Embedded system for calorie consumption estimation in gyms based on ESP32

Luis Eduardo Brito Ferreira Siebra[2], João Marcelo Xavier Natário Teixeira[1,2]
luis.siebra@ufpe.br,joao.teixe@ufpe.br
[1]Escola de Artes e Humanidades da Universidade de São Paulo (EACH - USP)
[2]Universidade Federal de Pernambuco
Recife, Pernambuco, Brazil

Figure 1: Proposed project being tested by two users and their corresponding measurements.

Abstract

In Brazil, as in many parts of the world, as health and well-being become increasing priorities, the search for ways to exercise, such as joining gyms, has increased. In the case of small and medium-sized gyms, it is crucial to implement new technologies to ensure student engagement and provide a more efficient training experience. Therefore, as a way to help modernize this gym sector, this work presents a software and hardware solution that aims to democratize access to health monitoring and management tools, through a device that tracks student performance in real time and provides calorie estimates and other data as instant feedback for an app-based platform. By offering this new experience, gyms are modernizing and guaranteeing users the possibility of tracking their progress, which can generate greater motivation and personal satisfaction. The designed embedded system consists of a printed circuit board equipped with sensors, fixed to the exercise machine and is fed in real time from a database, also accessed by the created application, allowing all data captured by the device to be available in real time in the palm of the student's hand for control and consultation.

Keywords

Embedded systems, caloric estimation, database

How to cite this paper:
Luis Eduardo Brito Ferreira Siebra[2], João Marcelo Xavier Natário Teixeira[1,2]. 2025. Embedded system for calorie consumption estimation in gyms based on ESP32. In *Proceedings of ACM IMX Workshops, June 3 - 6, 2025.* SBC, Porto Alegre/RS, Brazil, 6 pages. https://doi.org/10.5753/imxw.2025.1957

1 Introduction

In recent years, the search for gyms has intensified [10], reflecting a growing concern for health and physical well-being. This trend is driven by a variety of factors, including increased awareness of the benefits of physical activity for overall health [9], heightened worries about sedentary lifestyles, and the pursuit of a more active and healthier way of life [5]. However, this trend is not uniform, and a noticeable disparity arises between larger, often better-equipped chain gyms and neighborhood gyms, which generally operate on a smaller scale with limited resources [7].

This gap between chain gyms and neighborhood gyms can have significant public health implications. While larger gyms may offer a wide range of equipment and training programs, as well as access to qualified professionals, neighborhood gyms often face major challenges in meeting their clients' needs, such as limited space, constrained financial resources, and a lack of modern equipment.

The goal of this work is to develop a device connected to a virtual database that provides real-time information on exercises performed by users on older gym machines, with the aim of modernizing the equipment, maintaining public interest, and offering an enhanced experience. Through an embedded system, workout data are captured and made available on a web platform accessible via smartphone, enabling users to monitor their performance and adjust their workouts as needed, thereby promoting a healthier and more active lifestyle.

2 Related Work

With the spread of courses and development kits for microcontrollers, such as the ESP32, ESP8266, and even Arduino, there has been a significant increase in the number of enthusiasts and hobbyists in the field of the Internet of Things (IoT) [6]. This field has gained prominence especially due to the open-source nature of many of these projects, which encourages developers to continue working despite failures or challenges. Specifically, in the case of the ESP32, the existence of an active community in technical forums remains a valuable source of support and knowledge sharing to this day. Below are some projects that use technologies similar to those employed in this work but with different approaches and objectives.

In Mohamed Ibrahim's project [4], a prototype very similar to the one proposed in this work was presented. It featured a drone controller and an analyzer of drone stability data, where tools such as real-time data analysis and the use of the MPU6050 sensor are similar to this project. The project also used JSON-format files for data packets and the ESP32's own Wi-Fi to power a real-time control server built using HIVEMQ, a free public MQTT broker.

Another similar application uses an ESP32 in conjunction with a DHT11 temperature sensor [8] to analyze ambient temperature and send the data to Firebase, using an Arduino library responsible for data manipulation and transmission. This method provides a practical way to feed real-time data into the database with the temperature and humidity values measured by the DHT11 sensor.

