International Journal on Advanced Science Engineering Information Technology

Vol.13 (2023) No. 1
ISSN: 2088-5334

Developing Touchless Dispenser System Based on IoT to Support Hydration Needs for University Students in New Normal Phase in Indonesia

Fuhaidy Hafiizhan Ahnaf[a,*], Denny Sukma Eka Atmaja[a], Haris Rachmat[a], Muhammad Agung Hambali[a]

[a] *Industrial Engineering, School of Industrial Engineering, Telkom University, Bandung, 40257, Indonesia*
*Corresponding author: *fuhaidyahnaf@student.telkomuniversity.ac.id*

***Abstract*— Following the global pandemic of COVID-19, in August 2021, Indonesia achieved a total of 3.930.300 cases, the highest in Southeast Asia. However, the government is keen on promoting the new normal phase and planning to open schools and permit face-to-face learning, from elementary up to universities. This means that public facilities and infrastructures will be used and can be the medium for virus transmission, as it will require 48 to 72 hours for the virus to be inactive on those surfaces. This will make people reluctant to touch surfaces, especially when it comes to public facilities that can provide for their needs. One of the most important is the need for hydration which is often overlooked. About 25% of college students were found dehydrated, and 37,5% showed signs of it. Dehydration could prove a serious threat to health had it been overlooked and could affect physical and cognitive performance, having more effects on students and lectures, requiring both in their activities. To support the needs of hydration amidst the pandemic, this research developed a touchless water dispenser system using the waterfall model, utilizing a cloud database with ESP32, controlled by users through an android application. The design is easy and cheap to install, even on regular dispensers, making it an effective and efficient alternative public facility providing hydration service to support the new normal phase.**

***Keywords*— COVID-19; IoT; hydration; Android application; touchless systems.**

Manuscript received 13 Dec. 2021; revised 23 Apr. 2022; accepted 6 Jun. 2022. Date of publication 28 Feb. 2023

I. INTRODUCTION

The latest emergence of COVID-19 pandemic surely shocked the whole world with its high transmission rates, and Indonesia is no exception. The pandemic situation in Indonesia was officially announced in March 2020 and has been bagging 3.930.300 cases since then, ranking the country as one with the highest cases in Southeast Asia and fourth in Asia [1]. Furthermore, this situation is worsened by the recent Delta variant (B.1.617.2) outbreak which originated from India and is 60% more transmissible than the Alpha variant [2] in addition to its higher immunity evasion and vaccine resistance [3], this variant outbreak made the daily case in Indonesia skyrocketed to more than 50000 cases on July 21st [1].

Despite the still ongoing cases being high, the government of Indonesia is planning on moving into new normal activities by nationally opening schools and permitting face-to-face learning. Although the government still opposes the strict health protocol for the new normal phase and administering lots of vaccines, the general public still views this as a rash decision. According to the CDC (Centers for Disease Control and Prevention), COVID-19 commonly spreads during close contact, with the virus stable as short as three hours on tissue paper and seven days on an outer layer of a face mask and 48 hours to 72 hours on steel and plastic [4].

Public infrastructure (mainly railings) and facilities in Indonesia are commonly composed of steel and plastic, meaning that they can last three days before being inactive [5]. This will make people reluctant to touch surfaces, especially when it comes to essential public facilities that provide necessities like public eating and drinking.

This type of facility is important because it provides the hydration necessary for studying or working due to it affecting cognitive function, productivity, and health [6]. If the need for hydration cannot be fulfilled, the subject will suffer dehydration. Prolonged dehydration can prove a serious threat to human health, had it been overlooked, and could affect the physical and cognitive performance of those affected [7].

Research conducted at various southwestern universities in the United States has shown that one in four college students was dehydrated, with an additional 37,5% showing signs of dehydration [8]. The fact could be worsened due to the public reluctance to touch and use public facilities that provide hydration services, or even worse, people ignoring the COVID-19 transmission and just using the facilities as normal, which may end in increased confirmed cases. Therefore, there is a need to curb the spread of COVID-19 for the people who use drinking facilities when the new normal phase is implemented in Indonesia so that they can do their activities.

One example of how to restrain the spread of covid-19 is by implementing the internet of things (IoT) [9]–[11]. The Internet of Things or abbreviated as IoT is an advanced communication paradigm. The objects of everyday life will be equipped with devices able to communicate digitally and suitable protocol stacks that will enable them to communicate with each other. Users are becoming an integral part of the internet that nowadays has been utilized worldwide in everyday life [12] such as industrial automation [13]–[17], healthcare systems [18]–[20], smart transportation [21]–[24], agriculture [25], [26] and more.

