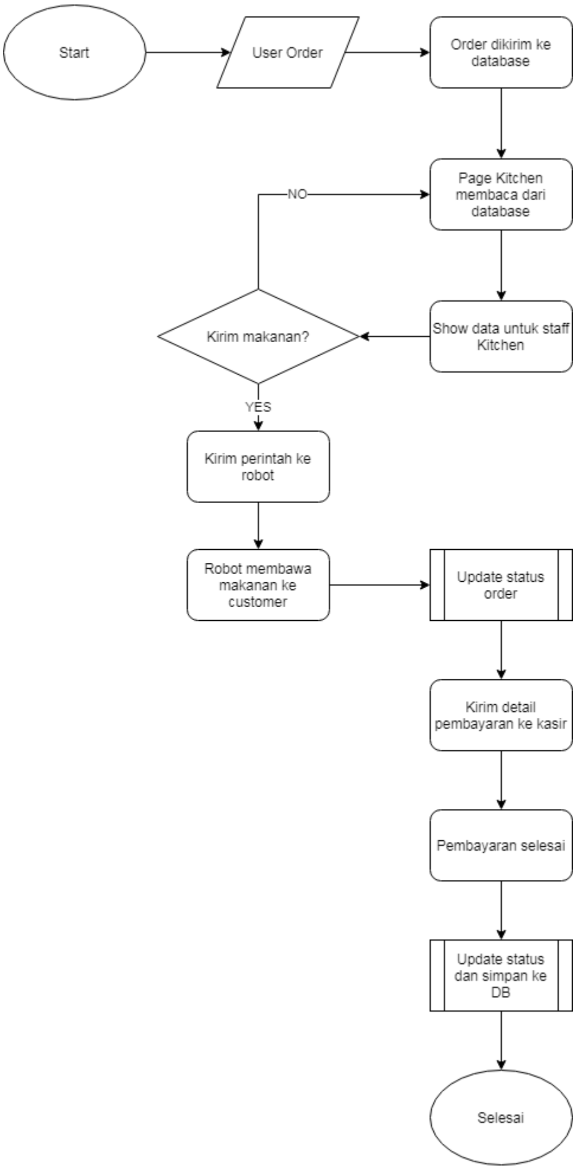

Figure 1. General Web-System Flowchart

The robot designed is a double line sensor following robot, with WiFi connection, weight sensor to detect load and PID control capabilities. The main controller of choice is an Arduino UNO. The hardware is chosen for its ease of usage and compatibility with wide assortments of modules.

The two line sensors are self-made, TCRT5000-based, 4 sensor arrays. The sensors are arranged with two inner sensors in the back to act as the main PID input, and two outer sensors further ahead to detect intersections and extreme deviations early. TCRT5000 is an infrared light sensor. It's constructed by having an infrared LED and a photodiode next to each other in a small plastic socket. The LED emits light, that will be bounced by the surface into the photodiode. The intensity of light depends on texture and color of the surface. This measured light intensity then can be used to distinguish between the lines and the floor.

WiFi connection is provided by an ESP-01 module. ESP-01 is a WiFi enabled board based on Espressif's ESP8266 chip. The hardware is small and compact, and can be used on its own (albeit very limited) or used as an add-on module to existing compatible hardware. In this case, the latter is utilized.

The weight sensor used is a bar-type load cell. A load cell is a piece of metal, typically aluminum, with a small patch of flexible resistor called strain gauge. As the metallic body bends, the strain gauge deforms and changes in resistance value. This value is then measured to detect forces applied on the sensor. Because the changes are rather subtle, load cells typically comes with an amplifier circuit to amplify the signal to reasonable levels. The amplifier used in this project is a HX711-based module.

For movement, four generic hobbyist grade gear motor will be used. They are chosen because they are good enough considering the robot will not be moving very fast when delivering items, and it is also relatively cheap and widely available. To drive the motors, a L298N motor driver module is used, for the same reason.

