A HEALTH STATUS MONITORING AND WATER INTAKE TRACKING SYSTEM BASED ON SMART CUP USING ARTIFICIAL INTELLIGENCE AND INTERNET OF THINGS

Chloe Hsu[1], Jonathan Sahagun[2]

[1]Harvard-Westlake School, 3700 Coldwater Canyon Ave, Los Angeles, CA 91604
[2]Computer Science Department, California State Polytechnic University, Pomona, CA 91768

ABSTRACT

Dehydration is a common and serious condition that affects millions of people globally, despite increasing access to clean drinking water. Staying hydrated is one of the most important habits for maintaining overall health and energy levels but sadly, due to higher levels of business among people. It is often overlooked. This causes health issues like fatigueness, headaces and etc [1].Dehydration is linked to various health issues, including heat-related illnesses, kidney diseases, and even death. Studies show that 75% of Americans are chronically dehydrated, with elderly populations being at higher risk. Addressing this, our project proposes a solution to ensure optimal hydration: a smart water bottle, Smartflask, integrated with sensors and an application to monitor and encourage regular water consumption. The Smartflask app allows users to set daily hydration goals, and through its combination of a distance sensor placed at the bottle's cap and real-time notifications, it tracks the user's water intake and sends reminders when needed. This smart system ensures that users meet their hydration targets consistently. Several challenges were addressed during the development, including selecting a suitable microcontroller (Particle Boron), ensuring sensor accuracy, and designing a user-friendly interface. The results of our experiments show that the ToF distance sensor is accurate in measuring water levels and that the smart water bottle's performance can be enhanced by optimizing power consumption and sensor integration. This project showcases the potential of smart technology to significantly improve hydration habits and overall health.

KEYWORDS

Hydration, Smart Bottle, Dehydration

1. INTRODUCTION

In present day, more and more countries are starting to have access to clean water. However, what will ensure the adequate amount of hydration our body needs when water is accessible [2]? Our bodies are made up to 60% of water, with main organs like the brain and heart being composed with 73% of water. Hence, when our body does not obtain the amount of water it requires, it can lead to dehydration [3]. Dehydration can be the cause to many serious diesases and conditions like certain heat-related illnesses; heat cramps, heat exhaustion, and potentially even life-threatening heat strokes [4]. It can also lead to kidney problems, ranging from urinary

David C. Wyld et al. (Eds): NWCOM, ASOFT, ITCCMA, CSITY, BDIoT, MLNLP, SIGPRO, AIFZ– 2024
pp. 167-175, 2024. - CS & IT - CSCP 2024 DOI: 10.5121/csit.2024.141915

tract infections and kidney stones, to complete kidney failure [5]. Severe dehydration can lead to reactions like electrolyte imbalance, where there is too much or too little minerals in your body, shock, where your body is not getting enough blood flow, and brain damage, which can lead to death [6]. In 2015, there were 505 deaths in United Kingdom where dehydration was a contributing factor of their death. According to researches conducted in the past few years, about 518,000 people are hospitalized annually due to dehydration, and 10,000 of them die in US hospitals each year; in addtion, 75% of Americans are chronically dehydrated with 15%-25% of them being elders.The National Library of Medicine said that dehydration is considered a common condition that can affect people of all ages. The United States of America is a country that has one of the safest and most reliable drinking water supplies in the world; in 2022, 97.47% of people had access to clean water from a safe and improved source. However, the previous statistics still clearly exhibit many Americans are getting illnesses and diseases due to lack of water, or dehydration. This enforces the concern of ensuring the right amount of hydration when clean water is provided and accessible.