Then, in this pandemic era, IoT proved to be able to assist in the prevention of covid-19 spread. IoT reportedly can be used for monitoring covid infected patients by monitoring their temperature and blood pressure [27]–[30], help covid-19 diagnosis and treatment [31]–[33], and with the help of artificial intelligence (AI), covid-19 detection can be done via X-ray and CT images [34]. Hence, internet of things (IoT) technology will be built for public drinking facilities so that the people who use these facilities won't get infected by covid-19.

The methodology used for building the proposed IoT system is the waterfall method of approach, which offers a free-flowing, sequential development process. This method is considered to offer a well-defined set of criteria and requirement indications before actually starting the design phase and implementation of the project, thus, offering a basic plan of the project before starting and continuing in an orderly sequence of phases [35].

The waterfall method, although used by many renowned software developers, is a good method for personal projects in which the requirements are defined at the start of the project, allowing the study to continue through its developing stages without the interference of ever-changing requirements, making it a good consideration for this study [36]. The result of this research will be a prototype of an automated dispenser system with integration of android application and IoT based on cloud that will serve as an alternative public facility providing hydration service to support the new normal phase.

II. Material and Method

A. System Requirements

The system requirements are made by identifying the system features and its capabilities. It will then set the least system requirements needed to be developed as a main baseline of the touchless dispenser system. The details of system requirements are described in Table 1.

TABLE I
SYSTEM REQUIREMENTS TABLE

No	Requirements	Description
1.	User Registration	Users able to register their account
2.	User Login	Users able to log into the application
3.	User logout	User are able to end their session by logging out from their personal account
4.	User Profile	User are able to see their profile based on the database within the application
5.	QR code scan	User are able to scan the unique QR code placed in the dispenser outlet which allows them to use the dispenser
6.	Retrieve outlet information	Users can see detailed information about the outlet after scanning the QR code on the dispenser.
7	Choosing water quantity	Users are able to get options of the desirable quantity of water
8.	User request processing	Users can obtain a confirmation of their request within the android application, and the dispenser will proceed with the request after the user confirms it.
9.	Water discharged counter	The dispenser system is able to calculate the water discharged and stops it once it reaches the requested quantity/amount
10.	Display request history	Users are able to see and track their purchase record alongside the purchase details

B. System Design

The system of dispensers will consist of two main parts: software and hardware system. Before designing both systems, the whole system should be designed first based on the user requirements.

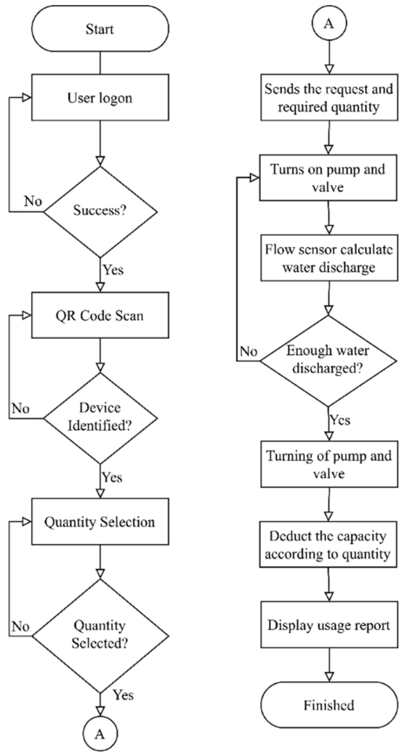

Fig. 1 System design of the dispenser

The system starts with users logging into their phone apps, and continues by scanning the dispenser's QR code. Afterward, the user will be directed to the dispenser menu. Within the menu, there are several options to choose from based on the volume of the water. Then, the user will choose one of the options within the menu as their request which will be sent to the cloud and synced with the database. After syncing, the hardware will proceed with the request and the user can get their request afterward. The system design is shown in Fig 1.

C. Hardware Requirements

After defining the system design, the hardware used for developing the automated dispenser needs to be determined. The hardware requirements for the automated dispenser system are described below:

1) ESP32 board: ESP32 is a series of low-cost, energy-efficient, hybrid Wi-Fi & Bluetooth microcontroller board. ESP32 is used for controlling the whole system of the dispenser. The microcontroller connects to Wi-Fi to get data from the internet to control the system.

