# IoT Smart Refrigerator: A Review

**Brandon Tan Qi Loong[1], Tay Jo Tien [2], Manzoor Hussain[3], Sumathi Balakrishnan[4], Mike Goh Chia Chung[5], Sabrina Afrine Sathi[6], Pawandeep Kaur a/p Karpal Singh[7], Imdad Ali Shah[8]**

[1,2,4,5,6,7,8] *School of Computer Science, Taylors' University, Malaysia*

**Article History: Received:** 02.10.2022 **Revised:** 23.12.2022 **Accepted:** 17.01.2023

**Abstract.** This research article focuses on the future of smart refrigerators and on integrating many functions to make life easier for users. The problem that our smart fridge tries to tackle is food waste, which is listed as one of the United Nations' Sustainable Development Goals (SDG). This report will go through the literature research chosen to reflect on our idea and enhance it to guarantee that our smart fridge is viable. This report will also cover how to create a virtual prototype of our proposal using TinkerCAD, including numerous sensors like the TMP-36, Gas Sensor, Force Sensor, and a Piezo to demonstrate how it should be implemented in the fridge. Temperature and Humidity Sensors (DHT-11) will, however, be used in the accurate implementation for improved precision and to measure humidity as well as temperature. Furthermore, to aid comprehension of our concept, our report includes a simple prototype of how our application "iFridge" would work in conjunction with the suggested concept. Moreover, our report will include the limits and recommendations for future study, as well as the success and failure of the concept, to guarantee that future research may exploit the basics of the idea that have been overlooked. Finally, this research attempts to expand on the basics of smart fridges, which are based on prior concepts but have not been generally adopted due to various factors, including security issues when data is carried over the internet.

[1,2,4,5,6,7,8] School of Computer Science, Taylors' University, Malaysia

## 1. INTRODUCTION

To begin, there is a common misconception about what the Internet of Things (IoT) is. The Internet of Things (IoT) is a network of physical devices that includes sensors, software, and other technologies such as servers used to connect and exchange data with the devices and systems via the internet. Why are smart devices, particularly smart refrigerators, so popular nowadays? According to, it makes cooking less of a hassle and less time-consuming for consumers because most modern smart refrigerators allow users to search for fast recipes with the touch of a finger. Furthermore, expiration dates can be [1-3] easily set for each item stored in the refrigerator, making things easier for users by listing the expiry dates of each item to prevent wastage. Some smart refrigerators also allow users to create their unique to-do lists and grocery lists that other family members can check on their applications to get the items stated in the grocery lists when they are out, saving time and effort. Finally, one of the smart refrigerator's supporting tasks is the ability to browse the internet to watch their favourite shows or listen to their favourite music while the user is cooking.

However, even with the benefits described above, we are left with a question; what distinguishes our proposed solution as a purple cow? Some think that smart fridges might not be cost-effective because the prices being too high. Some users may believe that this is a waste of money to start with as it just adds features like temperature sensing, a camera sensor, and more to view items stored inside the fridge [4-6]. However, our suggested solution provides a long- term effect for the user as it cooperates with our application “iFridge” to carry functions like adding items with a shortage into the grocery lists to ease the user’s daily life where the user does not need to open and close the fridge to check what items need to be included into the grocery list [7-9]. Following the last assignment, we decided to incorporate a touch screen into it, allowing the user to access the internet and watch their favourite shows or chefs providing instructions on how to make a meal, among other things. Although we have discovered that this may be a distraction, as some users may become too engrossed in the touch screen and forget about the dishes still being cooked. Moreover, one of our plans that distinguishes our product from others is integrating one or more hypermarkets like Lotus, Giant, or Aeon into our application. The integration is done to increase the user's convenience and prevent a shortage of food in the refrigerator. It is done by alerting the user and giving the option for the user to order raw materials online with the hypermarket partners, add the products to their shopping list, or purchase the materials by going to the grocery stores themselves.

To summarise, the development of new smart devices continually intends to simplify our daily life activities, but it may also put our devices at significant risk due to how interconnected devices are. As shown in Figure 1, the attacker can utilise the sensor alert to send the user a phishing link where sensitive information such as banking logins, social media logins, and more can be accessed directly from the user's mobile devices [10]. It is not implied, however, that all smart devices are dangerous; rather, some devices are more vulnerable than others. As a result, implementing a secure and appropriate security system will create a secure environment for the users to operate the smart fridge without the risk of personal information being taken during the invasion of cyberattacks.

## 2. LITERATURE REVIEW

The research and implementation of IoT technology in fridges has been in the works for a very long time, ever since 1999. Besides connecting the fridge to the internet, it can also provide various features that are convenient to the user, alongside solving issues regarding food wastage. Analysing past system architectures and proposals, this section reviews the general concept of implementing IoT devices and sensors used in smart fridges. Looking at past architectures, the model generally consists of the perception layer, [11,12] transport, and application layers. These layers are important for implementing IoT features into the smart fridge.

Under the perception layer, it was common to find a motherboard along with a few devices or sensors. The motherboard is the most important part of the system as it allows communication between the other hardware components. The first architecture reference, Fig. 1 (a), connects the mainboard with an external board called "NxBoard". This board is used to access the user's WiFi or Bluetooth for communication between the smart fridge and the user's phone when using the mobile application. The second architecture reference, Fig. 1 (b), uses the "WeMos D1 R2 Wi-Fi" board, which includes a Wi-Fi module on the board itself. This board is a microcontroller that serves the same purpose as a motherboard. Alongside this board, two sensors are connected to obtain certain information. The MQ3 sensor is a gas sensor used to detect the gases released by rotting products [13-15]. The DHT11 sensor is a temperature and humidity sensor used to measure the temperature and humidity inside the fridge. The third reference architecture, Fig. 1 (c), consists of a microcontroller named "ATMega 328". This board is connected to a load cell sensor used to measure weights detected on it. As well as a webcam, which is used to view and perform image processing of the contents inside the fridge.

Under the transport layer, the main method of communication throughout the three architectures is Wi-Fi. This allows wireless communication between the fridge, the server and the user's device. The three architecture references implement a mobile application to perform actions on the fridge through the user's devices; this means a connection is required to the server. The first architecture model, Fig. 1 (a), uses an IoT platform called "Ubidots", a cloud server that keeps real-time and other user data. Bluetooth is also an option for communication between the user and the fridge. The second architecture, Fig. 1 (b), uses another IoT platform called "ThingSpeak". Besides allowing communication between the system and user, it consists of many features, such as creating channels with a public or private view. Additionally, the architecture uses an application called "Pushbullet" to send notifications and alerts.