In the Methodology section, we mainly focused on comparing the solution methods tackling similar problems, like measuring hydration levels, used for my project to different scholarly sources. The first methodology, methodology A, the researches used an accelerometer to track the fill level of a water bottle. This method can be effective in certain scentarios where the accelerometers are provided with perfect temperature, humidity, and environment. However, these requirements seem to be excessive. This method relies on indirect measurements of the bottles tilt, causing it to be heavily influenced by factors like the bottle’s design and the accelerator’s placement. The second methodology, methodology B, uses ultrasonic sensors to measure liquid levels in the bottle. Except, this approach is more suitable and fit for larger scaled applications as it can be easily disrupted by noise. Hence, when applied to water bottles used for my project, it can lead to many limitations regarding its accuracy and sensitivity. Lastly, methodology C tackles the problem with a slightly different perspective. This method combines machine learning with certain wearable sensors, like fitness trackers, to monitor dehydration. As this can provide us with very accurate data and statistics, it also requires the use of multiple sensors. This inconvenience may lead to it not being suitable for all users due to its complexity and large power consumption. Nonetheless, our project improved upon these different approaches by ultimately using a ToF distance sensor mounted on the water bottle cap [7]. This method provides bith a direct and accurment measurement of the liquid level, is less complex, and is more resistance to environment factors. As this focuses more on a specific aspect of monitoring, or water consumption, and uses a more suitable senor, my project offers a more practical and user-friendly solution.

The solve this problem, we have proposed a smart waterbottle that will ensure our user’s hydration on a daily basis. With the software we have used to built the Smartflask app, our user can set a goal of how much water that would like to consume daily. That number will then be sent into our system, and with a combination of our hardware and software, majorily the distance sensor, will ensure that the goal is reached [8]. The sensor is placed on the bottle’s cap, having the ability to calculate the amount of water the user has drunk. Through calculations, we can use that number and calculate the amount of water our user still needs to drink to fufill their goal. When our program senses that water consuming is necessary for the user, it will trigger the notification system, sending a notification to the user’s devices, whether it be their watch or phone. These notifications will remind the user to drink water at certain moments throughout the day, ensuring proper hydration. This is an effective solution because with the wide use of mobile devices today, these notifications will act as if it is “forcing” the user to hydrate and drink water despite any circumstances.

In the experiment section of the paper we mainly focused on testing potential blind spots in the project. The two experiments explored water resistance of the senor and the data refresh speed within the application. The first Experiment aimed to assess the sensor's functionality in wet environments. By sealing the senor with various materials, the experiment determined that Kapton tape offered the best water resistance while maintaining sensor accuracy simultaneously. It is worth mentioning that even though Silicone and clear washers proved effective as seals, they did not allow the sensor to have accurate readings leading to them obstructing the functionality.
Moreover in the second experiment, we evaluated the impact of server's ability on data refresh and loading times. The results show that a web server improved performance significantly compared to one without the advancement. Local servers offered the fastest speed, but cloud servers provided a better balance between both speed and accessibility.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Choosing the Right Microcontroller

One of the main challenges for this project was choosing the right microcontroller. There were many factors to take into consideration such as the power consumption, ease of firmware update, connectivity, and speed. There were many boards that were faster and more efficient than Particle Boron, however LTE connectivity and over the air updates for the firmware made it a better choice over it's competitors [9]. These features provided significant advantages in terms of remote management and real-time data transmission, making the Particle Boron a more suitable choice despite its competitors' better processing power.

2.2. Ensuring Data Accuracy

While developing this project, another challenged faced was ensuring data accuracy. In order to provide precise calculations as to when it would be best for the user to drink water, a sensor that can bring forth precise and stable distance measurements, as well as having fast measurement updates is crucial. As there are other choices like infrared reflectance sensors and capacitive sensors, neither of them offer as accurate measurements as time of flight sensor VL53L4CD. This sensor is known for its accuracy in distance measurements, achieving precision to tenths of milimeters. In addition, it provides updates/refreshes at the rate of tens or even hundreds of measurements per second. These superior factors making the VL53L4CD ToF sensor to be a more efficient choice.

2.3. Designing a User Interface

Moreover one of the other key challenges in development was designing a user interface (UI) that is both elegant and simple. The UI needs to be intuitive for users while also providing all necessary features without overwhelming them. Potential problems that could arise include cluttered layouts, confusing navigation, or visually unappealing design elements that detract from the user experience. To address these issues, I could use a minimalist design approach, focusing on clear, consistent navigation and a clean layout that highlights the most important features. Additionally, conducting user testing and gathering feedback could help identify areas of confusion or frustration, allowing for iterative improvements to ensure the interface remains user-friendly and aesthetically pleasing.