2) Power supply: A power supply is an electrical device that supplies electric power to an electrical load. The input voltage power supply is 220 VAC with output voltage 12 VDC. The power supply is used to supply DC to devices that need 12 VDC input, such as water pump and solenoid valve.

3) Relay: A relay is a device that operates as a switch for opening or closing the circuits by receiving electrical current. Relay is used for controlling the solenoid valve and water pump by using current from the power supply as an input voltage and controlled by ESP32 micro-controller.

4) Solenoid Valve: A DC solenoid valve is a valve controlled by electrical current to use its solenoid for opening or closing the valve. The input voltage for the solenoid valve is 12 VDC, the solenoid valve is controlled by ESP32 which is connected to the cloud and used to open or close the water valve from the dispenser based on user requests.

5) Water pump: A DC brushless pump is used for pumping water with 12 VDC input voltage. The water pump has 240 L/H flow rate and is used to pump water from the dispenser to allow the water to pass through the solenoid valve.

6) Flow sensor: A water flow sensor is a tool used to measure water flow rate. The water flow sensor sits in line with the water line, containing a pinwheel sensor to measure how much liquid that has passed through it. The sensor used in this research is YF-S201 with input voltage of 5 VDC with a limit of 30 liters per minute.

D. Software Development

After defining the system design, the next in line in the development stage is to develop the software itself. In this activity, the software will be developed according to the requirements determined before and will be the baseline of the user interface. The software will be in the form of an android application which will become the medium of communication between the user and the system. In software development, there will be two diagrams for the baseline, namely the sequence diagram and entity relationship diagram (ERD).

To lay out the vocal points of the software development, the sequence of user interaction to use the application and its communication between the systems that exist on the implementation of the device itself. It helps design the device itself based on how the user will use it, meaning it will be tailored to suit the user experience. A sequence diagram and its communication scheme are shown in Fig. 2.

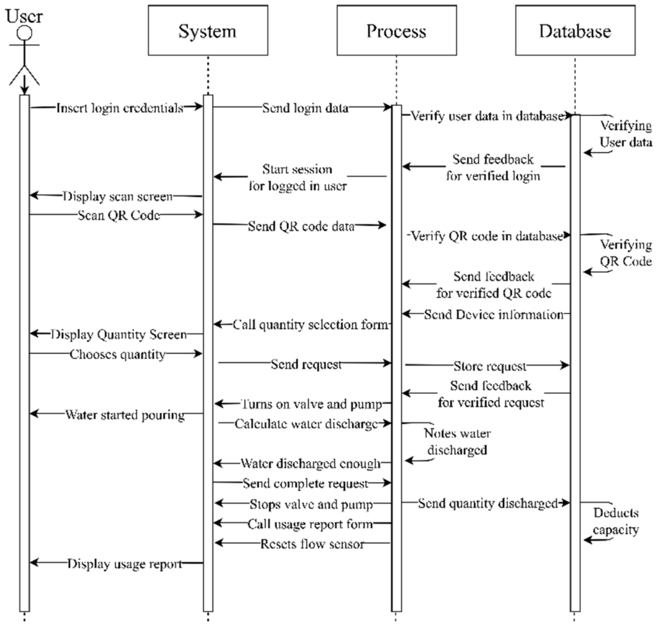

Fig. 2 Sequence diagram and communications scheme

The figure shows the sequence diagram and the communications scheme of the system. The process is as illustrated, and the software development will follow the sequence determined. Starting with the login interface which will require the students to use their student email. Then it will be verified inside the database and if it's valid and verified, the process can continue. The QR scan function is called using the module contained in the android to scan and convert a QR code into a string that will be verified in the database to see if it matches any outlet id.

The screen will then show the result of the scan in which the user can select the product quantity. After that, the request for the product will be sent and the water will start pouring, in which the flow sensor will calculate the amount discharged and send it within the mainboard, which in turn will stop both the pump and the valve. After the request is completed, the water flow sensor will be reset, and a usage report is shown to the user. These whole activities will be present in the developed android application.

