Smart Thermometer for Food Safety Management

Tiago Almeida
Instituto Superior Técnico, University of Lisbon
Lisbon, Portugal
tiago.teixeira.de.almeida@tecnico.ulisboa.pt

Abstract— **Efficient food safety measures, namely temperature control, can contribute to mitigating the global problem of foodborne diseases. Temperature control procedures are in place in many establishments but often lack efficiency and can strongly benefit from process optimization, both on the perspective of temperature control and record taking, mainly when measurements are required to be taken through the use of a thermometer. With this work, the objective is to create a functional prototype of a smart thermometer used to control temperatures more efficiently and reliably, enabling greater control over the measurements that food safety management systems, including HACCP and traceability systems, require. The proposed solution includes features such as allowing user authentication, machine and product RFID identification, presentation of daily tasks, and automatic remote submission to a database. Feedback was received from an HACCP consulting company responsible for more than 400 establishments. A survey was also conducted to four food safety managers from different sectors of the food industry, together responsible for food safety in more than 700 establishments. The received feedback confirmed that food safety managers see value in the presented solution. It's their belief that it can be used to prevent possible errors caused by the human factor and optimize the temperature control processes.**

Keywords- HACCP, FSMS, Temperature Control, IoT

I. Introduction

Every year, about 600 million people, almost 1 in 10, fall ill, 420 thousand die, and 33 million disability-free life years are lost due to foodborne illnesses[1]. Unsafe food represents not only a burden on world public health but also a burden on the world economy. A study from the World Bank [2] states that the impact of FBDs on low- and medium-income countries is about US$ 110 billion, where US$ 95.2 billion are attributed to productivity loss and US$ 15 billion to the cost of treating FBDs. Different studies appoint temperature abuses as a critical contributing factor that leads to FBD [3]. Some examples of temperature abuse that contribute to the survival and/or proliferation of etiologic agents responsible for FBDs are insufficient time and/or temperature during cooking/heating process, insufficient time and/or temperature during reheating, inadequate thawing (followed by insufficient cooking), allowing foods to remain at room or warm outdoor temperature for several hours, inadequate cold-holding temperature, insufficient time and/or temperature during hot holding [4]. To better control and prevent the issues mentioned above, Hazard Analysis and Critical Control Points (HACCP) based FSMS along with traceability systems are mandatory in several countries. In this work's scope, a HACCP consulting company responsible for more than 400 establishments supported the development of the smart thermometer prototype by providing relevant feedback regarding their experience on implementing HACCP system with their clients.

The company highlighted that, in fact, one of the most common issues that foodservice establishments had with their HACCP FSMS was the efficient and proper control of temperatures, mainly the ones that should be controlled through the use of handheld thermometers. In some cases, processes were manual, and the records kept on paper; in others, there was a degree of automation with records inserted in a database through a smartphone or tablet at the time of the control or at a later time. Either way, the company concluded that flaws arose from either of these processes, showing that exists room for improvement. The system developed within this research scope focuses on a new approach to temperature control. It is meant to improve the way food temperature control is made in current HACCP and traceability systems. The objective is to create a functional prototype of a smart thermometer that will be used to control temperatures more efficiently and reliably, enabling greater control over the measures that HACCP FSMS requires. The system comprises a smart thermometer and an additional backend that will provide an application programming interface (API) that can either be deployed on-premises, on the cloud, or even using a hybrid approach.

II. Literature review

A. HACCP

HACCP is a systematized, science-based, documented method internationally recognized as the best way to control significant food safety hazards[5]. Focusing on prevention instead of end-product testing, using a systematic approach, hazards are identified, and assessment and control of risk are made through the definition and control of CCPs [2]. This enables companies to reduce contamination risks to an acceptable level while reducing the dependency on final product inspection [6]. HACCP is widely adopted around the world. In the last few decades, several countries have made

HACCP principles mandatory by introducing legislation requiring to apply HACCP principles to varying degrees [5].

HACCP's seven principles were first described by the NACMCF and by the Codex. These seven principles set the basis for the requirements for the application of HACCP [7]:

Principle 1: Conduct a Hazard Analysis
Principle 2: Determine the Critical Control Points (CCPs)
Principle 3: Establish Critical Limits
Principle 4: Establish a monitoring system for each CCP
Principle 5: Establish Corrective Actions
Principle 6: Establish Verification Procedures
Principle 7: Establish Documentation and Record Keeping

HACCP, by itself, is not enough to ensure safe food. Strong prerequisite programs (PRPs) should be in place before implementing a HACCP system [8].

One important category of PRPs are operational PRPs (oPRPs). While PRPs are put in place to deal with generic hazards that do not represent an immediate and significant threat to food safety, oPRPs are specific to a hazard(s) and/or process that represent a more substantial threat [8].

When properly planned, implemented, and maintained, a HACCP based FSMS can bring several benefits, such as improvement in food safety due to better production procedures, greater control of critical points, constant monitoring, and higher hygienic and cleaning standards [9]. Nevertheless, several difficulties and barriers to a proper implementation exist, as well as problems derived from a poorly implemented, monitored, and/or maintained system [10]. The most obvious problem is related to food safety. If a system is not properly implemented, instead of contributing to the solution, it will probably contribute to worsening it. Hazard analysis and CCP determination can be challenging to get done correctly without the right knowledge, and if these are done wrong or if after planned they are neglected or controlled the wrong way then the system will most likely fail [11].

B. *Traceability in the Food Supply Chain*

The International Organization for Standardization (ISO) defines traceability as the "ability to trace the history, application, or location of that which is under consideration" and traceability system as "totality of data and operations that is capable of maintaining desired information about a product and its components through all or part of its production and utilization chain" [12]. Applied to the food supply chain, this translates into knowing who, what, where, when, and why something happened to a particular product or ingredient [13]. Traceability is therefore used as a way of responding to potential food safety issues, ensuring that food is safe to eat. It gives authorities or food businesses the ability to, when a risk is identified, trace it back to its origin and rapidly isolate the problem preventing contaminated food from reaching the consumer [14]. Although traceability is mainly viewed as a tool to ensure that food is safe by providing the means for a focused and fast recall if needed and to prove food authenticity, it is also an excellent tool for improving quality as it allows producers to pinpoint where faults occur, what needs to be improved and even to audit their suppliers or buyers (for example to know if the buyer uses the products in an intended way) [11].