Under the application layer, another way for the user to interact with the fridge is through a phone application. In the first architecture reference, Fig.1 (a), the application receives and sends data through a cloud server, Ubidots. The server provides data, resources or services to users and devices by sending, receiving or storing the data. Through the cloud servers, users can view the real-time temperature recorded from the sensors, as well as other features that the application may provide. Such as in the second architecture, Fig. 1 (b), the user can receive notifications from the phone application. This feature can provide the user reminders regarding expiring foods, rotting foods in the fridge, etc. Furthermore, in the third architecture reference, Fig. 1 (c), as the load cell provides the weight of the food products, the user can view the weight of a certain product and identify whether it is running low [16-18]. Image processing is also implemented to the webcam, which allows the application to recognise vegetables and recommend recipes.

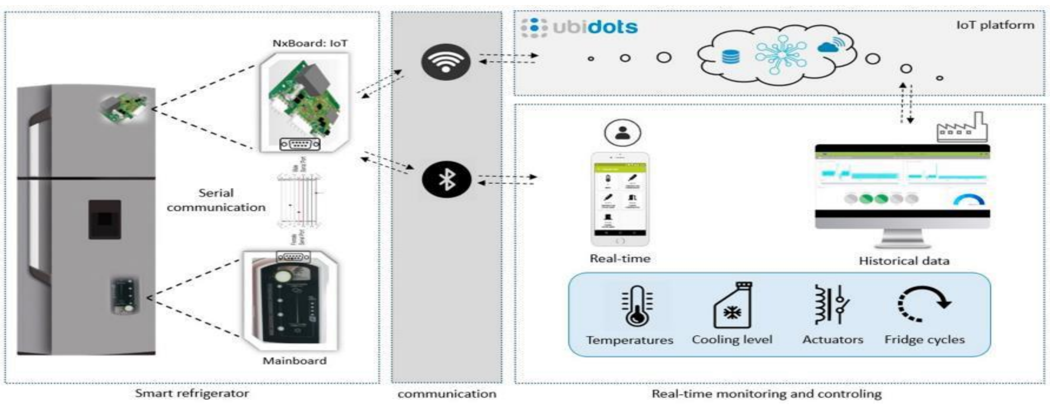



**Fig (a)**

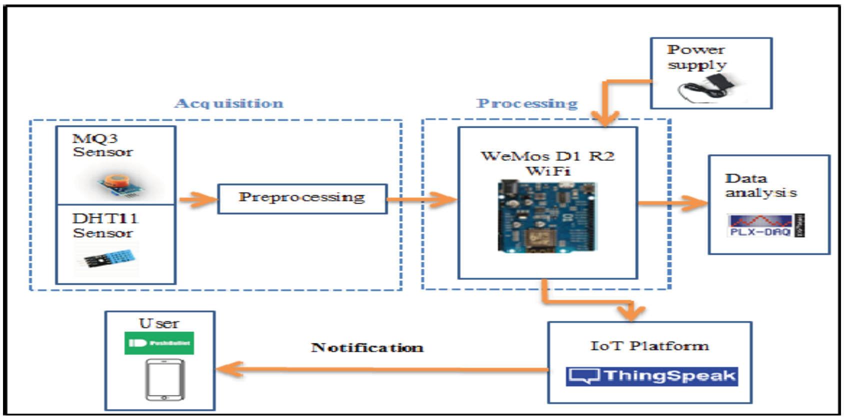



Fig (b)

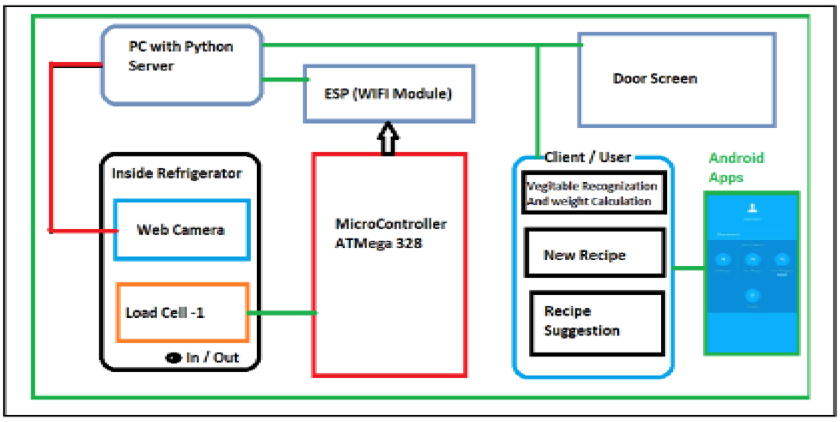



Fig (c)

**FIGURE 1. Architecture references based on smart fridges. (a) First architecture reference.** [7.] **(b) Second architecture reference.** [8.]
**(c) Third architecture reference.** [9.]

## 3. METHODOLOGY

The system architecture diagram in Fig. 2 is the overview of the proposed system, including vital layers of IoT – Perception Layer, Transport Layer, Application Layer and Data Processing Layer.

### 3.1 Perception Layer

The perception layer, also known as the physical layer, is the initial layer in IoT architecture. This layer incorporates many sorts of sensors and actuators for detecting and collecting information in real-time. The sensors and hardware components used are as followed: -

MQ-2 Gas Sensor- detects gases such as methane and alerts the user which will be placed on the ceiling of the fridge; methane is lighter than air, making it easier to float up and be sensed by the gas sensor.

DHT11 Temperature and Humidity sensor - placed on [19,20] the top edge of the fridge (near the door) to collect data inside the fridge, ensuring that the temperature is set to the user's preference so that the contents of the fridge are not harmed, and the user can also set the temperature dynamically by pressing the button.

Fig 2 Architecture of Smart Refrigerator system

Weight sensor - detects the amount of beverage left in the bottle container.

OV7670 camera sensor- record the refrigerator's contents and show them on the touch screen tablet.