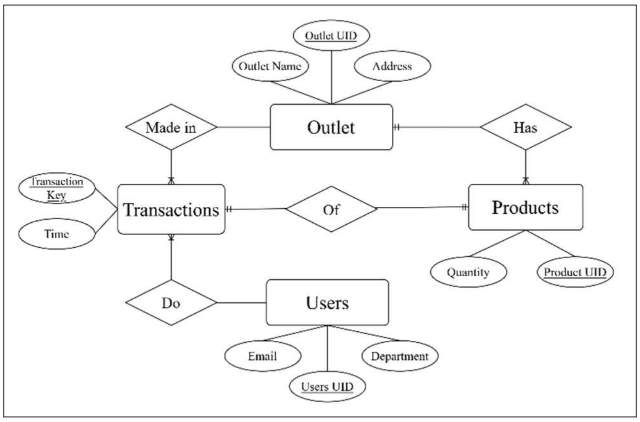

Fig. 3 Entity relationship diagram

The relationships are as illustrated in Fig. 3 where users will be stored with the primary key being its user unique id. The user data also stored the department and email related to each user. The key relationship for users is that users can do many transactions, but one unique transaction could only belong to one user. In order to distinguish between transactions, each transaction occurring will be stored using its own unique transaction key, along with when and where it occurred. Many transactions can be made in an outlet, but one transaction cannot belong to many outlets at once.

Outlets are identified using outlet unique id and stores related data such as its address. Each outlet can provide many different products, that however, can only exist exclusively for one outlet, and is identified using its product unique ID. The product in this case differs in quantities offered. Transaction made contains product bought and is a mandatory one-to-one relationship, meaning one transaction can only consist of one product at any one time.

The transaction data is useful for analysis as it can provide the number of transactions and time interval in between for each outlet, supporting the decision of adding an outlet for that particular location if deemed necessary. As for the real-time database, the key function is to provide a means of communication between the hardware and the software. Therefore, various device status will be stored there as a basis for the control of the hardware itself.

E. Hardware Development

The hardware development will follow the hardware requirements which have been defined in the previous stage.

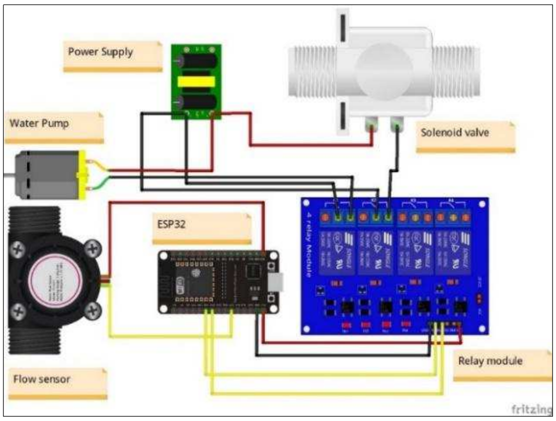

Fig. 4 Hardware system

The hardware system design will be centralized in ESP32 as the controller which is connected to the water flow sensor and relay. The water flows sensor is not required to be connected to the relay because the 3.3 VDC input from ESP32 is enough to power the sensor. Then, the relay is used to control the water pump and solenoid water valve. As for the 12 VDC input voltage which is needed for powering the water pump and solenoid water valve is provided by the power supply. For powering those two components, the power supply is connected to the relay. The hardware system design is shown in Fig. 4.

F. System Architecture

Based on the system design, there will be two subsystems. First is the hardware system which consists of an automated dispenser which ESP32 controls and the second one is a software system which consists of an android application and database which will be used by the users for enabling the communication between user and the dispenser system. Those two sub-systems are integrated real-time in a cloud database. So, whenever a user requests a request via android application, the system will record the request data and send the request to the cloud server, then the cloud will communicate with the dispenser sub-system. The system architecture is shown in Fig. 5.