The above-mentioned projects inspire and guide the implementation of the model presented here. The method for reading data from the MPU6050 follows a logic similar to that used by the drone controller to handle such data. As for communication with Firebase, the aforementioned project served as a basis for handling the JSON files in the main project, helping to standardize and organize data manipulation.

3 Methodology

A distinguishing factor for a gym to attract new members or even retain current ones is the modernization of its equipment, which is precisely the challenge that this proposal aims to address (particularly for many smaller gyms). Conventional machines typically count repetitions and exercise duration; however, to enhance user performance, providing data such as approximate calorie expenditure and the total duration of the workout is essential.

Smartwatches are an example of devices used to track workout effort, as they provide data on workout duration, heart rate monitoring, total calorie expenditure, among other metrics. However, because they are personal devices, as shown in Figure 1, their information is restricted solely to their users.

Treadmills offer a relatively accurate estimate of aerobic exercise metrics, but their data are not stored for future reference. Thus, an opportunity arises to develop a device capable of similar monitoring for anaerobic exercises, with the ability to store data in the cloud. Such a device would allow users to access these data at any time, enabling them to track their progress or identify potential shortcomings, for instance.

The requirements were defined primarily with the goal of creating an intelligent embedded system that could provide users with useful data to enhance their gym experience, while also offering good durability:

- Remote input of variables
- Definition of the repetition estimate
- Calorie expenditure estimate
- Local data debugging
- Calorie expenditure estimate
- Smallest possible project size
- Resistance and durability
- Low material cost

The proposed solution involves creating an embedded system that will operate in conjunction with a gym machine, capable of communicating with the user through an application and using a database as an intermediary. That is, the device sends the cloud data related to the exercise performed by the user and retrieves from the cloud the data the user has entered into the application.

Initially, the device waits for data input from the user, specifically information needed for its operation such as the user's name (used for registration and for creating a record in the database where exercise results will be saved) along with age, weight, and the type of exercise to be performed, which are necessary to calculate the calorie estimate.

The device features a local 0.96-inch OLED display to confirm whether the data sent from the application were indeed received correctly, thus allowing the device to begin its operation.

The embedded system itself was designed with a focus on energy efficiency. The components used (sensors and microcontrollers) will be detailed later, but the overall structure relies on the I2C protocol, which is widely used for serial communication.

Development of the embedded system emphasized rapid sensor response and efficient microcontroller processing, ensuring the project's low power consumption and compact design requirements could be met. These capabilities were implemented by adopting the I2C protocol for communication among the microcontroller, sensors, and display. I2C is a serial communication protocol that operates through a shared bus among all peripherals, using only two pins: SDA (responsible for the data line) and SCL (which controls the communication clock). The ESP32-C3 microcontroller manages all aspects of this communication, supported by the Adafruit libraries [1] [2] .

Initial tests were planned for a laboratory environment to validate and integrate the entire system and its cloud communication. Subsequently, the plan is to conduct tests in a gym, using three different machines and at least two people of varying ages, genders, and weights to explore different exercise execution patterns and other potential factors.

4 Proposed Solution

As previously mentioned, the solution proposed in this work involves creating a printed circuit board (PCB) with an embedded system (see Figure 2 and Figure 3), responsible for estimating the calories burned during physical exercises based on duration, the type of exercise performed, and the number of repetitions.

The firmware used in this project was developed with the help of the PlatformIO extension for Visual Studio Code. Since it is an embedded system using a microcontroller from the ESP32 family, the selected programming language was C.

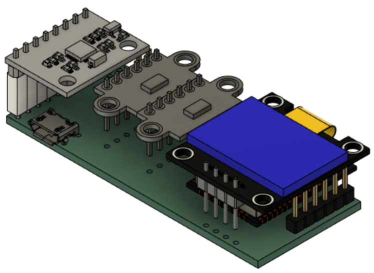

Figure 2: PCB of the designed system (3D view).

Figure 3: Complete system (3D view).

Some of the key data for estimating calories burned during gym exercises vary from one user to another, such as the user's weight, age, and the type of exercise being performed. All of this information is supplied through an application created with the MIT App Inventor platform [1]. This application can update the database, which is in turn accessed by the embedded system in real time. The database stores the data for the most recently registered user, making it possible to generate accurate estimates.