PID controller will be built right into the Arduino UNO itself. The robot is programmed to have four main states: idle, waiting, PID, and decision. In idle mode, the robot does nothing but checking the server from time to time, to see whether there's an order to be sent or not. If the robot finds a valid instruction, it will enter waiting mode. In waiting mode, the robot is on standby to deliver and is waiting for the weight sensor to detect a load on it. After that, the robot enters PID mode and starts moving forward with PID controller in charge of

e-ISSN : 2541-5867
DOI: 10.37253/telcomatics.v7i1.6797

movement. When the robot senses an intersection, it will stop for a moment and enter decision mode. In decision mode, the robot will turn left or right at the intersection depending on the instruction sent by the server. It will then re-enter PID mode until it reaches its destination. Upon arrival, the robot enters waiting mode again to detect when the load has been taken off. After detection, the robot will go back to its original position using PID and decision mode as before. When it reaches its initial position, the robot will go back to idle mode, and the cycle repeats.

IV. Results and Discussion

The web-based system is tested with trial runs of the whole process (ordering, sending, and payment) for few times.

The first step is to simulate customers' orders. To order, the customer needs to select table number, select the food wanted, and then click the order button. The order will then be added into the database with "Pending" as its status. Figure 2 is a screenshot of this step's attempt.

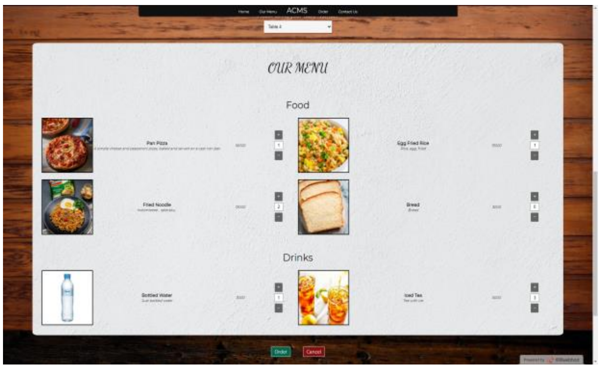

Figure 2. Ordering from the System

orders_id	orders_table	orders_item_id	orders_qty	orders_status	orders_start	orders_finished
1	4	1	1	PENDING	2021-02-10 15:27:11	NULL
1	4	2	1	PENDING	2021-02-10 15:27:11	NULL
1	4	3	2	PENDING	2021-02-10 15:27:11	NULL
1	4	5	1	PENDING	2021-02-10 15:27:11	NULL
1	4	6	3	PENDING	2021-02-10 15:27:11	NULL

Figure 3. Database Entry After Step One

Figure 3 shows the result of database entry aftar step one. Meanwhile, the kitchen webpage periodically checks the database and fetches any orders with "Pending" status. The orders fetched will then be arranged into tickets to show to the kitchen staffs. The kitchen staffs can then prepare the food according to the ticket created by the system, as seen in Figure 4.

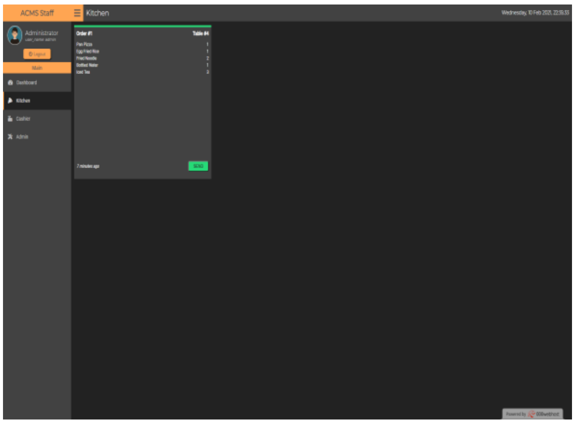

Figure 4. Kitchen System

When the staffs are ready to send the food, they may click the green "Send" button in order to call the robot. When the button is pressed, the order in question will have their status updated in the database, from "Pending" to "Sending". The database entry in this process shown in Figure 5.