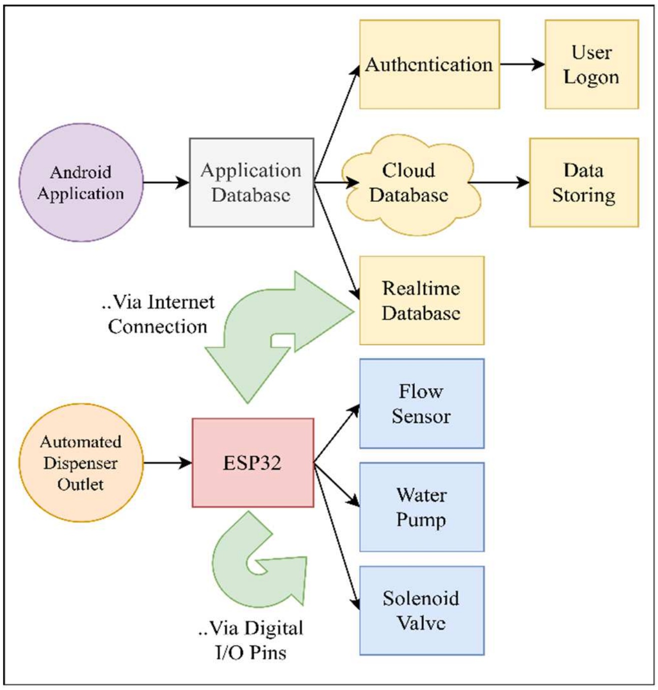

Fig. 5 System architecture

According to the system architecture design, the automated dispenser system consists of two systems, there are hardware and software systems. The software system, which in this case is in the form of an android application, uses both a cloud database and a real-time database for the database of the application. The application can track user logon by using the user data in its database. It can also store the data related to its user and purchases using the cloud database. Then, the android application is becoming the communication tool that connects user and the dispenser system which sends user's requests to the real-time database.

On the other hand, the hardware system, namely the automated dispenser outlet, uses ESP32 microcontroller board as its foundation for communication. ESP32 obtains user requests sent from the application, and it controls both the water pump and solenoid valve to dispense the water. It also receives inputs from the water flow sensor regarding the amount of water discharged to determine whether it's already reached the required amount as chosen by the user or not, and by then, will stop both the water pump and the solenoid water valve.

G. System Integration

To integrate the system, a means of communication needs to be made. Other than the real-time database, the application needs to first recognize the dispenser or hardware that is going to be used. As a solution, a QR Code is made. The QR code is taken from the outlet code string and will be converted to one so it can be scanned from the application and is not oblivious to the user's eyes, meaning the string cannot be inferred directly, only by using the application. The string converted will then be send to database to see if it matches any outlet code. The example of an QR code converted outlet code can be seen on Fig. 6.

Fig. 6 QR Code Example for Outlet Code

III. Result and Discussion

A. System Development Result

After stages of the system development, the prototype system is finally ready to use. The result of the development can then be analyzed to see whether the system already fulfills the determined system requirements or not. Based on table 1, The first three of the system requirements is that users can register their account then log into the application using their registered account and log out from the application. In Fig. 7 shown the interface of login into the application. The login system used in this application is using a user google account. So, users can easily register or login into the application via google account and logout from the menu within the application.

Fig. 7 Login interface

Then, the fourth of the system requirements is the user being able to see their profile based on the data they registered before. Fig. 8 shows the current system user interface, which contains the detailed information of the user. such as user's name, user's location, user's email, and user's phone number and user's registration date. The user can access this function by pressing the profile icon in the bottom left of the interface.

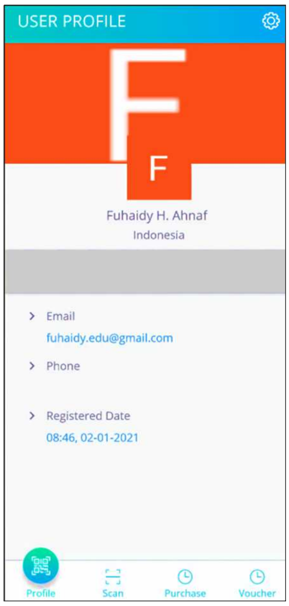

Fig. 8 User profile android interfaces

The next system requirement is QR code scan, The QR code scanner function of the android application that is used for scanning the QR code on the dispenser, by pressing the scan icon on the bottom of the interface, the user can access the QR code scanner. The QR code scanner interface consists of a scanner in the middle of the interface, and this scanner function has built-in flash function that allows the user to use flash when it's hard to scan the QR code. The scanner function doesn't require the user to take a photo of the QR code, the user just needs to point their camera to the QR code on the dispenser. The interface of the QR code scanning function is shown in the Fig. 9.

Fig. 9 QR code scanner interfaces

The next requirement is for the user to be able to retrieve dispenser information and choose the amount of water they desire. So, after the user successfully scans the QR code on the dispenser, the user will be directed to the next function, the scan result and quantity selection interface. The scan result interface is used to retrieve the outlet information, such as outlet's ID and location, along with the available product quantity options that the user can select based on the unique QR code on the dispenser. The scan result and quantity selection interface are shown in Fig. 10.