C. *Temperature Control*

To measure temperature in storage areas, different types of devices are used:

- Temperature loggers;
- Package indicators;
- Fixed sensors;

When it comes to the implementation of food temperature control that requires an active measure by the user, for example, during the process of food preparation, different devices can be used such as bi-metallic stem thermometers, digital thermometers, and temperature sticks or sensor strips [15]. Two of the most used types being thermocouple and infrared thermometers[16]:

- Thermocouple thermometers;
- Infrared thermometers.

Wireless Sensor Networks (WSN) can be implemented by using monitoring devices such as the ones described (thermocouple sensors, infrared, infrared sensors, or others) to real-time monitor and control the temperature of the targeted products continually, remotely, and accurately [17].
All kinds of temperature control devices previously presented fit in one of two categories: fixed devices (e.g., sensors, temperature loggers, package indicators) or handheld thermometers. However, these solutions are not mutually exclusive in establishment operations. They compliment each other and are often used together.
For FSMS effects, an efficient and sustainable process must be implemented in the food establishment [18]. Therefore, some temperature control processes depend on checklists that detail all the needed actions (which translate into required temperature readings) to check the conditions of food in different critical situations.
Still, most restaurants use thermometers with no communication capabilities and depend on the manual insertion of data in tablets or others, which leads to data being manually processed or physically introduced to a digital platform, many times resulting in a complicated system [20]. This can be a problem as employees still have a high probability of committing mistakes such as inserting wrong values and measuring temperatures in the wrong place or product.

III. REQUIREMENTS

The objective of the work developed in the scope of this work is to create a functional smart thermometer prototype used to control temperatures more efficiently and reliably during food-related establishments' daily operations. It can be used along the entire supply chain wherever and whenever temperature readings must be performed using a thermometer. To help mitigate the previously described issues and enable greater control, efficiency and reliability over the measurements that HACCP FSMS requires, the smart thermometer can either be integrated into existing systems or used as a standalone system. When an establishment already has a digital system, this solution can be integrated and complement it; when they do not have such systems, this solution can be used as a standalone solution., while also eliminates the need for using paper for this purpose. In both cases, the user no longer needs to manually insert reading results in a paper form or device.

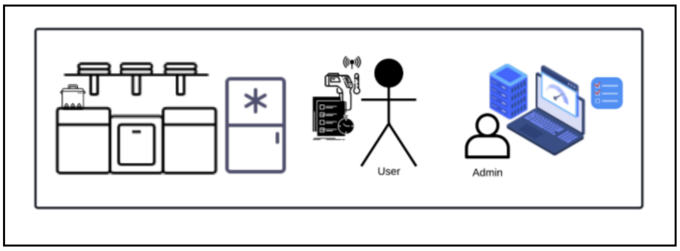

Figure 1. The user can identify machines/products, view tasks scheduled by the admin and automatically upload measures to a database

A. User Classes and Characteristics

Two classes have been considered: users and admins. The characteristics are the following:

- Users: users of the smart thermometer can be technically not proficient. Its use requires no particular education level or experience. When not authenticated users shall be able to use the device as a regular thermometer. When authenticated, they shall have the possibility to take scheduled and non-scheduled measurements that are automatically uploaded to an online database.
- Admins: Must have a more advanced technical understanding. Low computer skills are needed. It is necessary to know how to use the Internet to manage the data that needs to be inserted into the database. System admins shall be able to create, read, update, and delete entities such as stores, machines, products, tasks, users, measurements, and others. An admin can also be a user.

B. Main Features and Functional Requirements

The functional requirements presented are divided by user class and feature from where the requirements originate. Requirements are numbered and can repeatedly appear from feature to feature. These represent the essential features that the system shall have.

The requirements used in the different features are the following:

FR01 The system shall allow a user to take internal product temperature measures.

FR02 The system shall allow a user to take surface product temperature measures.

FR03 The system shall have an employee authentication mechanism.

FR04 The system shall allow to present a list of daily tasks.

FR05 The system shall allow authenticated users to see the available non-completed temperature tasks.

FR06 The system shall allow authenticated users to verify if the readings are being done in the right local.

FR07 The system shall have a machine identification mechanism.

FR08 The system shall have a product identification mechanism.

FR09 The system shall allow submitting measurements to a remote database.

FR10 The system temperature measurement features shall work in case of not having a wireless connection.

FR11 The system shall allow admins to manage stores.

FR12 The system shall allow admins to manage machines.

FR13 The system shall allow admins to manage products.

FR14 The system shall allow admins to manage users.

FR15 The system shall allow admins to manage scheduled temperature measurement tasks.

FR16 The system shall allow admins to manage non-scheduled temperature measurements.

FR17 The system shall allow bulk creation of objects.

FR18 The system shall allow the filtering of data.

For the employees user class, the features available are the following:

Feature - Scheduled Temperature Measurements: This system feature allows an authenticated user to take measures remotely scheduled by an admin and submit them to a database. Each task has the date and time when it shall be completed, and the list is updated daily. The user can filter the task list by scanning a machine, retrieving all the associated tasks related to the scanned machine and products. The measured object must be verified through its identification. The user must select an available task through a list and/or by scanning a machine or product, take the corresponding measure, and submit the result.

<u>Functional Requirements:</u> FR01, FR02, FR03, FR04, FR05, FR06, FR07, FR08, FR09, FR10

Feature – Non-scheduled Temperature Measurements: This system feature allows an authenticated user to take temperature measurements that were not scheduled by the admins. After identifying a machine and/or product, the measurement can be taken and submitted.

<u>Functional Requirements:</u> FR01, FR02, FR03, FR07, FR08, FR09, FR10.

Feature – Normal Temperature Measurements: This system feature allows the system to work as an ordinary thermometer. Any user can take a measurement.

<u>Functional Requirements:</u> FR01, FR02.

Feature – Login: This system feature allows a user to login into the system in case of being logged out.

<u>Functional Requirements:</u> FR03

For the admins user class, the features available are the following:

Feature – Managing Stores and Controlling Data: This feature shall allow an admin to remotely create, view, update and delete any entity related to the system's usage. It includes the ability to schedule and monitor measurements.

<u>Functional Requirements:</u> FR11, FR12, FR13, FR14, FR15, FR16, FR17, FR18.

C. External Interface Requirements

Users and admins interact with the system through different interfaces:

- Users: user interaction is carried out through a mobile interface associated with the temperature measuring device.
- Admins: admin interaction with the system is executed through a web browser.

The system shall have wireless communication to access the remote database, communicate with other FSMS, and identify machines and products.

The system shall have an API that allows the communication between the smart thermometer and the backend support services. Through this API the smart thermometer shall be able to submit non-scheduled measurements to the remote database, submit task results and get data updates (machines, products, scheduled tasks).

