Smart Coaster: Project and Implementation of an IoT Device for Monitoring Beverage Consumption

Eduardo Alexandre Pereira da Silva Júnior
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
Recife, Brazil
eduardo.silvajunior@ufpe.br

Andrea Maria Nogueira Cavalcanti Ribeiro
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
Recife, Brasil
andrea.marianogueira@ufpe.br

Aline Victória Cavalcanti Pereira
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
Recife, Brasil
aline.cavalcantipereira@ufpe.br

João Marcelo Teixeira
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
Recife, Brasil
joao.teixe@ufpe.br

***Abstract*—The Internet of Things (IoT) technology has become increasingly prevalent in our daily lives, enabling devices to connect and communicate for data collection and remote analysis. In this context, this present paper aims to present the development of a smart coaster that utilizes IoT to monitor liquid consumption and display the data in a dedicated application. This project, seeks to demonstrate how IoT can be applied to promote healthy habits and offer practical and efficient solutions for daily challenges. It is noteworthy that data collection on liquid consumption can provide valuable information about daily hydration, enabling users to adjust their routine to achieve an adequate balance. Thus, the smart coaster project presents an innovative solution for monitoring liquid consumption and highlights the potential of IoT to provide practical and efficient responses to the demands of modern life.**

***Index Terms*—Internet of Things, smart coaster, monitoring, product development, healthy habits.**

I. Introduction

The growing need to improve daily quality of life, especially in a context where remote classes and home office work have become increasingly common. The development of technologies that can assist in maintaining daily health care is crutial. In particular, for those who spend most of the day at their desks, it is essential to find solutions that facilitate proper monitoring of their habits, especially those that can directly affect their health.

Access to potable water is vital. However, due to busy schedules, remembering to drink enough water is becoming increasingly difficult. To develop a healthy habit of staying hydrated, it's essential to consume adequate high-quality water regularly. This is where an intelligent devices powered by IoT can be beneficial [1].

This paper aims to provide a practical and efficient solution for controlling liquid consumption through the development of a smart coaster. By using an IoT device connected to an application, users will have real-time access to information about the amount of liquid consumed, enabling the adoption of healthy hydration habits.

Furthermore, this work contributes to the field of the Internet of Things by demonstrating a practical and functional application that integrates physical devices and digital platforms [1]. By exploring available resources, such as analog sensors, and the communication capabilities of the ESP32, this project showcases the potential of IoT in the field of health and well-being.

II. Related work

In the vast world of connected devices, we find a wide range of solutions that cover various aspects of our daily lives. In the context of monitoring water intake, several approaches have been developed and tested [1]–[4].A review on intelligent water bottle powered by IoT where a comparison between various smart bottles was presented. Those bottles were introduced to enable effective management of daily water requirements [1].

The smart bottle developed by [2] focus not only in the water consumption but also the quality of the water. Additionally, a mobile app was developed to track the location of water sources and monitor water consumption using recorded sensor data. The app also functions as the communication interface between the bottle and the user. The hydration system presented in [3] uses a water float sensor to detect water level. It not only helps the users to measure the level of water, but also indicates the human to drink water when it's time to top up their body. Another notable example is a smart bottle that uses an ultrasonic sensor to track water consumption [4]. This solution employs the same principle of network-connected objects, similar to what was used in this paper.

The methodology employed in this paper is based on measuring liquid intake through the variation in the relative

[1]https://github.com/eduardoalexandree/Porta_Copos

weight of the water. The liquid level was monitored in real-time by incorporating pressure sensors into the surface of the coaster to detect the weight and pressure. On the other hand, the smart bottle projects presented by [2]–[4] measure water intake using a sensor to determine the water level inside the container. Both approaches make relative comparisons, one based on the weight of the liquid and the other on the height of the water column inside the container.

This project adopts the concept of monitoring water intake independently of the user's container. This is due to the capability of the smart coaster to passively capture data, requiring only that the container be placed on the coaster whenever the user wishes to measure liquid consumption. On the other hand, in the smart bottle projects, the device is incorporated into the bottle itself, preventing the user from measuring consumption if they do not have the bottle with them or if they are using a different type of container.