- SSD1963 Touch Screen- displays the temperature, gas, humidity, other sensor data and applications that can be used.
- HC-05 Bluetooth Module- connect the user's mobile device to the smart refrigerator, allowing information to be seen remotely.
- ESP8266 WiFi Module- provide internet connectivity for touchscreen functions such as streaming user-selected entertainment, looking for internet recipes, and more.

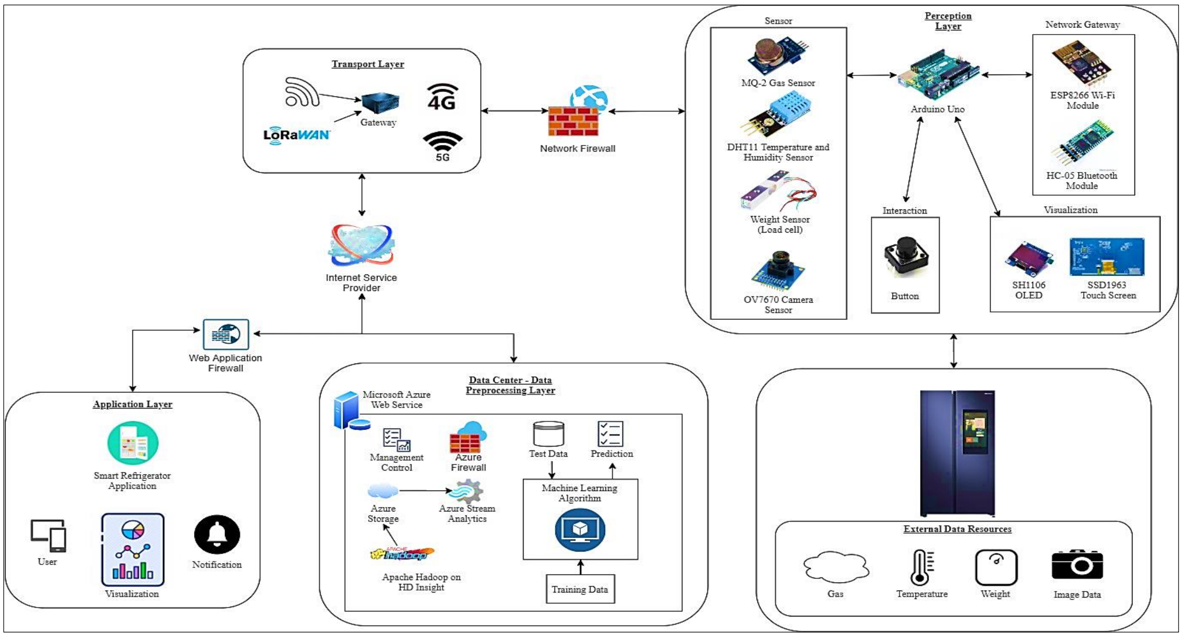



**FIGURE 2** Architecture of Smart Refrigerator system

### 3.2 Transport Layer

Wi-Fi and LoRaWan are used in the Transport Layer. The integration of the two types of connectivity produces considerably more outstanding achievements in both domains as the speed of transfer, bandwidth, and connectivity all see a positive boost due to the complementing skills of each mode of transfer. A network firewall is used between the Transport and Perception Layer to ensure no malicious traffic transactions between both layers. The transport layer is connected to the Internet Service Provider (ISP), increasing system vulnerabilities. Thus, more security should be focused on securing the environment and lowering the risks. Next, A web application firewall resides above the application layer, [21-23] monitoring, analysing, and filtering protocols like HyperText Transfer Protocol (HTTP) and Secure HyperText Transfer Protocol (HTTPS) from the ISP to the application. This web application firewall might potentially serve as a reverse proxy, shielding the application from malicious queries and preventing attacks such as SQL injection, Cross-Site Scripting (XSS), and others. Moreover, the use of the WiFi module, it meant that the fridge had to have some kind of unique way to recognise it. IPv6 protocols will be used to allow communication between the fridge and the application. Using IPv6 the system can communicate with various devices normally as mobile devices nowadays are implemented with both IPv4 and IPv6. Another reason IPv6 was chosen is that IPv4 is about to be phased out and replaced by IPv6; therefore, the use of IPv6 for our solution is more focused on the long term, like the next ten years.

3.3 Application Layer

The user can visualise data acquired by the sensor in the application layer. This application may be loaded on both Android and iOS devices to make it easier to use. It can also provide push warnings in an emergency, such as high humidity, methane, or temperature, to guarantee that the loss of contents in the smart refrigerator is kept to a minimum.

3.4 Data Preprocessing Layer

Microsoft Azure was chosen as the cloud service provider because of its dependability and capabilities, including databases or data warehouses, data analytics, and service management. Apache Hadoop with HDInsight will deal with irrelevant data before it is stored in the cloud because data will be collected [24-26], stored, and analysed in big batches and in real-time. Furthermore, a cloud firewall will be subscribed to Microsoft in the data centre to provide data security. The key reason is that it enables scalability, with the option to add bandwidth and new site protection if the business requires it or expands. Fig 3 Implementation of Smart Refrigerator system.