Fig. 10 Scan result and quantity selection interface

Fig. 11 Post-purchase interface with red fond indicator

Then, the next requirements are system ability to process user requests and the water flow sensor can count the amount of water that comes from the dispenser. So, the next interface is the post-purchase interface after the scan result and quantity selection interface. This interface shows that the system is processing the user request. Then, also within the interface, there is the indicator that indicates whether the request is done or not yet. The indicator is based on the water flow sensor counter. If the counter hasn't reached the amount that the user requested, the indicator color font will become red, and if the counter has reached the requested amount the indicator color font will be green. This can be shown in Fig. 11 and Fig.12.

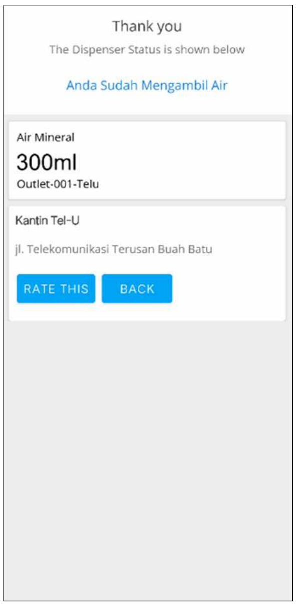

Fig. 12 Post-purchase interface with green font indicator

Next is the last system requirement: the user can track and see their purchase history. Fig. 13 shows the interface of the purchase history that tells the user their purchase history along with the date and place of the purchase.

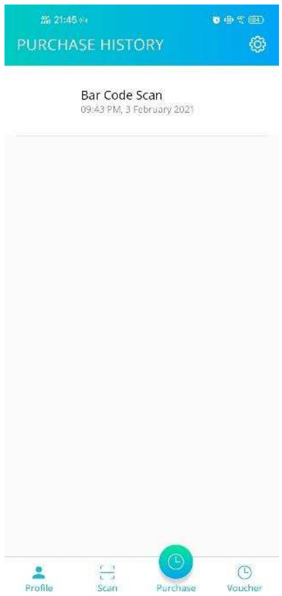

Fig. 13 Purchase History

However, the system can only fulfill one request at a time for each outlet, even if those requests use different water valves. This is due to the overlapping request will not be acknowledged as the program priority is to complete the first request it encounters. In this manner, the user will need to queue manually and, in turn, will require to queue in an orderly manner by themselves. Although, this might be a good thing considering it self-enforces physical distancing between users.

IV. Conclusion

The touchless dispenser system developed performed quite well in accordance with the determined system requirements at the beginning of the development. This means that the system fulfills all the requirements and is successful in its development, with the main objective of providing an alternative public facility to support hydration needs amidst the pandemic in the new normal phase. Because the utilization itself doesn't require the user to touch the dispenser directly, it is expected that people reluctant to fulfill their hydration needs because of the possible covid transmission will be willing to use it as well. As the installation is easy and inexpensive, the implementation can begin quite immediately, especially for universities that host a large number of students, to restrain the spread of COVID-19 while maintaining normal activities in the new normal phase. However, the system can only fulfill one request at once for each outlet, requiring the users to stand in a queue. For suggested improvement, the programming can be made to queue requests in order and fulfill another request that uses a different valve. In the future, it is expected that many systems like this one can be developed to support not only the prevention of spreading transmissible diseases but also as a part of modernizing society and lifestyle towards a better quality of life.

References

[1] WHO, "Indonesia Situation," *WHO Coronavirus (COVID-19) Dashboard*, 2021. [Online]. Available: https://covid19.who.int/region/searo/country/id.

[2] N. Samieefar, R. Rashedi, M. Akhlaghdoust, M. Mashhadi, P. Darzi, and N. Rezaei, "Delta Variant: The New Challenge of COVID-19 Pandemic, an Overview of Epidemiological, Clinical, and Immune Characteristics," *Acta Biomed.*, vol. 93, no. 1, p. e2022179, 2022, doi: 10.23750/abm.v93i1.12210.

[3] Y. Guo *et al.*, "How to Deal with Vaccine Breakthrough Infection With SARS-CoV-2 Variants," *Front. Public Heal.*, vol. 10, no. March, pp. 1–8, 2022, doi: 10.3389/fpubh.2022.842303.

[4] D. Ho, R. Low, L. Tong, V. Gupta, A. Veeraraghavan, and R. Agrawal, "COVID-19 and the Ocular Surface: A Review of Transmission and Manifestations," *Ocular Immunology and Inflammation*. 2020, doi: 10.1080/09273948.2020.1772313.

[5] N. van Doremalen *et al.*, "Aerosol and Surface Stability of SARS-CoV-2 as Compared with SARS-CoV-1," *N. Engl. J. Med.*, pp. 0–3, 2020.