However, the smart bottle projects offer greater portability compared to the smart coaster, as it has its own battery, allowing water consumption to be measured in outdoor environments such as gyms and walks. Both projects have their positives and negatives, and both approaches enable liquid consumption measurement in different ways. The choice between them depends on the user's individual needs and preferences.

III. The Proposed Solution

This project was developed using a passive way to measure the volume of liquids without requiring the user to take any additional actions other than placing the cup on the coaster. The solution was achieved by measuring the mass of the container to determine an approximate estimate of the volume consumed. The formula used to calculate the volume is the relationship between density, mass, and volume, represented by the equation 1:

$$\rho = \frac{m}{V}, \tag{1}$$

where ρ is the density of the liquid, m is the mass, and V is the volume.

Every liquid has a specific density, Table I presents the densities long with their respective sources:

TABLE I
Density Table

Liquid	Density (kg/m³)	Reference
Water (25°C)	997	[5]
Coffee	± 1190	[6]
Coca Cola	1026	[7]
Coca Cola Zero	911	[7]

After calculating the volume using the equation 1, the system will check if the volume value has decreased, and the values will be stored in the ESP32's memory and sent via Bluetooth connection to an application. This application will display the consumption data of the selected liquids, allowing the user to track and monitor their intake.

Measuring the weight of the container requires a component capable of converting a physical force into a digital signal that can be read by the microcontroller's GPIOs. This component is known as a load cell, a device that converts the applied force into an electrical signal, which can be translated into an approximate mass measurement.

The HX711 load cell amplifier is a device used for reading load cells, allowing the measurement of weights and forces. It consists of a 24-bit analog-to-digital converter and a high-precision instrumentation amplifier. It operates after the Wheatstone bridge, where the load cell is connected in a bridge configuration along with compensation resistors. When a force is applied to the load cell, a bridge imbalance occurs, resulting in a variation in the output voltage. For the HX711 to work correctly, the load cell must have an integrated Wheatstone bridge in its circuit.

For project prototyping and testing, the ESP32 WROOM DEV KIT V1 board was used, which has 36 GPIOs and an integrated antenna in the circuit. To program the microcontroller, a USB Type-C cable is used, which utilizes UART serial communication for code compilation and integration into the microcontroller. The programming language used is a simplified variant of C++ with a specific Arduino Application Programming Interface (API). Although the language is similar to C++, there are some differences and restrictions specific to the Arduino environment.

For the prototype circuit assembly, jumpers were used to connect the HX711 amplifier outputs to the ESP32 Wroom Devkit V1 development board. The load cell outputs were soldered to the HX711 amplifier inputs, as shown in the circuit in Figure 1. The load cell was manufactured in the 3D printed case, while the other components were placed inside the coaster.

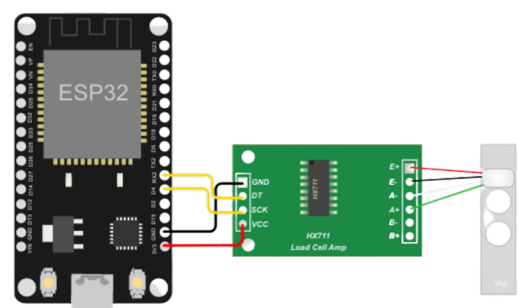

Fig. 1. Circuit connection diagram of the project. Source: https://wokwi.com/

In the prototyping stage, the load cell was manufactured using PLA (Polylactic Acid). The HX711 amplifier was also positioned inside the case. To facilitate testing, the ESP32 was kept outside the case. Figure 2 illustrates the first prototype of the coaster.

A. Code

The coaster code was written in the C++ programming language, using the Arduino IDE application as the programming interface and serial port viewer. The Arduino IDE is an application developed by the Arduino team that allows for

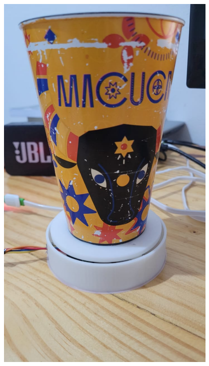

Fig. 2. First test prototype of the coaster.

coding and sending code to the device via the USB serial port. It is also compatible with many other microcontrollers that use the same programming language [8].