This API shall also enable the system to be integrated into other existing systems. Through the API backend functionalities are exposed. It is possible to create stores, machines, schedule tasks and fetch previously existing data.

D. Non-functional Requirements

<u>Performance Requirements:</u>

NFR1 Measurements cannot be slower than the current process.

NFR2 Measurement resolution shall be 0.1ºC.

NFR3 Physical Probe accuracy shall be 1ºC +-1% of reading.

NFR4 Physical probe range shall be -100ºC to 700ºC minimum

NFR5 System boot time shall be less than 500ms.

Safety Requirements:
NFR6 Physical probe should be food compatible.

Security Requirements:
NFR7 Users shall be authenticated.
NFR8 Network communications shall be secured.
NFR9 Server data shall be secured.
NFR10 Physical Device shall be secured.

IV. System Architecture

A. System Context View

In this section, the focus will be on the system as a whole and the interactions that users and external systems have with it, as shown in Figure 2.

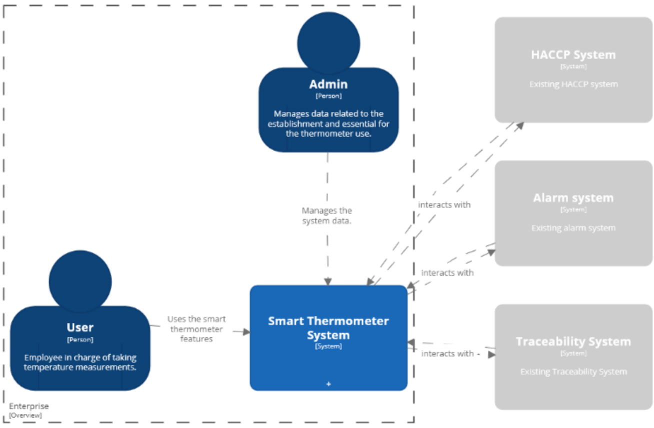

Figure 2. System context diagram for the smart thermometer system

External systems are systems other than the smart thermometer system that sit outside its scope but can interact with it. These are existing HACCP, alert, or traceability systems.
For example, the case where the smart thermometer system is deployed in an establishment with a digital HACCP system already implemented. The existing system might already have all the information in digital support. If this is the case, then the existing HACCP system can interact with the smart thermometer system and make the information readily available instead of being inserted again in the smart thermometer system. Another example could be the smart thermometer system could send all the completed measurements directly to a previously existing database.

B. Smart Thermometer System View

By expanding the smart thermometer system block presented, in Figure 2, a diagram of the system composition overview is obtained (Figure 3).
The system shall be composed of the smart thermometer itself, an admin web app, an API, a backend, and a database.
The system's backend will provide a REST API that will be the primary interface that the smart thermometer and external systems have to access the backend services and the database. It will also serve the admin web application. The backend can also directly interact with external systems, for example, to send measurements to another system.
The REST API shall have endpoints that allow the smart thermometer to make API calls asking for data updates that include machines, products, users, and scheduled tasks. It shall also provide an endpoint for the smart thermometer to send the

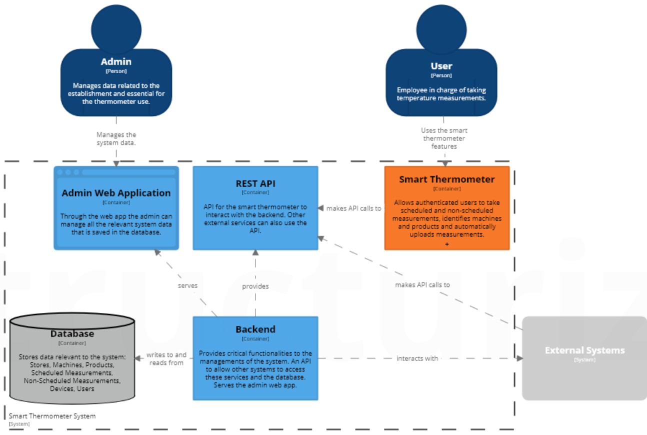

Figure 3. Smart Thermometer System Diagram

completed measurements. External systems can also consume the API, either to ask for data or to manage it.
The admin web application shall allow the admin to manage all the relevant data necessary for the system to function. Through the application, the admin shall be able to create, update or delete stores, machines, products, and users. Schedule new tasks, assigning what, where, and when should be measured. And view the existing data.
The database shall store all the essential data that has already been mentioned in the previous points.
The backend, API, and database can be deployed either on the cloud or the client premises.

C. Smart Thermometer

The focal point of this work is the smart thermometer itself. The backend, API, and the admin web application are just supporting services to enable the smart thermometer features. Figure 4 shows the smart component diagram.

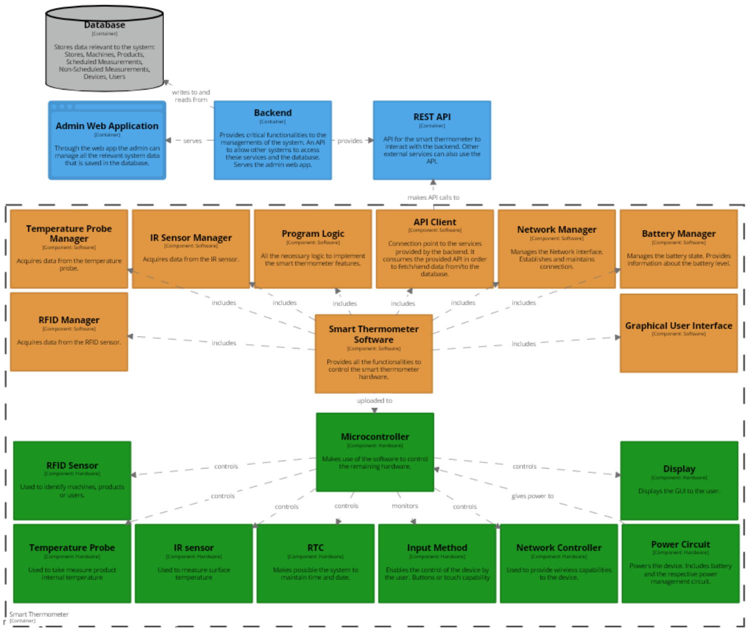

Figure 4. Smart Component Diagram

The necessary hardware is the following: RFID sensor (to identify machines, products users); food compatible temperature probe (to measure product internal temperature); IR sensor (to measure surface temperature); Real-time clock (RTC) module (to keep time and date accurate); Display (to present information to the user); input method (physical buttons or touch capabilities via the display, so the user can control the

device); network controller (to provide wireless capabilities); battery (to power the device); microcontroller (to control the remaining hardware by making use of the system software).
The system software shall comprise all the necessary modules to control the corresponding hardware as well as the program logic necessary to implement the desired features and the API client which will consume the API provided by the backend.