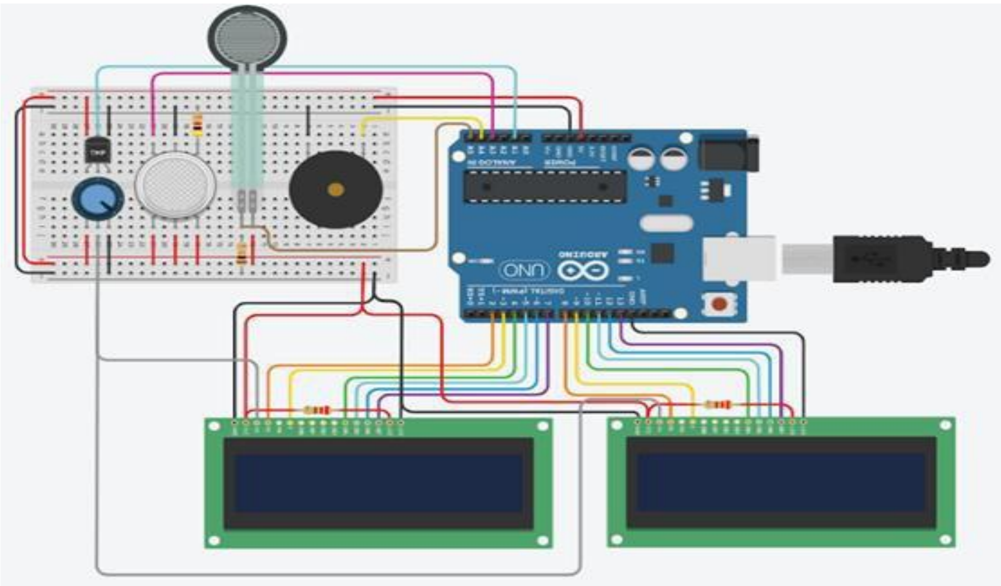


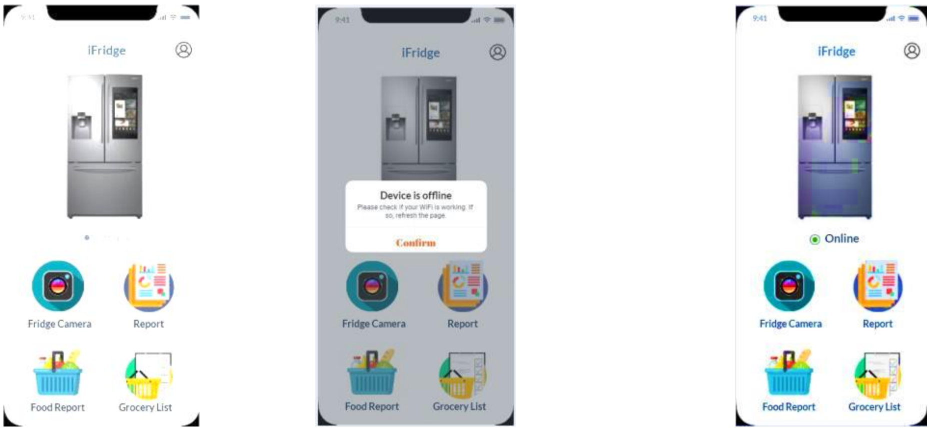
**FIGURE 3 Implementation of Smart Refrigerator system**

The primary function of our smart refrigerator is to receive inputs in the form of data collected from the humidity, temperature, gas, and weight sensors. Due to TinkerCAD's functionality constraints, an estimate of only 50% of the project could be implemented, leaving out the temperature and humidity sensor (DHT11), the image sensor (OV7670), the Bluetooth module (HC-05), the Wi-Fi module (ESP8266), and the two visualisation methods (SH1106 OLED and SSD1963). The temperature and humidity sensor was replaced with the TM35 temperature sensor, while an LCD screen was used as a visualisation method [27-29]. The remaining components were not implemented due to insufficient support for programming libraries and components, unavailability of Bluetooth and Wi-Fi modules, and lack of pins on the Arduino board. The components used in our TinkerCAD project is listed below;

- Arduino Uno R3
- Breadboard (Small)
- Piezzo (Buzzer)
- Force Sensor (Load cell)
- Gas Sensor (MQ-2)
- Temperature Sensor (TM36)
- Potentiometer
- LCD
- Resistor

3.5 Mobile Application

Along with the smart fridge, a mobile application can be installed from the user's phone named "iFridge". This mobile application can connect to the smart fridge for the user to operate the fridge wirelessly as well as provide the user with additional features. The mobile application prototype was designed using the Marvel app website.




(a) (b) (c)

**FIGURE 4** Home page statuses of iFridge application. (a) Home page while offline. (b) Home page with offline prompt. (c) Home page while online.

Figure 4 (a) shows the homepage of the iFridge application while offline. This means that the phone is not connected to the smart fridge, an alert box will appear telling the user that the device is offline, seen in Figure 4 (b). After the user successfully connects to the fridge, the status will appear online with a green indication, as seen in Figure 4 (c). Users can select a few features available on the dashboard, such as Fridge Camera, Report, Food Report and Grocery List. The user can also select to view their profile page on the top right of the application.

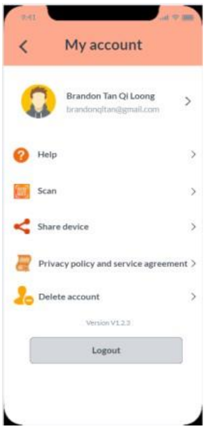



**FIGURE 5.** Account page.

Figure 5 shows the details regarding the user profile and other options. In this section, users can:

- View further information about their profile.
- Report complaints or request maintenance service in the help section.
- Scan a QR code for adding new users or devices to the fridge.
- Share an invitation link or QR code to add other devices to connect to the fridge.
- View the privacy policy and service agreement.
- Delete the account when the smart fridge is no longer in possession of the user.
- Log out of the account.

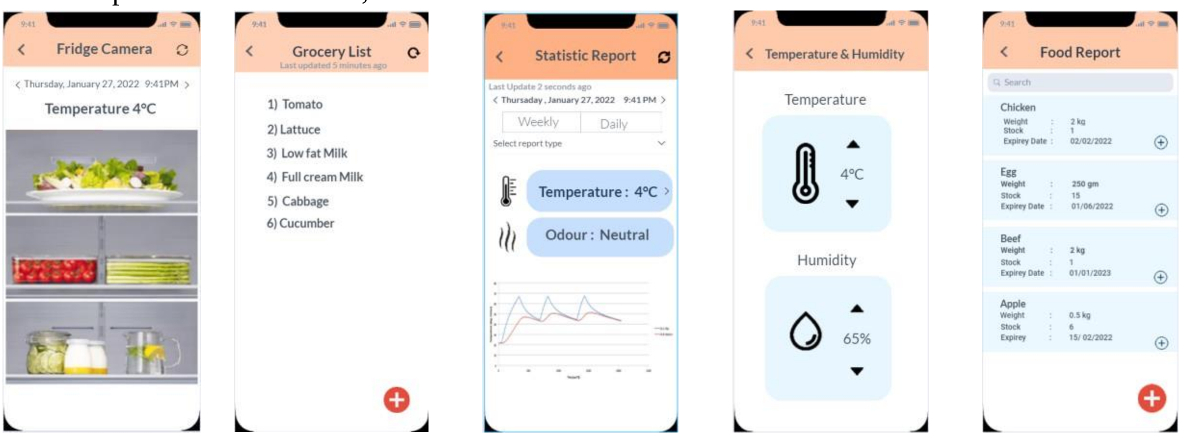



Fig (a) Fig (b) Fig(c) Fig(d) Fig (e)

**FIGURE 6.** Pages of each feature in the iFridge application. (a) Fridge Camera page. (b) Grocery list page. (c) Statistic Report page. (d) Temperature and Humidity page. (e) Food Report page.