orders_id	orders_table	orders_item_id	orders_qty	orders_status	orders_start	orders_finished
1	4	1	1	SENDING	2021-02-10 15:27:11	NULL
1	4	2	1	SENDING	2021-02-10 15:27:11	NULL
1	4	3	2	SENDING	2021-02-10 15:27:11	NULL
1	4	5	1	SENDING	2021-02-10 15:27:11	NULL
1	4	6	3	SENDING	2021-02-10 15:27:11	NULL

Figure 5. Database Entry After Step Two

Orders with "Sending" status will then be read by the Arduino page. The page consists of a string of text constructed with special syntax to send to the robot. The first number is the order number, with a path instruction following it. The path will be used to decide turning directions. The instruction example sent to Arduino shown in Figure 6.

|1|4ssrsst|

Figure 6. Instruction String for Arduino

TELCOMATICS, Vol.7 No. 1, June 2022, pp.8-13
e-ISSN : 2541-5867
DOI: 10.37253/telcomatics.v7i1.6797

After the robot has confirmed receiving and sending the order, the webpage will update the order's status from "Sending" to "Sent". The database entry after step three shown in Figure 7.

orders_id	orders_table	orders_item_id	orders_qty	orders_status	orders_start	orders_finished
1	4	1	1	SENT	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	2	1	SENT	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	3	2	SENT	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	5	1	SENT	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	6	3	SENT	2021-02-10 15:27:11	2021-02-10 15:50:53

Figure 7. Database Entry After Step Three

The orders with "Sent" status is then read by the cashier page. The cashier page will then create a receipt and show it to the cashier. This process can be see as in Figure 8.

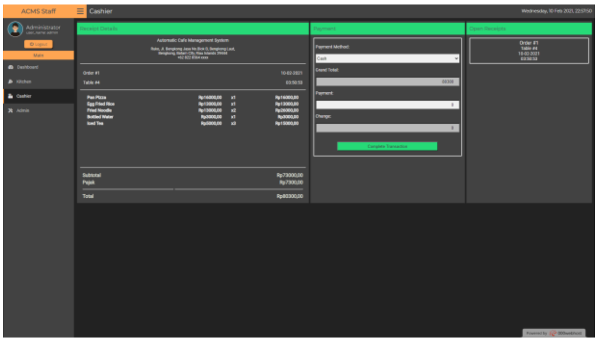

Figure 8. Cashier System

When the payment is done, the cashier can click on "Complete transaction" button to update the order entry one last time from "Sent" to "Completed", and the entire cycle is done. Figure shows the result of database entry after step four.

orders_id	orders_table	orders_item_id	orders_qty	orders_status	orders_start	orders_finished
1	4	1	1	COMPLETE	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	2	1	COMPLETE	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	3	2	COMPLETE	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	5	1	COMPLETE	2021-02-10 15:27:11	2021-02-10 15:50:53
1	4	6	3	COMPLETE	2021-02-10 15:27:11	2021-02-10 15:50:53

Figure 9. Database Entry After Step Four

As for the hardware, it has been decided that an Arduino based robot would be the most suitable candidate. The robot will also be equipped with a load cell, two TCRT sensor arrays, an LCD display, and an ESP-01 for communication.

Figure 10. Physical Form of Robot

The robot will be powered by four DC motors and a L298N driver. Ideally, the system will not stall. In such system, the motors can be controlled by sending a PWM signal from Arduino to the driver (ranging from 0-255). However, because of the motors' relatively high stalling current, workarounds to be devised. In order to kickstart the motors, the Arduino will send a high PWM signal for a brief moment and immediately switch to intended, typically lower PWM value. This eliminates the initial stalling. To remove the rest, a trick must be employed in determining the PWM value. The physical robot can be seen in Figure 10.

$$PWM = \begin{cases} Offset \pm OUTPID, \; PWM > threshod \\ -(Offset \pm (Offset - OUTPIC)), PWM < threshod \end{cases} \quad (1)$$

With a threshold as seen above in Equation (1), the motor will skip PWM value from negative threshold to positive threshold. Therefore, the motor will not stall as much as it was.