V. Prototype Implementation

A. Smart Thermometer Prototype

Although close to the final product form, the prototype has the liberty of being built using components that enable faster implementation. To achieve faster implementation, the M5Stack system was chosen to be used as the system base [21]. The system is open-source, makes use of an ESP32 microcontroller, comes in an encasement with a display and all the necessary components for its functioning, a 2.4G antenna, power management controllers, and interface components. It also allows the direct integration of components such as an RFID reader and an IR Sensor. It supports several development platforms and programming languages such as Arduino, Blockly language, and Micropython. The main factor to choose M5Stack was the ability to take advantage of the already encased hardware, which covers all the architectural needs while having the possibility to extend it if needed, along with the option to implement custom made software to control the device.
Nevertheless, the prototype could also be achieved using individual components and other development boards or by building a custom printed circuit board, PCB, and encasement.

Microcontroller: The used microcontroller is an ESP32 [22]. As the ESP32 has integrated Wi-Fi, there is no need for an external network module, allowing for a smaller design and out of the box network capability. The built-in analog to digital converter (ADC), however, proved to be inaccurate. It suffers from noise and linearity problems, which did not allow for accurate readings when necessary (e.g., temperature probe readings). This has been solved by using amplifiers with built-in ADC. The built-in RTC also proved to be imprecise, showing drift over time. As recording time and date is a necessity, this represents a setback. However, feedback received from the HACCP consulting company stated that a testing version with no offline mode was more than acceptable. Allowing the user to take measurements only when Wi-Fi is available allows the transfer of the responsibility of maintaining measurements date and time from the device to the backend. In a more advanced version of the smart thermometer, an external RTC can be added and the problem easily solved.

Temperature Probe: the most common type of probes used to measure internal product temperature are thermocouple probes. There are different thermocouple types, some examples being type K, J, T, and E. The chosen type was a Type K thermocouple. The reasons are: being the most commonly available, inexpensive, meet temperature range and accuracy needs, and are more chemically inert than Type T and J. The other main reason was that the device available for comparison also uses a type K thermocouple. The chosen thermocouple was an RS PRO Type K Thermocouple with 150mm length and 1.5mm diameter [84].
Different solutions have been tried. First, an analog output amplifier, AD8495, was tested. However, the ESP32 ADC suffers from accuracy problems, and the obtained measurements were not reliable. Therefore a digital thermocouple amplifier was used, namely the MAX31856 [23]. The MAX31856 performs cold-junction compensation and digitalizes the signal from any type of thermocouple. It also includes automatic linearization correction and a 19 bit ADC. To easily integrate this board with the existing design, it was used the Adafruit MAX31856 breakout board. This board was then placed into an M5Stack module case, and wires were soldered to the available pins. A thermocouple mini plug was attached to the module and connected to the MAX31856, allowing for the thermocouple probe to be detachable.

IR Sensor: One of the most commonly available IR temperature sensors is the MLX90614. These are used in a wide range of applications and are easy to integrate. They already come with the necessary low noise amplifier, a 17-bit ADC, and a digital signal processor to achieve higher accuracy and resolution. [24].

RFID Sensor: Considering that RFID will be used to identify users, machines, and products, the range necessary to scan each tag must be short. To make the prototype development more straightforward, the chosen RFID reader was the one made available by M5Stack. The RFID module uses an RC522 module. It works in the 13.56MHz frequency, and the reading and writing distance is less than 20mm.

Display: The used display, which corresponds to the m5stack display, it is a 2.2" TFT display module with 320x240 pixels. It uses an ILI9341 driver that connects via SPI to the microcontroller.
One thing to have in consideration is the fact that the MAX31856 and the display share the SPI bus. When the display and the MAX31856 were connected at the same time a direct memory access error (DMA) happened. The error was: "spi_master: spi_device_queue_trans(620): txdata transfer > host maximum". Meaning that the data transfer was more than what the host (in this case the microcontroller) could handle. To solve this issue it was necessary to increase the factor argument of the ILI9341. By doing this the display consumes less RAM, the downside is that the refresh rate is lower. Nevertheless, for the smart thermometer case the difference is not noticeable.

Buttons: Three physical buttons are used. They are connected to pins 37, 38, and 39 of the microcontroller.

Power Circuit: Although the M5Stack system comes with the powering circuit already implemented, and no work was necessary to make it function, it was analyzed to understand how it works and make it possible, in the future, to implement a similar solution in a custom board. The powering circuit includes a 3.7V rechargeable lithium-ion battery, an IP5306_I2C, and EA306 chips. The IP5306 is a multi-function power management system on chip (SoC). It has an integrated boost converter to step up battery current to 5V. It also provides battery charge management and by connecting it using I^2C to the microcontroller, it is possible to monitor the battery level. The EA306 has three buck converters and is used to regulate the 5V voltage that comes from the IP5306 down to 3.3V. This way, the system has available either 5V or 3.3V for its use.

B. Smart Thermometer Software

Although the M5Stack comes from factory with its own software that allows programming the device by using its own version of micropython or blockly, these proved to be very basic. Documentation was scarce and it did not give the necessary freedom to customize the device. Therefore to implement the desired features, it was required to create custom software for the smart thermometer.

As mentioned before, the ESP32 supports micropython, Arduino and ESP-IDF. The choice was to use micropython as it is the ideal language for fast prototyping due to being high level, fast and easy to implement. Although micropython is not as lightweight as Arduino or ESP-IDF, it has all the required software libraries and if required it is extensible and can be used together with C or C++.

To implement the GUI the choice was to use an open-source graphic library called Light and Versatile Graphic Library (LVGL) [25]. LVGL is specially designed to work in embedded devices. Although being written in C, there are bindings that make it compatible with micropython. It is also compatible with the ILI9341 display driver.

To program the device, an event-driven programming approach was used. Two main reasons contributed to that: the device program flow being determined by user actions such as a user pressing a button and scanning an RFID card and the need to ensure a non-blocking UI.

Implemented Features

Figure 5 shows the main program flow after turning on the thermometer.

<u>Scheduled Measurements</u>

To use the scheduled measurements feature the user selects the option “Scheduled Measurement” when on the main menu. Once this option is selected, the user must select one of the options: “Update tasks”, “Scheduled Tasks List”, “Scheduled Tasks RFID” or “Home”.