From the previously discussed list of requirements, the project must be as compact as possible while maintaining robustness and ease of use, as shown in Figure 4. ESP32C3S was selected due to its versatility in choosing which pins will be used for I2C communication. Additionally, it has an internal USB-JTAG Serial converter integrated into the chip, requiring only the traces of pins 19 and 20 of the ESP32C3S to connect to a female USB port with the power pins present. This provides a platform to program the PCB in question without needing an external TTL converter or similar device, which would be necessary if the UART were used for programming.

[1]https://appinventor.mit.edu/

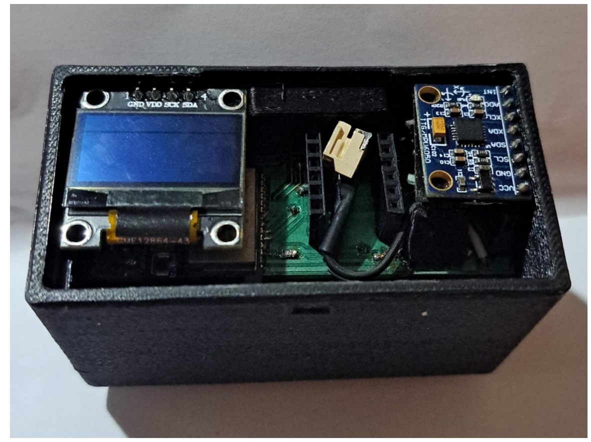

Figure 4: Printed circuit board with sensors mounted in a 3D-printed case.

The MPU6050 accelerometer was chosen to detect motion. The repetition count is managed by code that monitors acceleration variations across the three axes and uses a safety margin to determine when a repetition is completed. Additionally, the total duration of the exercise is calculated from the start until 10 seconds after the last repetition, ensuring accurate detection of when the activity ends.

The complete hardware list used in the proposed solution is shown as follows:

- ESP32C3S module
- MPU6050 sensor
- SSD1306 OLED display
- 2 VL53L0X sensors
- 2 SMD 0603 10μF capacitors
- 1 SMD 0603 10KΩ resistor
- 1 AMS1117 3V3 regulator
- 1 18650 lithium battery
- 1 female micro USB connector

All of the embedded system circuit design was carried out using the open source software KiCad. KiCad offers several plugins to optimize its use and has a very active online community discussing bugs and improvements. The circuit is shown in the diagram in Figure 5.

Explaining the electrical schematic in detail, it can be observed in the circuit in Figure 6 that pins 0 and 1 of the ESP32C3S are used for I2C communication, in addition to having well-defined UART and JTAG pins. JTAG was used as the main mechanism for programming and debugging; however, there are also the six necessary pins for programming the ESP32 via serial using an appropriate TTL converter if needed.

The schematics of the sensors, shown in Figures 7, 8, and 9, basically consist of the power supply and the SDA and SCL pins shared among all. Initially, the PCB was designed considering the use of two VL53L0X sensors for configuration and interaction between the user and the device; however, with the introduction of an application that will be discussed later, the sensors were removed

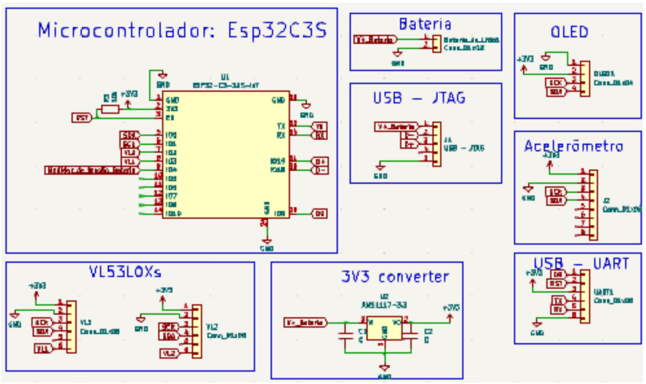

Figure 5: Schematic of the project developed in KiCad.