Moving onto the features: In Fig. 6 (a), the fridge camera allows the user to view the contents inside the fridge, alongside the current temperature and previously recorded captures using the arrows between the date and time.

In Fig. 6 (b), the grocery list page shows all items that need to be restocked. The application can suggest which items should be restocked and thus be added to the list when the user confirms it. Additionally, the user can make changes using the plus button to add items.

In Fig. 6 (c), the report shows statistics and analysis regarding the data gathered by the sensors, such as the temperature and odour. Previous reports can also be viewed by using the arrows between the timelines. The temperature can also be changed by pressing the temperature, which will direct the user to the temperature page shown in Fig. 6 (d).

In Fig. 6 (d), the user can change the temperature and humidity by using the arrows to change the values.

In Fig. 6 (e), the food report page allows the user to view the quantity and condition of each item available. This function requires the user to manually input each new product to the list. The food description consists of the name, weight, stock and expiry date. Users can also search for the items that are registered in the list.

## 4. RESULTS AND FINDINGS

Traditional refrigerators store food items and lengthen food freshness levels; however, it does not guarantee food items will not get rotten. By implementing temperature, gas, humidity, weight, and camera sensors, users would be notified of food conditions and the refrigerator's condition. The mobile application system generates reports that visualise data collected from the sensors using graphs and charts, e.g., weekly distribution of refrigerator temperature, quantities of foreign gases, and time history of food items being stored in the refrigerator [30-32]. The implementation of unsupervised machine learning allows the system to learn users' buying and storing habits to provide food item recommendations to the users then.

### 4.1 Constraints and Assumptions

Users may find the "smart" aspect of the smart refrigerator unnecessary and expensive to both users and developers. Users may find that a refrigerator with autonomous features such as notifications, monitoring, and reporting their fridge and food condition may not justify its high price. Initial costs for developing and deploying security software for the system would be high for the developers and manufacturers.

A fast internet connection is required for instant data transfer between the refrigerator system and users via the mobile application. The user's mobile phone is assumed to be connected to either Wi-Fi, 4G, or 5G networks at all times to receive real-time notifications to ensure no important notifications are missed [33-35]. Wi-Fi routers may be placed closer to the refrigerator, or users may set Wi-Fi extenders to boost Wi-Fi signals. Food items are also assumed to be placed in the refrigerator one at a time to detect and weigh food items correctly. Table 1 mobile application functional requirement.

**TABLE 1.** *Mobile Application Functional Requirement*

| ID | Functional Requirement | Description |
|---|---|---|
| FR1 | Account Management. | Account management covers the ability of the user to register, login, logout, view and edit their profile. |
| FR2 | Create daily, weekly, or monthly reports, allowing users to view history of existing records, and customise dashboards containing the reports. | Creating reports over a specific timeline allows for users to track the management of their smart refrigerator properly. |
| FR3 | Settings which include automatic updates, instruction manuals, searching for fridge settings, altering parameter limits, and remote controlling settings. | Settings are a standard functional requirement as the user should be able to control their personal devices including being able to alter the limits for the temperature, gas and humidity, or being able to opt-out of getting automatic updates, etc. |
| FR4 | Display dashboard, which has other functions. | The dashboard is the method for the user to view and interact with the smart refrigerator via the cloud. |
| FR5 | Notifications for immediate gas spoilage, food and fridge conditions, etc. | Notifications received on the mobile application alerts the user if there are any issues arising in the refrigerator, such as the spoiling of food, the fridge not being closed properly, components of the refrigerator requiring maintenance, etc. |

| | | |
|---|---|---|
| FR6 | View, add or remove multiple users in the share list. | As the method of connecting to the smart refrigerator is via Bluetooth, then being linked permanently to that account, it is important to have the ability to manage other users and invite users to manage or remove previously added users. |
| FR7 | Customer Service for automated maintenance report and help centre and customer service. | The user can discern any anomalies in the data and choose to report such issues to the maintenance centre. |
| FR8 | Connection to Smart Fridge and Syncing the data with the cloud. | The smart fridge connection includes being able to pair with the refrigerator via Bluetooth and then accessing the refrigerator from the application via the cloud. |
| FR9 | Checking real-time temperature, humidity, gas, weight, and camera. | From the application, the user should be able to view the real-time data being produced which includes the temperature, humidity, gas, weight, and images of contents in the refrigerator. |
| FR10 | Can see touch screen drawings for any notes. | Allows the users to interact with the touch screen to write certain notes. Those written notes should be converted to image format and be seen in the application after being stored in the cloud. |
| FR11 | Can view algorithm-generated shopping list. | Using machine learning, the Arduino will undergo unsupervised learning where it recognises the users' habits based on the patterns that they do. |
| FR12 | Can add new items into the fridge and then fill in the corresponding information. | Users can add new items without having to interact with their fridge. With the application, they can do it anywhere to save time and effort. |

**TABLE 2.** *Arduino Functional Requirements*

| ID | Functional Requirement | Description |
|---|---|---|
| FR1 | Piezo produces a buzzing sound if the gas concentration, humidity, or temperature has passed the limit defined. | In the Arduino, the Piezo serves the function of producing a buzzing sound. It will buzz only when a certain limit has been exceeded based on the users' settings or the default value if not changed. |
| FR2 | Touch screen must be sensitive to touch. | The user interacts with the fridge via the touch screen if not on their phone. Using the touch screen allows for the user to write their notes and access other functions. |
| FR13 | Data on the touch screen must change when parameters that are inside the fridge change. | For the user to identify and correctly use the smart refrigerator as intended, the internal parameters of the fridge must be displayed. |
| FR4 | The Wi-Fi module must be functional to sync and write data to the cloud. | As the Wi-Fi module is a component, it must be usable for the data to be read and written to the cloud. When the sensor reads data, the data is filtered and then uploaded to the cloud. Without an internet connection, the data |

| | | |
|---|---|---|
| | | cannot be reported to the cloud or downloaded without an internet connection. |
| FR5 | The Bluetooth module must be functional to pair with the mobile application before syncing to the cloud. | The Bluetooth module connects the phone to the smart fridge. Bluetooth is a Peer-to-peer system so without Bluetooth, it would be challenging to pair the smart refrigerator with the device. |
| FR6 | All the sensors must be sensitive to change. | To serve all the functions in detecting the environment inside the fridge, it is crucial to have the sensors notice the changes and react accordingly. |