If the user selects “Update Tasks” then the update tasks menu is shown, and the user can choose to update the scheduled tasks list. The task update process is described in subsections “Update Task List”.

If the user selects “Scheduled Tasks Lists”, the system will display the list of possible tasks to choose from.

If the user selects “Scheduled Tasks RFID” then the system expects an RFID (machine or product) badge to be scanned. If the identification is not successful, the system will keep asking until successful or until the user decides to cancel; once the identification is successful, the system will display the tasks associated with the machine or product identified.

At this moment, in any of the last two cases described, the user must select one of the tasks from the list presented and take the temperature measurement. Finally, the user can choose to submit the measurement to the database. After pressing submit, the system presents the main menu once again.

<u>Non-scheduled Measurements</u>

To use the non-scheduled measurements feature the user selects the option “Non-scheduled Measurement” when on the main menu. Once this option is selected, the user must select one of the options: “Scan Machine Label”, “List Machines” or “Home”.

If the user selects “Scan Machine Label” then the system expects a machine RFID badge to be scanned. If the identification is not successful, the system will keep asking until

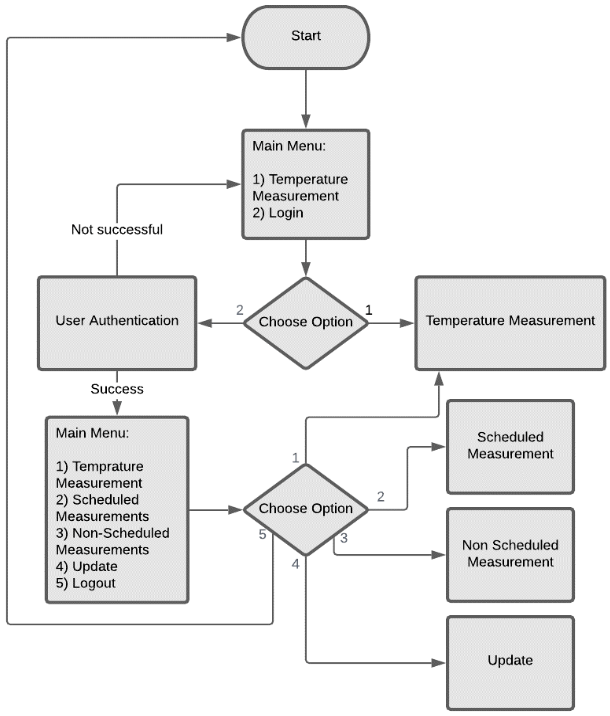

Figure 5 – Main flowchart

successful or until the user decides to cancel; once the identification is successful, the system will display the product’s list of the machine identified.

Similarly, if the user selects “List Machines”, the system will display the list of possible machines to choose from. Once the user selects one machine, the system will display the product’s list of the machine selected.

In any of the cases, at this moment, the user must select one of the products from the list presented and take the temperature measurement (that can be repeated as many times as necessary). Finally, the user can choose to submit the measurement to the remote database. After pressing submit, the system presents the main menu once again.

<u>Normal Measurements</u>

On the normal measure feature the user selects the option “Normal Temperature Measurement” when on the main menu. Once this option is selected, the user can start measurement or return to the main menu. The user repeats the measurements as many times as necessary.

<u>Authentication Mechanism</u>

The system’s process for user authentication starts when the device reads the user RFID label. Once the system scans the badge, it compares it to the user list in the filesystem. If the IDs don’t match the process repeats itself until a match is found (i.e. a valid RFID badge is presented). Once this happens the user authentication is completed, and the login is confirmed.

<u>Schedule Measurement Submission</u>

The system’s process for submitting a scheduled measurement starts by making sure that WiFi is available. If it is not available or if it is off and the HTTP request was unsuccessful, the measurement data is saved in the filesystem. When WiFi is available, a patch HTTP request with the measurement data is sent to the server. If the request was successful and the data was saved in the file system, then it shall be erased.

Once the data as been saved to the file system or successfully sent to the server, the task is considered completed, and therefore erased and the user no longer have access to it.

<u>RFID machine identification for scheduled measurements</u>

The system's process for identifying tasks related to a certain machine (for a product, it would be the same) starts by reading an RFID label attached to that machine. Once the system scans the badge, it searches in the task list for tasks related to the machine. If found, all the tasks associated with the machine are fetched.

RFID machine identification for non-scheduled measurements

The system's process for identifying a machine and product that shall be the target of a measurement starts by reading an RFID label. Once the system scans the badge, it compares it to the machine list in the filesystem. If the IDs don't match the process repeats itself until a match is found (a valid RFID badge is presented). Once this happens the corresponding product list is fetched.

Update Smart Thermometer Machine List

In order for the smart thermometer to work when network connection is offline, it needs to save essential data made available by the server in its own filesystem. A subset of that data is the list of machines present in the store, along with the products included in each machine.

These lists do not suffer frequent changes and can be large. Therefore, in this case, a caching system is used in order to prevent the same data from being fetched once again.

The system's process for updating the machine and products lists starts by loading the ETag available from the previous update. Then if WiFi is available, an get request is submitted to the server. This request will contain the ETag present in a "if-none-match" header. If server data as changed and the request is successful, the task list is returned by the server and is written to the file system and the ETag is updated. If data has changed, nothing happens.

Update Smart Thermometer Scheduled Tasks List

The other subset of that data essential for the system proper functioning is the list of scheduled measurements (tasks).

The system's process for updating the scheduled measurements tasks starts by making sure that WiFi is available. If WiFi is available, a get request is submitted to the server. If the request is successful, the task list returned by the server is written to the file system. Otherwise the user is warned

To implement the backend services, the REST API, and the database, Django REST Framework (DRF) was the chosen tool [7]. DRF allows these to be implemented simply and efficiently. It has extensive documentation, and it is widely tested and used. DRF handles relational databases, object-relational mapping, can serve a native admin panel and provides a REST API.

DRF allows for integration with several different database management systems, PostgreSQL, SQLite, MySQL, and others. In this case, PostgreSQL is used due to out-of-the-box compatibility with DRF and for allowing integration with cloud providers like Google App Engine or Azure. To implement the backend, API, admin web application and database, the first step was to declare the data models. Data models are the object relational-mapping between Python data objects and database models. The following models have been created: store, user, device, machine, product, measurement, scheduled measurement.

The second step was to declare all the serializers used to transform resource representations and send them from the server to the client and vice versa. These serializers define which fields of a data model are sent when the client asks or submits a particular resource.