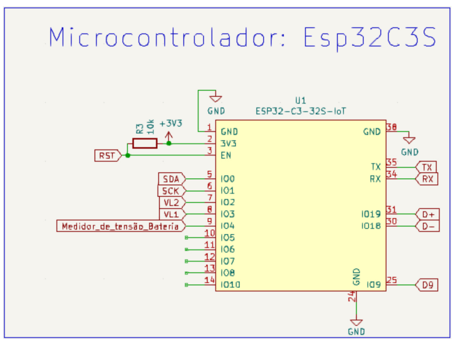

Figure 6: Pinout of the ESP32C3S.

from the final test circuit, although they are still present on the assembled PCB.

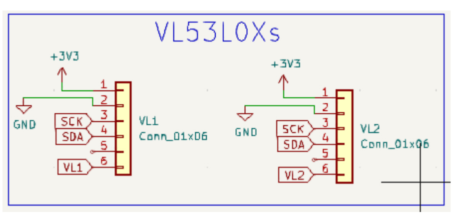

Figure 7: VL53L0X circuit.

For the programming interface, a bus was provided containing the necessary pins for programming the ESP32C3S via TTL, as shown in Figure 11. That is, the pins for enable, DTR, 3v3, GND, RX0, and TX0. In addition, another interface was used as the primary one for programming and debugging the project, previously mentioned, which consists of the USB JTAG represented in Figure 10. In this interface, the D+ and D- pins (pins 19 and 18, respectively) are connected to the USB port of the board, along with the overall circuit GND and the power pin of the 3v3 regulator, which in this case is a shared pin for the USB 5V and the battery V+. The voltage regulator circuit used in the project is highlighted in Figure 12.

For the calorie expenditure estimate, the MET method was chosen (MET stands for Metabolic Equivalent of Task [3]). This method

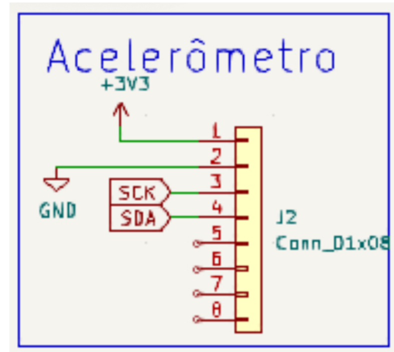

Figure 8: MPU6050 circuit.

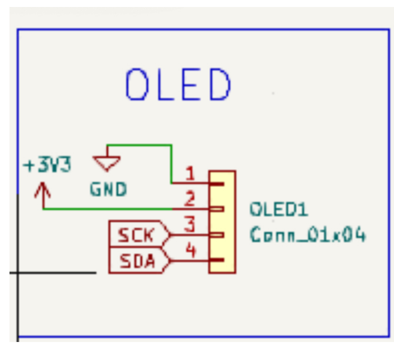

Figure 9: SSD1306 OLED circuit.

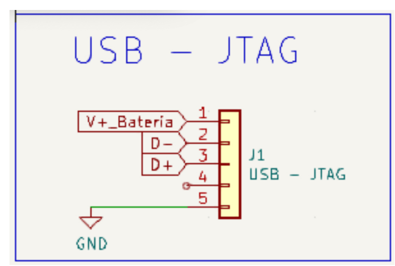

Figure 10: Circuit of the female micro USB available in the project for programming, power supply, and debugging.

is commonly used to estimate calorie burn based on a standardized unit known as a MET. One MET is defined as the oxygen consumption for adults corresponding to 3.5 ml of oxygen per kilogram of body weight per minute. The standard MET values used in this project were estimated as shown in Table 1.

Table 1: Estimated METs

Type of Exercise	METs
Light exercise	3
Moderate strength exercise	3.5
Intense strength exercise	6
High-intensity, short-interval exercise	8

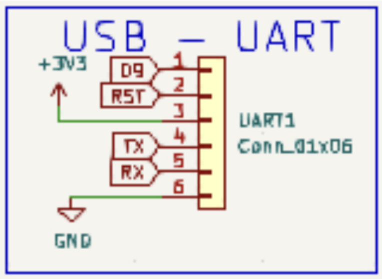

Figure 11: UART0 circuit for programming via an external serial converter.