4.2 Strengths

Throughout any project, it is without a doubt that it will be presented with its own strengths, making it the selling point of any individual product. In the case of the smart refrigerator, some strengths include remote monitoring, long- term cost-effectiveness, scalability, improved family health, and reduced food wastages. Remote monitoring is enabled by allowing users to access their functionalities directly on their mobile devices. Some functions include modifying temperature and humidity, creating grocery lists that are generated using unsupervised machine learning, automatic report generation, and so on. Long-term cost-effectiveness includes the grocery list, as mentioned earlier, and the ability to remotely check the refrigerator's contents without needing to be there physically [36-38]. This, in the long run, allows the user to save fuel, time and effort indirectly. Moving onto one of the most crucial aspects, scalability. It is defined as the ability to increase the size and performance to match the workload, or at least meet such demands [39]. In the case of Smart technology, it can be interpreted as the ability to match the demand of internet traffic through upgrades in hardware, or improvements in software. As the Smart Fridge is built upon the cloud infrastructure, it has the inherent advantage of being able to delegate more hardware with little effort. This results in savings of effort, time for installation, cost of hardware and labour expenses.

4.3 Weaknesses

Much like strengths, any project will be met with its downsides through any additions. The first weakness, which may be a potential issue, is that users may find that adding items individually is time-consuming for the user should there be many types of products being added. Then, moving on to the security weaknesses, although there are security precautions in place, there are still natural threats of data breaches or attacks [40-42]. These attacks may lead to compromised personal information, malware infections, notification spamming, and malicious pranks being done. Besides personal information being leaked, it is also possible for the communication to be overloaded by sending heavy traffic to the Arduino via Wi-Fi. Though, this largely depends on the security capabilities implemented in the users' network system.

## 5. DISCUSSION

The system can prevent food waste, which, if not addressed, could result in higher food costs, world hunger, and a loss of biodiversity. The smart refrigerator system aids in these issues by alerting users of food items stored and when it begins to rot. Important notifications, such as the condition of food items, the interior temperature of the refrigerator, and the presence of gases due to food decomposition, are sent to users via the system's application [43-45]. As a result, users will be more conscious of the food in their refrigerators, leading to a decline in food waste by removing the need for users to purchase more food products, leading to over-purchasing.

Limitations to ensure the system will still be reliable in three years include power consumption management, functionalities of future versions, traffic and bandwidth, and system components. A refrigerator with "smart" capabilities running continuously wastes energy. Without proper power management, this could have a detrimental impact on the environment. Implementing a 'low power mode' on the Arduino board and sensors would reduce power usage. Future functions of the system could be auto-detection of the liquid weights, notifications when ice is ready, and alerts when hot food items are placed inside the cold refrigerator. With more functionalities, the refrigerator would produce more traffic and higher volumes of data. The bandwidth would have to be increased to ensure the system does not slow down while accommodating seamless data transmission and reception [46,47]. New components such as sensors with improved features and updated software could also provide relevancy and better functionality to the system.

5.1 Manage Resources

To manage resources, many aspects need to be identified. Order includes wireless capability and interoperability, functionality, secure storage and transfers, immediate boot capacity, power management, and other challenges. In terms of wireless capability and interoperability, the system requires an internet connection as, without it, data and messages/protocols cannot be transmitted. To add information, the data is transmitted over the combination of Wi-Fi and LoRaWAN, a highly effective variety that increases network performance and range, whilst reducing costs of power. Then, as bandwidth falls under wireless interconnectivity, it is a worry that the number of users would be too great to be handled, as well as whether the data is being filtered before being uploaded [48-50]. To manage the former point, according to [22.], the bandwidth can easily be upgraded as their hardware is often underutilised therefore delegating what is only required allows extreme flexibility and scalability. The data is appropriately filtered to manage the latter point, ensuring no immense volume of continuous data from any user.

For functionality, the Arduino, codes, and server must be fully operational and fault free. To ensure that there are no issues, a few precautions can be taken, such as thoroughly testing and configuring all components before distribution, regularly updating the system software, and patching bugs or issues. The main benefit to hosting the server through the cloud infrastructure is that it remains fully operational even if one server crashes as a secondary backup server can take its place. Moving onto secure storage and transfers, the data server will be hosted on Microsoft Azure's cloud server. According to, Microsoft does their best for encryption and security [51]. To ensure their security capabilities, Microsoft Azure implements the IEEE 802.1AE MAC Security Standards, Internet Protocol Security (IPsec), and Transport Layer Security (TLS). Secure storage is fully implemented to prevent security threats. As Microsoft Azure secures the data, there is a low probability of data being breached or tampered with as it is stored on their cloud.

As the boot capacity and power management is a device concern, this can include the concerns of the power and speed management of both the smart refrigerator and application speed. Typically, when identifying the speed of both devices, the smart refrigerator runs on bare hardware capabilities with minimal operating system functions, therefore maximizing the speed available. Furthermore, data does not clog the Arduino as data is only stored on the Arduino if it has lost connection with the server. On the other hand, the application's speed is often dependent on the user's phone [52-54]. When looking at power management, there are two main aspects to consider: redundant processing and how it can be improved. Redundant processing involves the optimization of the device's codes. The smart fridge that has been created is made to filter data; thus, energy wasted is minimised since data is not repeatedly sent to the server. The power management improvement can be the implementation of low power modes for inactive periods, an example being nighttime, as well as increasing the delay between each computation iteration, increasing latency, but likewise using significantly less power over a long period.

Finally, under other challenges, there are physical hazards and unexpected data loss via internet connection drops. When the compressor is overworking, it may generate too much heat and thus lead to damage to physical components such as the Arduino wires. If such parts are damaged, data inaccuracy or no connection to the components themselves may result in data inaccuracy. This can be managed by the logging system as if there is a high standard of deviation present, it can be taken as an indication of a connection or sensory issue and thus, maintenance staff can be sent to handle the problem [54]. For connectivity loss, if the connection is dropped suddenly, data may not be adequately sent, and the transmission could be considered a failure. Although, if it is deemed to have failed, the consequent generation of data will be backed up onto the Arduino until the connection is restored.