Finally, to set endpoints that can be the target of HTTP requests, some views were created. These views define what kind of HTTP requests can be made to each endpoint. Some endpoints can only handle safe method requests like GET, others can also handle POST requests. These endpoints will be consumed by the smart thermometer, either to get data updates, or to submit measurements. External services can also use them in order to create or manage data as mentioned in the previous chapters.

Regarding security there are four main components that were taken into consideration. User Authentication, Device Security, Communication Security and Server Security.

To authenticate users with the device the choice was to take advantage of its RFID functionality. Each user has a personal RFID tag that must be used in order to access the smart thermometer advanced functionalities. At the server side, the admin must be authenticated in order to access the backend services.

To further secure communications to the server, either from the smart thermometer or from external services, HTTPS shall be used to assure integrity and confidentiality of data. The ESP32 has transport layer security (TLS) support. The device shall also authenticate itself through the use of json web tokens (JWT) or even by using pre-provisioned certificates that would enable mutual TLS.

VI. Results

A. *Proposed Solution: Smart Thermometer*

Figure 6 shows the prototype developed and its temperature probe.

Figure 6 – Smart Thermometer Prototype

As a result of the implementation previously discussed, the thermometer can display the following menus:

- Login menu
- Main menu with authenticated user
- Main menu with non-authenticated user
- Update menu
- Menu for non-scheduled tasks
- List of scheduled tasks

Furthermore, and on a web app, it is possible for the admin to access the admin view, to allow them to manage users, tasks, machines, stores and others.

B. *Smart Thermometer Performance*

In order to assess the thermometer performance, a practical experiment was conducted using the smart thermometer and a reference thermometer. The HACCP consulting company

provided the reference thermometer, which is ThermaQ® Blue Thermocouple Alarm Thermometer [26]. To conduct the practical experiment, it was first created an ice bath. In an ice bath, the water temperature shall reach 0ºC (it is not exact, since the water used is not distilled due to the presence of minerals nd other substances)[27]. The water was then heated until the boiling point, where the temperature should be around 100ºC. Figure 7 graphically represents these measurements and shows that the values of the thermometers are quite similar and overlap in almost all the points. In red, it is represented the prototype measurements and in blue the reference thermometer's.
The most discrepant measures are the first and last one, which present a variation of +/- 2ºC. Taking into account the specifications of the hardware used and the results presented here, it is possible to conclude that in future versions the thermometer developed can reach the accuracy degree of the reference thermometer.

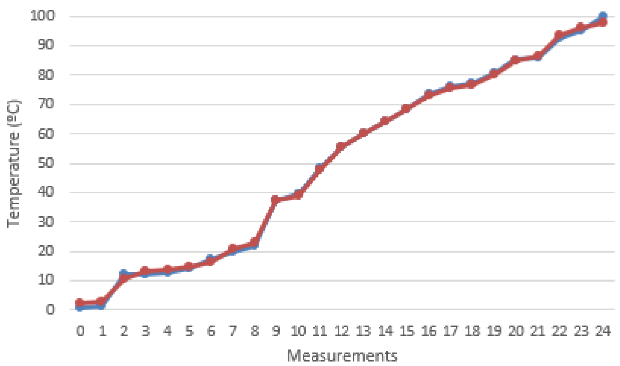

Figure 7 – Ice bath to boiling point temperature measurements

Although the hardware components chosen theoretically allow the thermometer to have an accuracy similar to the reference thermometer, the discrepancies observed can be due to the hardware utilized in the prototype not being optimized for this use (extreme conditions, such as the ones encountered in a kitchen environment).

C. Temperature Control and Records – Market Perspective

Along the period of this research there was a constant interaction with an HACCP consulting company. The company's feedback allowed to obtain a broader view of the catering/food supply market, and gather all the requirements to develop the solution presented.
The HACCP consulting company is specialist in the digitalization and automatization of data collection, tasks and procedures through the development of custom tailormade software and the introduction of IoT technologies. Currently they are responsible for the services in more than 400 establishments.
The initial information obtained from the company is in line with the results of the questionnaires that will be discussed next: most clients with which the company works use handheld thermometers with on-screen temperature for temperature control. After the measurements, and for record taking purposes, these are input on specific Software or paper (before Software adoption).
One of the major concerns of the company as data collectors is to ensure the data is reliable: measured at the defined time, using the correct method, and that values are not forged to fit the range of pretended values. Using Software instead of a paper solution allows store and unit managers to quickly analyze the collected data and act on it guaranteeing a closer control of the operation and employees. Decision making is better informed, and problems reach the upper chains (or responsible entities) faster, allowing faster action.
Mainly, it allows managers to control at a distance and in real-time (close to real-time when offline mode is used) whether registries are being filled and if any problems are reported in a plethora of units.
When collecting temperature data, the company verified that, currently, the use of a mobile device, (such as a phone or tablet) together with a handheld thermometer (Bluetooth or manual input), is not optimal as it occupies employees' both hands and makes it take longer. When using Bluetooth thermometers, sometimes the simple synchronization between devices becomes a problem. At the moment using multiple pieces of hardware the main challenges include damages or disappearances, which are reported problems that delay the operation.
A survey was also prepared with the goal of confirming the feedback given by the HACCP consulting company and to better understand what processes are implemented in different kinds of establishments along with the current needs.
Four different entities were contacted in order to obtain feedback from different types of establishments that also differ in size: one entity in the business of food distribution (retail), managing more than 600 establishments; one industrial entity (production of frozen goods); one entity responsible for a restaurant chain, managing between 51 to 100 establishments; one small catering business, responsible for one restaurant. In total, the survey analyzed a universe of around 700 establishments, meaning that although it was not an extensive survey it still represents a significant number of establishments with a relevant accumulated experience that cover different parts of the food supply chain.
When asked if they consider that a smart thermometer system would benefit the records' reliability, the answers were positive, as there is a belief that it would reduce the human errors mentioned before.
When considering the temperature control and record taking in the different establishments, all respondents except one, who asked for more information, accepted the smart thermometer as a solution that would simplify the system due to saving resources, such as time, and mitigate errors. The respondent who asked for more information was the one responsible for the big restaurant chain's food safety quality. He stated that they would need more information to understand the system and know if they would benefit from it because they already had a WSN in place. After a brief phone interview, they acknowledged the solution's relevance and stated that it would fit and much likely improve their current practices. They also showed a particular interest in the fact that it would be able to identify machines and make the process of knowing which tasks they should complete at each place easier.