While being in subject of stalling, the motor can also stall by adding weight. Therefore, to help with that issue, the PWM value will be multiplied by a certain value when carrying weight as shown in Equation (2).

$$PWM(maju) = PWM * Mutiplier \quad (2)$$

To guide the robot, a simple circuit is devised as shown in Figure 13. With the red lines indicating targets (tables) and blue lines indicating start area. The robot is then tested on this circuit with consistent weight and varying PID values until the success rate reaches at least 95%. From the experiment, the parameter that is found to be suitable as in Table 1.

e-ISSN : 2541-5867
DOI: 10.37253/telcomatics.v7i1.6797

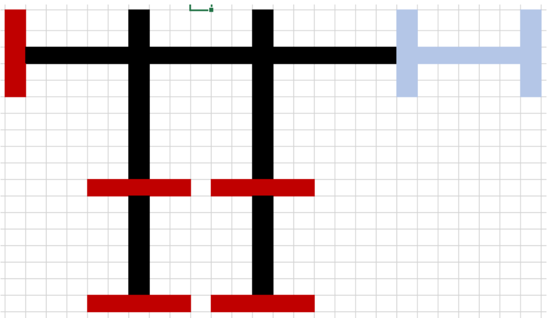

Figure 11. Figure 13. Test Circuit

With the red lines indicating targets (tables) and blue lines indicating start area. The robot is then tested on this circuit with consistent weight and varying PID values until the success rate reaches at least 95%. From the experiment, the parameter that is found to be suitable as in Table 1.

Table 1. Experiment Parameter

Weight	**~300gr**
Kp	20
Ki	0.3
Kd	0.6
Threshold	40
Offset	50
Multiplier	1.1

V. CONCLUSIONS

The web-system for Automatic Café Management System prototype has been tested and is working properly. The webpages show the information as intended by design, and the database can be updated without any problems.

VI. REFERENCES

[1] S. Mohapatra, V. Priyanka, S. Mohapatra, I. Kohli, and R. K. Mishra, “Impact of corona virus covid-19 on the global economy,” *Int. J. Agric. Stat. Sci.*, vol. 16, no. 2, pp. 771–778, 2020.

[2] Y. Yang, H. Liu, and X. Chen, “COVID-19 and restaurant demand: early effects of the pandemic and stay-at-home orders,” *Int. J. Contemp. Hosp. Manag.*, vol. 13, no. 12, pp. 3809–3834, 2020, doi: 10.1108/IJCHM-06-2020-0504.

[3] Z. Abbas, S. A. Khan, S. Abbas, A. Mazhar, and M. Gulzar, “Automatic Cafe Management System Using Waiter Robot,” *2019 2nd Int. Conf. Commun. Comput. Digit. Syst. C-CODE 2019*, pp. 211–214, 2019, doi: 10.1109/C-CODE.2019.8681023.

[4] M. Kamruzzaman and M. Tareq, “Design and implementation of a robotic technique based waiter,” *3rd Int. Conf. Electr. Inf. Commun. Technol. EICT 2017*, vol. 2018-January, no. December, pp. 1–5, 2018, doi: 10.1109/EICT.2017.8275250.

[5] V. N. Thanh, D. P. Vinh, N. T. Nghi, L. H. Nam, and D. L. H. Toan, “Restaurant Serving Robot with Double Line Sensors Following Approach,” *Proc. 2019 IEEE Int. Conf. Mechatronics Autom. ICMA 2019*, pp. 235–239, 2019, doi: 10.1109/ICMA.2019.8816404.