For the project, the libraries HX711.h [9], SPIFFS.h [10], and BluetoothSerial.h [11] were used. These libraries provide additional functionalities and facilitate communication with the components used in the project. The presented code snippet is used to calibrate the load cell. The pins to which the DT and SCK outputs of the HX711 module are connected are defined. In the project, these outputs are connected to pins 16 and 4, respectively.

1) Calibration.ino: The code contained in the "Calibracao.h" file plays a crucial role in the initial calibration of the coaster's data. The function ‘$scale.get_{v}alue()$‘ is responsible for calculating the average value of 10 readings taken by the HX711 module, which is connected to the scale. This value is displayed on the Arduino IDE serial port and represents the raw reading made by the scale.

To perform the calibration, a known weight is placed on the scale, and the raw value measured by the ESP32 ports is recorded. This raw value is then divided by the known weight, generating a calibration factor. This factor will be applied in the main code to convert the raw scale readings into accurate weight values.

2) Coaster.ino: The code contained in "Coaster.ino" is responsible for the overall operation of the smart coaster system. It covers data capture from the scale in the coaster, as well as bidirectional data communication via Bluetooth.

A critical part of developing this code was ensuring data stability once a container is placed on the coaster. This stability is achieved by verifying the arrays that store the received data. The system evaluates the difference between two consecutive data points stored in different array positions. When this difference becomes minimal, the value is considered stable and is used as the correct reading.

The code also includes functions that respond to specific commands from the application. For example, when a message in the format ‘["function": "liquid select", "value": "water"]‘ is received, the system identifies that the liquid selected in the application is water.

3) Application: The complementary application was developed using MIT App Inventor, an open-source platform developed by Google and maintained by the Massachusetts Institute of Technology (MIT) [12]. This tool is known for its user-friendly interface and ability to integrate with various resources.

It was possible to demonstrate real-time Bluetooth communication between the application and the smart coaster using the MIT App Inventor. Additionally, the application allowed for effective data capture from the coaster and its display to the user. The main goal was to show that the data was being correctly collected and transmitted from the coaster to the application.

A prototype application was created using MIT App Inventor to illustrate how a more complete and customized application could be developed in specific programming languages. Although the main focus of this paper was to develop the coaster product, this step was relevant to demonstrate future expansion possibilities. Figure 3 shows a prototype of the application on an editing screen in MIT App Inventor.

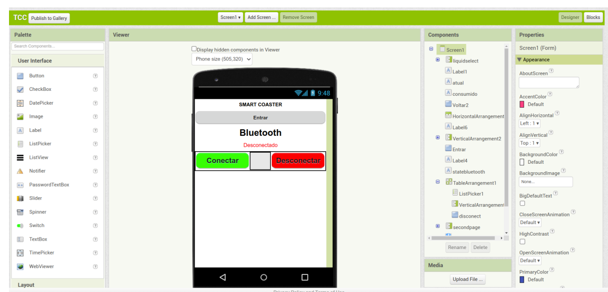

Fig. 3. Prototype application developed using MIT APP INVENTOR.

IV. Tests

Different tests were conducted to verify the stability of the recorded values and the accuracy in measuring the collected volume. The results of these tests are presented in detail in the following subsections.

A. Initial Test

Initially, tests were conducted to verify the consistency of the measurements. The same volume of liquid was repeatedly placed in the container to assess whether there would be significant variations in the values. A small error of approximately 1 mL was observed.

Table II presents the results obtained for a volume of 500 ml of water.The measured values show a small variation around the expected value of 500 g. These results demonstrate the smart coaster's ability to approximately measure the volume of liquid placed in the container. However, to ensure greater precision in measurements, additional adjustments and calibrations are necessary.

B. Test for Different Volumes

A container marked with different volume levels was used to evaluate the smart coaster's capacity and measurement accuracy relative to different weights recorded on the scale. The results for each volume level marked on the container are presented in Table III.

TABLE II
RESULTS OBTAINED FOR A VOLUME OF 500 ML OF WATER, WITH 10 MEASUREMENTS TAKEN BY THE SMART COASTER

Measurement	Measured Value (g)
1	500.77
2	500.81
3	501.01
4	501.92
5	500.88
6	501.07
7	500.74
8	500.95
9	500.84
10	500.92

TABLE III
MEASUREMENT VALUES OBTAINED BY THE SMART COASTER FOR DIFFERENT VOLUMES

Container Volume (mL)	Measured Value by Smart Coaster (mL)
80	78.82
140	140.98
210	209.43
280	278.26
360	358.09

C. Calculation of Consumed Value

The consumed value test was performed using a 500 mL bottle as a reference. This test verifies the smart coaster's accuracy in measuring the amount of liquid consumed over different periods, such as 1 day, 3 days, and 7 days. The comparison was made between the amount of liquid shown in the application and the number of bottles consumed during the determined period.