D. Smart Thermometer Prototype Costs

The costs of the smart thermometer solution developed in this work can be divided into three parts: software and hardware development, and hardware parts.
With respect to the costs of development it is impossible at this point to estimate or evaluate it, thus only hardware parts cost is evaluated here.
When it comes to the hardware, it was necessary to purchase the eight components mentioned in the section dedicated to Implementation. the total cost of the smart thermometer's hardware components totals around 113,00€.

If produced on a commercial level, it is most likely that the parts' price will be different since the hardware components will be bought in a higher number, possibly leading to lower costs for hardware.

E. Smart Thermometer vs. Current Solutions

The HACCP consulting company consulted in this work's scope provided information about some of the thermometer models used the most by its clients in food temperature control procedure. Thus, the smart thermometer prototype (1) was compared to the following solutions: ThermaQ® Blue Thermocouple Alarm (2), Saf-T-Log® Paperless HACCP (3) and Testo 735-2 Digital Multichannel (4).

Through the comparison displayed on Table 1, it is possible to understand that the smart thermometer developed in the scope of this work enhances the capabilities of the solutions currently in the market and tries to establish new ones, with the goal of optimizing food temperature control and record taking procedures.

Table 1 - Comparison between thermometers

Characteristics	Thermometers			
	(1)	(2)	(3)	(4)
Connectivity	Wi-Fi	Bluetooth	Via USB	Via USB
Scheduled Tasks	Yes	No.	No	No
RFID	Yes	No	No	No
Remote tasks update	Yes (web app)	Yes (mobile app)	No	No
Measurement Submissions	Yes (automatic to a database)	Yes (mobile app)	No	No
Cost	112,97€ (Prototype)	179,16 €	162,82 €	586,73 €

Nevertheless, to understand if the prototype is well succeeded, it is important to check if the requirements previously defined are met. Tables 2 and 3 present a comparison of functional and non-functional requirements between the thermometers mentioned and present in the market and the prototype developed.

As can be understood by Table 3 the smart thermometer fulfils all functional requirements defined. Nevertheless, Table 2 shows that the non-functional requirements are not entirely met, namely NFR5 (boot time less than 500ms), which can be bettered with further development.

Table 2 – Comparison of Non-Functional Requirement

Non-functional Requirements	Thermometer			
	ThermaQ® Blue Thermocouple Alarm	Saf-T-Log® Paperless HACCP	Testo 735-2 Digital Multichannel	Smart Thermometer
NFR1	---	---	---	Not enough data
NFR2	✓	✓	✓	✓
NFR3	✓	✓	✓	✓
NFR4	✓	✓	✓	✓
NFR5	Not enough data	Not enough data	Not enough data	✗
NFR6	✓	✓	✓	✓

The tables that aid the comparison of the thermometers show that the solution proposed adds various functionalities to the thermometers currently existent in the marker, such as RFID identification of machines and products and user authentication, allowing to associate relevant information whenever data is submitted.

Table 3 – Comparison of functional requirements

Functional Requirements	Thermometer			
	ThermaQ® Blue Thermocouple Alarm	Saf-T-Log® Paperless HACCP	Testo 735-2 Digital Multichannel	Smart Thermometer
FR01	✓	✓	✓	✓
FR02	✓	✓	✓	✓
FR03	✗	✗	✗	✓
FR04	✗	+/-	+/-	✓
FR05	✗	✗	✗	✓
FR06	✗	✗	✗	✓
FR07	✗	✗	✗	✓
FR08	✗	✗	✗	✓
FR09	+/-	✗	✗	✓
FR10	✓	✓	✓	✓
FR11	✗	✗	✗	✓
FR12	✗	+/-	+/-	✓
FR13	✗	+/-	+/-	✓
FR14	✗	✓	✗	✓
FR15	✗	✗	✗	✓
FR16	✗	✓	✓	✓
FR17	✗	✗	✗	✓
FR18	Not enough data	Not enough data	Not enough data	✓

Thus, it is possible to conclude that the prototype developed meets the requirements established and can serve the needs related to food safety of different establishments taking part in the food supply chain.

VII. CONCLUSIONS

When properly implemented, FSMS significantly contribute to the improvement of food safety practices, positively impacting public health and establishments economics. However, FSMS are often poorly implemented and far from perfect, and when it happens, instead of contributing to help solve the problem, the contrary occurs.

Foods are sensitive to time and temperature, with temperature being a crucial factor when regarding food safety. Temperature abuse is, therefore, often appointed as the cause of FBDO with origin in the whole food supply chain. Hence the importance of optimizing temperature control and record taking processes is significant.

The feedback from a HACCP consulting company responsible for implementing digital FSMS in more than 400 establishments contributed to identifying and confirming the most significant issues that companies have regarding temperature control procedures. The feedback was useful for understanding temperature control procedures implemented across different types of food-related establishments. It also allowed to reflect on improvements that would benefit the establishment and FBD prevention.

Thus, this dissertation's goal was to design a handheld thermometer for temperature measurements and record taking that will be used to control temperatures more efficiently and reliably, enabling greater control over the measurements that HACCP and FSMS require. This goal was achieved by developing a solution based on a smart thermometer that integrates new and improved features, such as allowing user authentication, machine and product RFID identification and association with the temperature measurement, presentation of daily tasks, and automatic remote submission to a database.

To develop the solution and before implementing the system, the main features and requirements were defined. These included the essential features, necessary external interfaces, performance, safety and security requirements, and user classes and characteristics. The system architecture was also defined. This allowed further understanding of how the system would interact with the users and other external systems, such as existing HACCP, or traceability systems. It also detailed the necessary supporting services that include a remote database, an admin web application, and an API for the smart thermometer and external systems to consume.

Finally, the system was implemented and tested through a practical experiment and compared with a reference thermometer, proving that it is able be perform its function properly as a normal thermometer while integrating all the features that make it a smart tool.

An additional survey was conducted to four food safety managers that represent different parts of the food supply chain and combined are responsible for more than 700 establishments. The survey allowed to verify that the problems described along this dissertation's work are, in fact, present in their daily operations. The interest showed in the proposed solution was big, and it further confirmed that food safety managers find value in a smart thermometer that presents to the staff what measurements they shall take during the day identifies who is taking them, when, where, and what should be measured, automatically uploads the measurements to a remote database with all the corresponding information. It was considered that it has great potential to improve food safety operations.

This smart thermometer prototype proves that adding a solution that brings new and improved features to the market can ease temperature control and record taking tasks in a wide range of food establishments. Features such as the automatic submission of results to a database that is easy to access and verify older records are not present in thermometers now available and are of great interest to commercial establishments and the regular consumer, which is increasingly more alert and concerned about food safety issues.