## 6. CONCLUSION

Overall, this paper's entirety falls back to the Smart Refrigerator. As to what the system solves and how it solves it, there are many benefits, which include: remote monitoring of the fridge's internal environment, therefore, preventing food wastage, reminding the users of upcoming expiry dates, avoiding over-purchasing of food already owned, and finally, long term time, effort and resources saved. These were all challenges faced by traditional resources that had yet to be adequately solved before Smart Refrigerators came into the picture. Firstly, in terms of remotely monitoring the fridge's internal environment, this was a challenging task to manage for those who may have ample supplies of vegetables or fruits as they may not keep the product lasting as long as possible. However, with the ability to manage internal parameters, extending the lifespan of such items is much easier to manage. Then, expiry date reminders allow users to ensure that there is no food left unchecked in the corner of the refrigerator, growing mould or bacteria without knowledge, therefore spoiling other food in its vicinity. Finally, when a tracker of what exactly is in the refrigerator exists, there is no problem of over-purchasing food a second or a third time because it is forgotten that there is the same item tucked away. These improvements allow for resources, time, and effort to be saved since users need not go out of their way to scour the refrigerator when everything is easily accessible within a few taps. As for what may be improved soon, this includes improvements in power consumption management by choosing less power-demanding components as the system's heart, a greater range of functionalities, and lastly, data transmission improvements, which point to traffic and bandwidth.

## REFERENCES

[1] Howard's, 6 smart fridge features that make life easier (2020).

[2] P. Silvia, Spencer's TV & Appliances (2020).

[3] E. Smith, DIY Appliance Repairs, Home Repair Tips and Tricks (2021).

[4] V. Burns, What is a smart fridge and do I need one?, Real Homes (2019).

[5] PVS Builders & Developer (2021).

[6] M. Burkhalter, Perle (2020).

[7] S. Aheleroff, X. Xu, Y. Lu, M. Aristizabal, J. Pablo Velásquez, B. Joa, and Y. Valencia, Advanced Engineering Informatics **43**, 101043 (2020).

[8] H. Nasir, W.B. Aziz, F. Ali, K. Kadir, and S. Khan, 2018 2nd International Conference on Smart Sensors and Application (ICSSA) (2018).

[9] P.C. Sane, P.H.K. Barapatre, and P.A. Sanghavi, SMART REFRIGERATOR AND VEGETABLE IDENTIFICATION SYSTEM USING IMAGE PROCESSING AND IOT **6**, (2021).

[10] J. Cook, Business Insider (2016).

[11] What Is a Motherboard? (2021).

[12] Paessler, What is a server?.

[13] Spotlight on Safety Thank you for reaching out. MSA Corporate Blog (2019).

[14] M. Wilson, Restaurantnorman.com (2021).

[15] K. Stremel, EETimes (2019).

[16] CrowdStrike.com (2022).

[17] M. Patil, S. Subramaniam, and R. Hegadi, Proceedings of the 2016 International Conference on Communications, Information Management and Network Security (2016).

[18] PhoenixNAP Blog (2021).

[19] A. Hayes, Investopedia (2021).

[20] Javatpoint, Advantages of Cloud Computing (2022).

[21] i-Scoop, LoRa and LoRaWAN: the technologies, ecosystems, use cases and market.

[22] What Is Cloud Computing? A Beginner's Guide | Microsoft Azure.

[23] Data Privacy in the Trusted Cloud | Microsoft Azure.

[24] Ali, S., Hafeez, Y., Jhanjhi, N. Z., Humayun, M., Imran, M., Nayyar, A., ... & Ra, I. H. (2020). Towards pattern-based change verification framework for cloud-enabled healthcare component-based. *Ieee Access, 8,* 148007-148020.

[25] Khan, N. A., Brohi, S. N., & Jhanjhi, N. Z. (2020). UAV's applications, architecture,

security issues and attack scenarios: A survey. In *Intelligent Computing and Innovation on Data Science: Proceedings of ICTIDS 2019* (pp. 753-760). Springer Singapore.

[27] Zaman, N., Low, T. J., & Alghamdi, T. (2014, February). Energy efficient routing protocol for wireless sensor network. In *16th international conference on advanced communication technology* (pp. 808-814). IEEE

[28] Gaur, L., Afaq, A., Solanki, A., Singh, G., Sharma, S., Jhanjhi, N. Z., ... & Le, D. N. (2021). Capitalizing on big data and revolutionary 5G technology: Extracting and visualizing ratings and reviews of global chain hotels. *Computers and Electrical Engineering*, *95*, 107374.

[29] Diwaker, C., Tomar, P., Solanki, A., Nayyar, A., Jhanjhi, N. Z., Abdullah, A., & Supramaniam, M. (2019). A new model for predicting component-based software reliability using soft computing. *IEEE Access*, *7*, 147191-147203.

[30] Sennan, S., Somula, R., Luhach, A. K., Deverajan, G. G., Alnumay, W., Jhanjhi, N. Z., ... & Sharma, P. (2021). Energy efficient optimal parent selection based routing protocol for Internet of Things using firefly optimization algorithm. *Transactions on Emerging Telecommunications Technologies*, *32*(8), e4171.

[31] Hussain, S. J., Ahmed, U., Liaquat, H., Mir, S., Jhanjhi, N. Z., & Humayun, M. (2019, April). IMIAD: intelligent malware identification for android platform. In *2019 International Conference on Computer and Information Sciences (ICCIS)* (pp. 1-6). IEEE.

[32] Shafiq, M., Ashraf, H., Ullah, A., Masud, M., Azeem, M., Jhanjhi, N., & Humayun, M. (2021). Robust cluster-based routing protocol for IoT-assisted smart devices in WSN. *Computers, Materials & Continua*, *67*(3), 3505-3521. ]23\ https://www.researchgate.net/publication/233823923_Traffic_Monitoring_Using_M2M_CommunicationM2M_tongsin-eul_sayonghan_teulaepig_moniteoling/figures?lo=1

[33] Adeyemo, V. E., Abdullah, A., JhanJhi, N. Z., Supramaniam, M., & Balogun, A. O. (2019). Ensemble and deep-learning methods for two-class and multi-attack anomaly intrusion detection: An empirical study. *International Journal of Advanced Computer Science and Applications*, *10*(9) doi:https://doi.org/10.14569/IJACSA.2019.0100969

[34] Gaur, L., Singh, G., Solanki, A., Jhanjhi, N. Z., Bhatia, U., Sharma, S., ... & Kim, W. (2021). Disposition of youth in predicting sustainable development goals using the neuro-fuzzy and random forest algorithms. *Human-Centric Computing and Information Sciences*, *11*, NA.