[6] R. W. Kenefick and M. N. Sawka, "Hydration at the Work Site," *J. Am. Coll. Nutr.*, 2007, doi: 10.1080/07315724.2007.10719665.

[7] B. M. Popkin, K. E. D'Anci, and I. H. Rosenberg, "Water, hydration, and health," *Nutrition Reviews*. 2010, doi: 10.1111/j.1753-4887.2010.00304.x.

[8] K. Zemek and C. Johnston, "Hydration status of healthy college students in a large Southwestern university," *Exp. Biol. 2016*, vol. 30, no. S1, p. 207, 2016, doi: 10.1096/fasebj.30.1_supplement.lb207.

[9] M. Nasajpour, S. Pouriyeh, R. M. Parizi, M. Dorodchi, M. Valero, and H. R. Arabnia, "Internet of Things for Current COVID-19 and Future Pandemics: an Exploratory Study," *J. Healthc. Informatics Res.*, vol. 4, no. 4, pp. 325–364, 2020, doi: 10.1007/s41666-020-00080-6.

[10] M. Javaid and I. H. Khan, "Internet of Things (IoT) enabled healthcare helps to take the challenges of COVID-19 Pandemic," *J. Oral Biol. Craniofacial Res.*, vol. 11, no. 2, pp. 209–214, 2021, doi: 10.1016/j.jobcr.2021.01.015.

[11] S. Rahman *et al.*, "Defending against the Novel Coronavirus (COVID-19) outbreak: How can the Internet of Things (IoT) help to save the world?," *Heal. Policy Technol.*, vol. 9, no. 2, pp. 136–138, 2020, doi: 10.1016/j.hlpt.2020.04.005.

[12] S. Balaji, K. Nathani, and R. Santhakumar, "IoT Technology, Applications and Challenges: A Contemporary Survey," *Wirel. Pers. Commun.*, vol. 108, no. 1, pp. 363–388, 2019, doi: 10.1007/s11277-019-06407-w.

[13] K. Karunanithy and B. Velusamy, "Cluster-tree based energy efficient data gathering protocol for industrial automation using WSNs and IoT," *J. Ind. Inf. Integr.*, vol. 19, no. June, p. 100156, 2020, doi: 10.1016/j.jii.2020.100156.

[14] I. Mistry, S. Tanwar, S. Tyagi, and N. Kumar, "Blockchain for 5G-enabled IoT for industrial automation: A systematic review, solutions, and challenges," *Mech. Syst. Signal Process.*, vol. 135, p. 106382, 2020, doi: 10.1016/j.ymssp.2019.106382.

[15] N. Muthukumar, S. Srinivasan, K. Ramkumar, D. Pal, J. Vain, and S. Ramaswamy, "A model-based approach for design and verification of Industrial Internet of Things," *Futur. Gener. Comput. Syst.*, 2019, doi: 10.1016/j.future.2018.12.012.

[16] L. O. Aghenta and M. T. Iqbal, "Low-cost, open source IoT-based SCADA system design using thinger.IO and ESP32 thing," *Electron.*, 2019, doi: 10.3390/electronics8080822.

[17] M. Asif *et al.*, "Industrial automation information analogy for smart grid security," *Comput. Mater. Contin.*, vol. 71, no. 2, pp. 3985–3999, 2022, doi: 10.32604/cmc.2022.023010.

[18] S. M. Riazul Islam, J. Lloret, and Y. Bin Zikria, "Internet of things (Iot)-based wireless health: Enabling technologies and applications," *Electronics (Switzerland)*. 2021, doi: 10.3390/electronics10020148.

[19] V. Subramaniyaswamy *et al.*, "An ontology-driven personalized food recommendation in IoT-based healthcare system," *J. Supercomput.*, 2019, doi: 10.1007/s11227-018-2331-8.

[20] P. Rajan Jeyaraj and E. R. S. Nadar, "Smart-Monitor: Patient Monitoring System for IoT-Based Healthcare System Using Deep Learning," *IETE J. Res.*, 2019, doi: 10.1080/03772063.2019.1649215.

[21] D. Chattaraj, B. Bera, A. K. Das, S. Saha, P. Lorenz, and Y. Park, "Block-CLAP: Blockchain-Assisted Certificateless Key Agreement Protocol for Internet of Vehicles in Smart Transportation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 8092–8107, 2021, doi: 10.1109/TVT.2021.3091163.