Since the work here presented is a prototype and since the idea has gathered a lot of attention and has been proven that there is a great interest in a solution like the one proposed, future versions of the product shall be considered. Now that the base features and hardware are established it will be easier to further develop the idea.

ACKNOWLEDGMENTS

Thank you to the HACCP company who supported and provided feedback all throughout this work.

REFERENCES

[1] 'WHO | WHO estimates of the global burden of foodborne diseases', *WHO*. http://www.who.int/foodsafety/publications/foodborne_disease/fergreport/en/ (accessed Nov. 16, 2020).

[2] S. Jaffee, S. Henson, L. Unnevehr, D. Grace, and E. Cassou, 'The Safe Food Imperative: Accelerating Progress in Low-And Middleincome Countries', p. 211.

[3] C. C. Correia, I. C. Cunha, A. Coelho, C. Maia, and C. Pena, '_Food-borne disease outbreaks laboratory investigation: data from 2017', *Bol. Epidemiológico*, vol. 25, no. 1, p. 52, 2019.

[4] F. L. Bryan, J. J. Guzewich, and E. C. D. Todd, 'Surveillance of Foodborne Disease III. Summary and Presentation of Data on Vehicles and Contributory Factors; Their Value and Limitations', *J. Food Prot.*, vol. 60, no. 6, pp. 701–714, Jun. 1997, doi: 10.4315/0362-028X-60.6.701.

[5] S. S. Khandke and T. Mayes, 'HACCP implementation: a practical guide to the implementation of the HACCP plan', *Food Control*, vol. 9, no. 2, pp. 103–109, Apr. 1998, doi: 10.1016/S0956-7135(97)00065-0.

[6] F. Liu, H. Rhim, K. Park, J. Xu, and C. K. Y. Lo, 'HACCP certification in food industry: Trade-offs in product safety and firm performance', *Int. J. Prod. Econ.*, vol. 231, p. 107838, Jan. 2021, doi: 10.1016/j.ijpe.2020.107838.

[7] Codex Alimentarius Commission and Joint FAO WHO Food Standards Programme, Eds., *Food hygiene: basic texts ; Codex Alimentarius*, 4. ed. Rome: FAO [u.a.], 2009.

[8] 'Commission Notice on the implementation of food safety management systems covering prerequisite programs (PRPs) and procedures based on the HACCP principles, including the facilitation/flexibility of the implementation in certain food businesses', p. 32.

[9] P. M, F. Ag, P. N, and L. I, 'Implementation of the Hazard Analysis Critical Control Point (HACCP) System to a Dairy Industry: Evaluation of Benefits and Barriers', *J. Food Nutr. Diet.*, vol. 01, no. 01, Jan. 2016, doi: 10.19104/jfnd.2015.102.

[10] C. A. Wallace, W. H. Sperber, and S. E. Mortimore, *Food Safety for the 21st Century: Managing HACCP and Food Safety Throughout the Global Supply Chain*. John Wiley & Sons, 2018.

[11] S. Mortimore and C. Wallace, *HACCP: A Practical Approach*. Springer Science & Business Media, 2013.

[12] 'ISO 22005:2007(en), Traceability in the feed and food chain — General principles and basic requirements for system design and implementation'. https://www.iso.org/obp/ui/#iso:std:iso:22005:ed-1:v1:en:biblref:3 (accessed Dec. 14, 2020).

[13] M. M. Aung and Y. S. Chang, 'Traceability in a food supply chain: Safety and quality perspectives', *Food Control*, vol. 39, pp. 172–184, May 2014, doi: 10.1016/j.foodcont.2013.11.007.

[14] European Commission and Directorate-General for Health and Consumer Protection, *Food traceability*. 2007.

[15] *Thermometer Use in Retail Foodservice Establishments -- What Managers Need to Know*. .

[16] L. G. Brown *et al.*, 'Restaurant Food Cooling Practices†', *J. Food Prot.*, vol. 75, no. 12, pp. 2172–2178, Dec. 2012, doi: 10.4315/0362-028X.JFP-12-256.

[17] R. Badia-Melis, L. Ruiz-Garcia, J. Hierro, and J. Villalba, 'Refrigerated Fruit Storage Monitoring Combining Two Different Wireless Sensing Technologies: RFID and WSN', *Sensors*, vol. 15, pp. 4781–4795, Mar. 2015, doi: 10.3390/s150304781.

[18] M. M. Aung and Y. S. Chang, 'Temperature management for the quality assurance of a perishable food supply chain', *Food Control*, vol. 40, pp. 198–207, Jun. 2014, doi: 10.1016/j.foodcont.2013.11.016.

[19] A. Pal and K. Kant, 'Using Blockchain for Provenance and Traceability in Internet of Things-Integrated Food Logistics', *Computer*, vol. 52, no. 12, pp. 94–98, Dec. 2019, doi: 10.1109/MC.2019.2942111.

[20] X. Xiao, Q. He, Z. Fu, M. Xu, and X. Zhang, 'Applying CS and WSN methods for improving efficiency of frozen and chilled aquatic products monitoring system in cold chain logistics', *Food Control*, vol. 60, pp. 656–666, Feb. 2016, doi: 10.1016/j.foodcont.2015.09.012.

[21] 'M5Stack - Modular Rapid ESP32 IoT Development Board - ESP32 dev kits', *m5stack-store*. https://m5stack.com/ (accessed Dec. 27, 2020).

[22] 'ESP32 Wi-Fi & Bluetooth MCU I Espressif Systems'. https://www.espressif.com/en/products/socs/esp32 (accessed Dec. 27, 2020).

[23] 'MAX31856.pdf'. Accessed: Dec. 27, 2020. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX31856.pdf.

[24] 'Digital plug & play infrared thermometer in a TO-can', *Melexis*. https://www.melexis.com/en/product/MLX90614/Digital-Plug-Play-Infrared-Thermometer-TO-Can (accessed Dec. 27, 2020).

[25] L. LLC, 'LVGL - Light and Versatile Embedded Graphics Library', *LVGL*. https://lvgl.io/ (accessed Dec. 30, 2020).

[26] 'ThermaQ Blue Dual Channel Thermocouple|ThermoWorks'. https://www.thermoworks.com/ThermaQ-Blue (accessed Dec. 29, 2020).

[27] 'Creating a Properly Made Icebath'. https://www.thermoworks.com/thermapen101_creating_an_icebath (accessed Dec. 31, 2020).