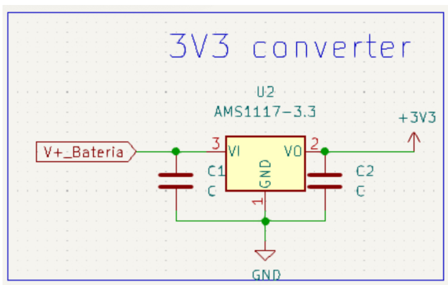

Figure 12: Voltage regulator circuit for 3.3V.

Figure 13: Initial screen of the developed application.

An application was developed to establish the connection between the user and the device, using the App Inventor platform. The app consists of a single screen divided into three sections, as shown in Figure 13. At the top, there are four text boxes and three buttons that allow the user to input their data (weight, age, name, and the type of exercise to be performed). In this context, the input for the exercise type must be 1, 2, or 3, depending on whether the exercise is classified as light, intense, or strength-based. The three buttons in the top section have the following functions: the first adds or modifies user data, the second deletes a previously selected user, and the third clears the text boxes.

In the middle section of the screen, there is a text box and a button for searching and selecting a user. Once the name of a registered user is entered into this text box and the search button is pressed, the text boxes at the top are filled with the pre-registered data retrieved from the database. This middle section also contains a vertical menu listing all previously registered users. If one of them is selected, it automatically fills the four text boxes at the top with that user's data, just as the search function does.

In the bottom section of the screen, there are three text boxes displaying the total duration of the exercise, the number of repetitions performed, and the exercise's estimated calorie expenditure.

For the budget estimate, it is possible to observe three columns in Table 2. The first column lists the item for which the cost is estimated, and the second column indicates the purchase location. For purchases made in Brazil, only the item price and shipping costs were considered together in the third column. However, for purchases outside Brazil, the price includes the sum of the item cost, shipping, and the estimated import tax.

When choosing components from outside Brazil, project parts were selected based on the "choice" free-shipping package. This promotion applies to specific components labeled with that tag; if the combined cost of these items exceeds 90 BRL, shipping for the entire set is free. The total cost of the selected components did reach this 90 BRL threshold. Nonetheless, in the budget shown in Table 2, the individual price of each item is listed. Note that some items are sold only in packs of 20 units, as is the case with the AMS1117 voltage regulator.

A cloud consumption plan capable of handling 20 million messages per month was considered, which is a generous margin and allows up to 100 simultaneous connections. If more than that is needed, the cost becomes 5 USD. The cost estimate for the 3D case was based on the material consumed and machine-hour usage from the 3D printing service provider.

5 Validation

Tests were conducted with three different individuals: two males and one female. After signing a consent form for image usage, each participant was given an explanation of how the app and the device worked. While the device was being installed on the selected machine, the participant filled in their personal information on the app's home screen, allowing for registration and the initial configuration of the embedded device.

User 1 was a 27-year-old civil engineer whose workout is high-intensity; he typically uses heavy weights, as shown in Figure 59. Accordingly, when registering, he selected "2" in the text box referring to workout intensity.

Table 2: Estimated prices for the components described in the solution.

Item	Purchase Location	Price (BRL)
Cloud consumption	Brazil (Firebase)	28.00
3D prototyping	Brazil (TechFab)	15.00
PCB prototyping	China (JLCPCB)	11.32
Esp32C3S	China (AliExpress)	25.96
OLED SSD1306 Display	China (AliExpress)	18.01
MPU6050 Sensor	China (AliExpress)	7.30
VL53L0X Sensor	China (AliExpress)	12.76
18650 Li-ion Battery	Brazil (Mercado Livre)	38.94
1 AMS1117 3V3 Regulator	China (AliExpress)	1.67
2 Capacitors 10mFaraday 0603	China (AliExpress)	0.22
1 Resistor 10kOhms 0603	China (AliExpress)	0.16
1 Female Micro USB Port	China (AliExpress)	0.75
1 Li-ion Battery Charging Circuit	China (AliExpress)	1.67
Total		**161.76**

There was a difference in the total exercise duration, the estimated calorie expenditure, and the number of repetitions. The first two differences were expected because the device also counted rest time and estimates calorie expenditure during rest, uploading those values to the cloud. The data calculated by the embedded system were successfully updated in real time during the exercise and at its conclusion.