This analysis evaluates the system's accuracy in measuring liquid consumption over time and verifying whether the values displayed in the application corresponded to the actual number of bottles consumed. The values can be seen in Table IV.

TABLE IV
MEASUREMENT OF LIQUID CONSUMPTION OVER TIME

Time Period	Bottles Consumed	Accumulated Value (mL)
1 day	4	1986
3 days	11	5432
5 days	19	9420

The values presented in Table IV represent the number of bottles consumed and the values measured by the smart coaster over different periods. These data were used to evaluate the measurement system's accuracy.

D. Application Functionality Test

This test verified the Bluetooth communication between the smart coaster and the application developed in MIT App Inventor. The main objective was to evaluate the effectiveness of this communication, as well as the smart coaster's ability to receive and process commands from the application in real-time.

1) Receiving Measured Values: The test was done to analyse if the smart coaster could correctly receive and display the liquid measurement values. For that different containers with liquids were placed on the smart coaster and the accuracy of the measurements displayed in the application was verified.

2) Sending Calibration Functions: The application's ability to send calibration commands to the smart coaster was tested. The successful calibrations' performance and the smart coaster's response to this command were analysed.

3) Changing the Selected Liquid: The application's ability to allow the user to select different liquids (e.g., water, coffee, soda) and send this function to the smart coaster was evaluated. The liquids' recognition and the adjustment of the density according to the selected liquid ware tested.

V. CONCLUSION

The proposed smart coaster proved to be effective in measuring liquid intake, presenting a very low average error during the tests conducted. This stability in measurements suggests that the project is viable and promising when considering its implementation in the context of the Internet of Things (IoT).

In summary, this paper represents an example of how IoT devices can be used to improve quality of life by providing valuable information about habits that can impact various aspects such as health and well-being.

For future work, the following improvements are suggested to enhance the functionality and quality of the smart coaster, making it more suitable for use in various situations:

- Develop a more robust case;
- Investigate the possibility of manufacturing a custom PCB that integrates all components;
- Usage of temperature sensors to track the temperature of the beverage and notify users in real-time of how warm or cold their drink is.
- Create an user-friendly application for the coaster.

REFERENCES

[1] P. N, N. V. S. Sai Balaji, G. V. Varthan, R. Sivakumar, L. C. T. Maule, and D. Naveenkumar, "A review on intelligent water bottle powered by iot," in *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, pp. 2335–2338, 2023.
[2] J. Vithanage, R. de Silva, K. Karunaratne, M. D. Silva, P. Bogoda, R. Kankanamge, P. Kehelella, K. D. Jayakody, and J. L. Wijekoon, "Smartone: Iot-based smart platform to manage personal water usage," in *2019 International Conference on Advancements in Computing (ICAC)*, pp. 398–403, 2019.
[3] P. Pankajavalli, R. Saikumar, and R. Maheswaran, "Hydration reminding smart bottle: Iot experimentation," in *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)*, pp. 1–5, 2017.
[4] L. L. Jalagam, "Iot based smart water bottle," *Available at SSRN 3919060*, 2021.
[5] USGS - Science for a changing World , "Water density - u.s. geological survey," 2018. [Online; accessed 12-Jul-2024].
[6] A. P. Khomyakov, S. V. Mordanov, and T. V. Khomyakova, "The experimental data on the density, viscosity, and boiling temperature of the coffee extract," *IOP Conference Series: Earth and Environmental Science*, vol. 548, p. 022040, aug 2020.
[7] "Coke can buoyancy."
[8] Arduino, "Arduino ide." [Online; accessed 12-Jul-2024].
[9] "Hx711 library by:robtillaart hx711."
[10] "arduino-esp32 spiffs library."
[11] "Bluetooth serial library."
[12] M. I. of Technology, "Mit app inventor - about us." [Online; accessed 12-Jul-2024].