[22] B. Jan, H. Farman, M. Khan, M. Talha, and I. U. Din, "Designing a Smart Transportation System: An Internet of Things and Big Data Approach," *IEEE Wirel. Commun.*, vol. 26, no. 4, pp. 73–79, 2019, doi: 10.1109/MWC.2019.1800512.

[23] H. Zhang and X. Lu, "Vehicle communication network in intelligent transportation system based on Internet of Things," *Comput. Commun.*, vol. 160, no. March, pp. 799–806, 2020, doi: 10.1016/j.comcom.2020.03.041.

[24] S. Majumdar, M. M. Subhani, B. Roullier, A. Anjum, and R. Zhu, "Congestion prediction for smart sustainable cities using IoT and machine learning approaches," *Sustain. Cities Soc.*, vol. 64, no. March 2020, p. 102500, 2021, doi: 10.1016/j.scs.2020.102500.

[25] R. Akhter and S. A. Sofi, "Precision agriculture using IoT data analytics and machine learning," *J. King Saud Univ. - Comput. Inf. Sci.*, no. xxxx, 2021, doi: 10.1016/j.jksuci.2021.05.013.

[26] B. Murdyantoro, D. S. E. Atmaja, and H. Rachmat, "Application design of farmbot based on Internet of Things (IoT)," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 4, pp. 1163–1170, 2019, doi: 10.18517/ijaseit.9.4.9483.

[27] R. P. Singh, M. Javaid, A. Haleem, and R. Suman, "Internet of things (IoT) applications to fight against COVID-19 pandemic," *Diabetes Metab. Syndr. Clin. Res. Rev.*, vol. 14, no. 4, pp. 521–524, 2020, doi: 10.1016/j.dsx.2020.04.041.

[28] S. S. Vedaei *et al.*, "COVID-SAFE: An IoT-based system for automated health monitoring and surveillance in post-pandemic life," *IEEE Access*, 2020, doi: 10.1109/ACCESS.2020.3030194.

[29] S. K. Elagan, S. F. Abdelwahab, E. A. Zanaty, M. H. Alkinani, H. Alotaibi, and M. E. A. Zanaty, "Remote diagnostic and detection of coronavirus disease (COVID-19) system based on intelligent healthcare and internet of things," *Results Phys.*, vol. 22, p. 103910, 2021, doi: 10.1016/j.rinp.2021.103910.

[30] M. M. Khan, S. Mehnaz, A. Shaha, M. Nayem, and S. Bourouis, "IoT-Based Smart Health Monitoring System for COVID-19 Patients," *Comput. Math. Methods Med.*, vol. 2021, pp. 1–11, 2021, doi: 10.1155/2021/8591036.

[31] M. Ndiaye, S. S. Oyewobi, A. M. Abu-Mahfouz, G. P. Hancke, A. M. Kurien, and K. Djouani, "IoT in the wake of Covid-19: A survey on contributions, challenges and evolution," *IEEE Access*, vol. 8, pp. 186821–186839, 2020, doi: 10.1109/ACCESS.2020.3030090.

[32] S. Anand, V. Sharma, R. Pourush, and S. Jaiswal, "A comprehensive survey on the biomedical signal processing methods for the detection of COVID-19," *Ann. Med. Surg.*, vol. 76, p. 103519, 2022, doi: 10.1016/j.amsu.2022.103519.

[33] M. Thangamani, M. Ganthimathi, S. R. Sridhar, M. Akila, R. Keerthana, and P. S. Ramesh, “Detecting coronavirus contact using

internet of things,” *Int. J. Pervasive Comput. Commun.*, vol. 16, no. 5, pp. 447–456, 2020, doi: 10.1108/IJPCC-07-2020-0074.

[34] M. K. Pandit and S. A. Banday, "SARS n-CoV2-19 detection from chest x-ray images using deep neural networks," *Int. J. Pervasive Comput. Commun.*, vol. 16, no. 5, pp. 419–427, 2020, doi: 10.1108/IJPCC-06-2020-0060.

[35] B. V. Thummadi and K. Lyytinen, "How much method-in-use matters? A case study of agile and waterfall software projects and their design routine variation," *J. Assoc. Inf. Syst.*, vol. 21, no. 4, pp. 864–900, 2020, doi: 10.17705/1jais.00623.

[36] S. Balaji, "Waterfall vs v-model vs agile : A comparative study on SDLC," *Int. J. Inf. Technol. Bus. Manag.*, vol. 2, no. 1, pp. 26–30, 2012.