[35] Lim, M., Abdullah, A., & Jhanjhi, N. Z. (2021). Performance optimization of criminal network hidden link prediction model with deep reinforcement learning. *Journal of King Saud University-Computer and Information Sciences*, *33*(10), 1202-1210.

[36] Hussain, K., Hussain, S. J., Jhanjhi, N. Z., & Humayun, M. (2019, April). SYN flood attack detection based on bayes estimator (SFADBE) for MANET. In *2019 International Conference on Computer and Information Sciences (ICCIS)* (pp. 1-4). IEEE.

[37] Srinivasan, K., Garg, L., Datta, D., Alaboudi, A. A., Jhanjhi, N. Z., Agarwal, R., & Thomas, A. G. (2021). Performance comparison of deep cnn models for detecting driver's distraction. *CMC-Computers, Materials & Continua*, *68*(3), 4109-4124.

[38] Muhammad Ibrahim Khalil, N.Z. Jhanjhi, Mamoona Humayun, SivaKumar Sivanesan, Mehedi Masud, M. Shamim Hossain, Hybrid smart grid with sustainable energy efficient resources for smart cities, Sustainable Energy Technologies and Assessments, Volume 46, 2021, 101211, ISSN 2213-1388, https://doi.org/10.1016/j.seta.2021.101211

[39] A. Almusaylim, Z., Jhanjhi, N. Z., & Alhumam, A. (2020). Detection and mitigation of RPL rank and version number

attacks in the internet of things: SRPL-RP. *Sensors*, *20*(21), 5997.

[40] Lim, M., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). Hidden link prediction in criminal networks using the deep reinforcement learning technique. *Computers*, *8*(1), 8.

[41] Fatima-tuz-Zahra, N. Jhanjhi, S. N. Brohi and N. A. Malik, "Proposing a Rank and Wormhole Attack Detection Framework using Machine Learning," 2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS), Karachi, Pakistan, 2019, pp. 1-9, doi: 10.1109/MACS48846.2019.9024821.

[42] Humayun, M., Jhanjhi, N. Z., Alruwaili, M., Amalathas, S. S., Balasubramanian, V., & Selvaraj, B. (2020). Privacy protection and energy optimization for 5G-aided industrial Internet of Things. *IEEE Access*, *8*, 183665-183677.

[43] Lee, S., Abdullah, A., Jhanjhi, N., & Kok, S. (2021). Classification of botnet attacks in IoT smart factory using honeypot combined with machine learning. *PeerJ Computer Science*, *7*, e350.

[44] Zaman, N., Low, T. J., & Alghamdi, T. (2014, February). Energy efficient routing protocol for wireless sensor network. In *16th international conference on advanced communication technology* (pp. 808-814). IEEE.

[45] Khan, N. A., Brohi, S. N., & Jhanjhi, N. Z. (2020). UAV's applications, architecture, security issues and attack scenarios: A survey. In *Intelligent Computing and Innovation on Data Science: Proceedings of ICTIDS 2019* (pp. 753-760). Springer Singapore.

[46] Ali, S., Hafeez, Y., Jhanjhi, N. Z., Humayun, M., Imran, M., Nayyar, A., ... & Ra, I. H. (2020). Towards pattern-based change verification framework for cloud-enabled healthcare component-based. *Ieee Access*, *8*, 148007-148020.

[47] Shah, I. A., Jhanjhi, N. Z., Humayun, M., & Ghosh, U. (2022). Impact of COVID-19 on Higher and Post-secondary Education Systems. In *How COVID-19 is Accelerating the Digital Revolution* (pp. 71-83). Springer, Cham.

[48] Shah, I. A., Jhanjhi, N. Z., Amsaad, F., & Razaque, A. (2022). The Role of Cutting-Edge Technologies in Industry 4.0. In *Cyber Security Applications for Industry 4.0* (pp. 97-109). Chapman and Hall/CRC.

[49] Shah, I. A. (2022). Cybersecurity Issues and Challenges for E-Government During COVID-19: A Review. *Cybersecurity Measures for E-Government Frameworks*, 187-222.

[50] Shah, I. A., Wassan, S., & Usmani, M. H. (2022). E-Government Security and Privacy Issues: Challenges and Preventive Approaches. In *Cybersecurity Measures for E-Government Frameworks* (pp. 61-76). IGI Global.Shah, I. A. (2022). Cybersecurity Issues and Challenges for E-Government During COVID-19: A Review. *Cybersecurity Measures for E-Government Frameworks*, 187-222.

[51] Shah, I. A., Sial, Q., Jhanjhi, N. Z., & Gaur, L. (2023). Use Cases for Digital Twin. In *Digital Twins and Healthcare: Trends, Techniques, and Challenges* (pp. 102-118). IGI Global.

[52] Shah, I. A., Jhanjhi, N. Z., & Laraib, A. (2023). Cybersecurity and Blockchain Usage in Contemporary Business. In *Handbook of Research on Cybersecurity Issues and Challenges for Business and FinTech Applications* (pp. 49-64). IGI Global.S Umrani, S Rajper, SH Talpur, IA Shah, A Shujrah - Indian Journal of Science and Technology, 2020

[53] Shah, I. A., Sial, Q., Jhanjhi, N. Z., & Gaur, L. (2023). The Role of the IoT and Digital Twin in the Healthcare Digitalization Process: IoT and Digital Twin in the Healthcare Digitalization Process. In *Digital Twins and Healthcare: Trends, Techniques, and Challenges* (pp. 20-34). IGI Global.

[54] Kiran, S. R. A., Rajper, S., Shaikh, R. A., Shah, I. A., & Danwar, S. H. (2021). Categorization of CVE Based on Vulnerability Software By Using Machine Learning Techniques. *International*

*Journal*, *10*(3).