In Figure 1, the OLED display shows 10 repetitions, which are reset as soon as the series duration resets. Consequently, there is a discrepancy between the local data and the data seen in the database when accessed.

The second user was a 60-year-old dentist whose workout is low-intensity; she typically uses light weights. Therefore, during registration, she selected “1” in the text box referring to workout intensity. After completing her registration, she began her exercise.

The third user to test the device was a 30-year-old physical education professional with a normally high-intensity routine. However, he was not feeling at his best and decided to use a light weight, as shown in Figure 1. Therefore, when registering, he selected “1” in the text box referring to workout intensity.

The same expected data discrepancy observed for the previous user was found between the data for users 2 and 3. This will be discussed in more detail in the future improvements section. However, based on observations after the exercise ended, the system correctly counted the repetitions, as well as the expected duration and calorie estimate.

6 Conclusion

In this work, an embedded system was developed and implemented in both hardware and software through the design and manufacture of a battery-powered printed circuit board. Its goal is to read and write data to a database in real time. These data pertain to the estimated calories burned and the quality of the physical exercise performed. The system was applied in gyms to democratize health monitoring and management tools. Additionally, an application was developed to communicate in real time with the device installed on exercise machines, using data transmitted through the database. Through this application, it is possible to set up a user profile and track exercise-related data.

One of the challenges encountered during the development of this project involved defining the exercise pattern most commonly performed by gym users in order to ensure a reliable method of detecting repetitions. This was due to the wide variety in workout styles—for instance, some people perform rapid repetitions, while others hold a peak contraction for two seconds. The system had to adapt to these differences to accurately count repetitions and avoid data loss, which could result in an error and consequently a false negative.

The project was designed with potential future upgrades in mind, both in terms of hardware and software. Thus, for future versions, some possible improvements include:

- Enhancing the application by adding more screens and user registration with email and password.
- Improving the embedded firmware by adding a local database.
- Using other Firebase platform tools to improve project performance, possibly even hosting the application on Firebase.
- Adding a compact, efficient, and safe battery charging system to maintain secure and streamlined usage.

References

[1] 2023. Adafruit MPU6050 Library. https://github.com/adafruit/Adafruit_MPU6050. Accessed: 2025-03-23.

[2] 2023. Adafruit Unified Sensor Library. https://github.com/adafruit/Adafruit_Sensor. Accessed: 2025-03-23.

[3] Barbara A Bushman. 2023. Metabolic Calculations Cases. *ACSM’s Health & Fitness Journal* 27, 2 (2023), 6–10.

[4] Mohamed Ibrahim. 2023. Drone_ESP32. https://github.com/mohamed-ibrahim-21/Drone_ESP32. Accessed: 2025-03-23.

[5] Neville Owen, Geneviève N Healy, Charles E Matthews, and David W Dunstan. 2010. Too much sitting: the population health science of sedentary behavior. *Exercise and sport sciences reviews* 38, 3 (2010), 105–113.

[6] Karen Rose, Scott Eldridge, Lyman Chapin, et al. 2015. The internet of things: An overview. *The internet society (ISOC)* 80, 15 (2015), 1–53.

[7] James F Sallis, Myron F Floyd, Daniel A Rodríguez, and Brian E Saelens. 2012. Role of built environments in physical activity, obesity, and cardiovascular disease. *Circulation* 125, 5 (2012), 729–737.

[8] Sergioeabarcaf. 2023. ESP32-DHT11-Firebase. https://github.com/Sergioeabarcaf/ESP32-DHT11-Firebase. Accessed: 2025-03-23.

[9] U.S. Department of Health and Human Services. 2018. *Physical Activity Guidelines for Americans.* U.S. Government Printing Office, Washington, DC. https://health.gov/sites/default/files/2019-09/Physical_Activity_Guidelines_2nd_edition.pdf

[10] World Health Organization. 2010. *Global Recommendations on Physical Activity for Health.* World Health Organization, Geneva, Switzerland. https://www.who.int/dietphysicalactivity/publications/9789241599979/en/