3. SOLUTION

The app mainly utilizes Dart code, provided in the program Flutterflow. When the user registers for an account, they have the ability to set hydration goals they want to achieve on a daily basis. On the hardware side of things,the head of the operation, particle boron, controls and reads the distance sensor data. The distance sensor plays a large role in ensuring the user is staying hydrated and achieving their goals, as it determines the distance between the top of the bottle to where the water is. In other words, the distance sensor determines how much water how been drank, hence calculating the amount of water the user still has to drink. These data are then sent and stored in the Firebase, where it can be updated and synchronized whenever the user needs to, even when offline. The data is also transferred to the bottle itself; when the user needs to drink water to maintain their goal, the app will then send a notification to the user to make sure they are staying hydrated.

The Main 3 components are going to be

1. Particle boron to connect to the sensor and send data to firestore
2. The app itself, showing the water level so users can check
3. UV light

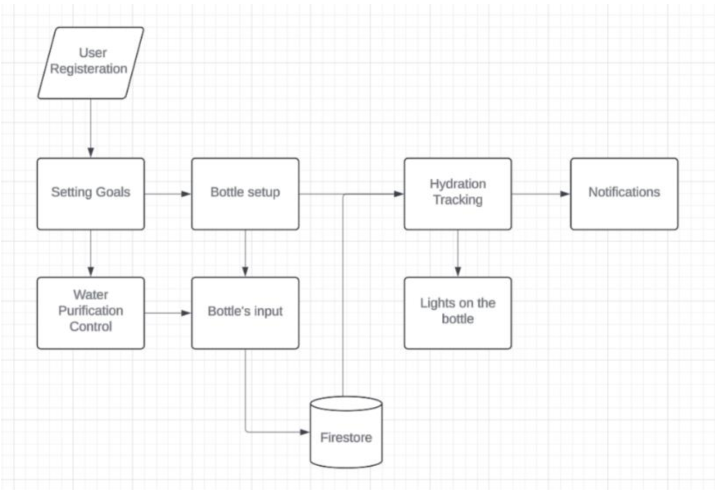
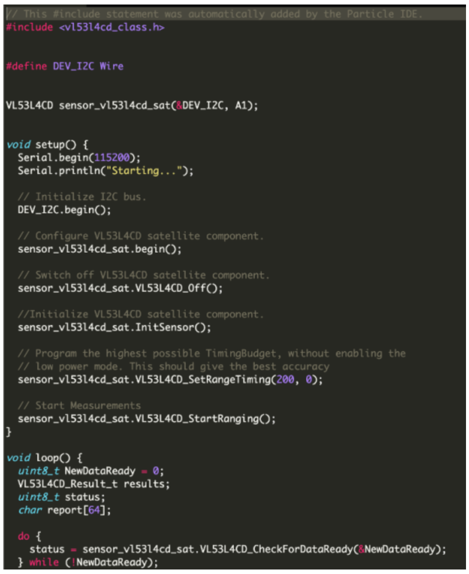


Figure 1. Overview of the solution

One of the most important components in this project is Particle boron that is going to connect to the sensor and send data to firestore [10]. The component plays the key role of identifying how much water our user has consumed to calculate the amount they need to drink in order to reach their goal. As our database uses nearest cellular towers, our database can be updated every second, sending information and storing it in our firestore.



```
// This #include statement was automatically added by the Particle IDE.
#include <vl53l4cd_class.h>


#define DEV_I2C Wire


VL53L4CD sensor_vl53l4cd_sat(&DEV_I2C, A1);


void setup() {
  Serial.begin(115200);
  Serial.println("Starting...");

  // Initialize I2C bus.
  DEV_I2C.begin();

  // Configure VL53L4CD satellite component.
  sensor_vl53l4cd_sat.begin();

  // Switch off VL53L4CD satellite component.
  sensor_vl53l4cd_sat.VL53L4CD_Off();

  //Initialize VL53L4CD satellite component.
  sensor_vl53l4cd_sat.InitSensor();

  // Program the highest possible TimingBudget, without enabling the
  // low power mode. This should give the best accuracy
  sensor_vl53l4cd_sat.VL53L4CD_SetRangeTiming(200, 0);

  // Start Measurements
  sensor_vl53l4cd_sat.VL53L4CD_StartRanging();
}

void loop() {
  uint8_t NewDataReady = 0;
  VL53L4CD_Result_t results;
  uint8_t status;
  char report[64];

  do {
    status = sensor_vl53l4cd_sat.VL53L4CD_CheckForDataReady(&NewDataReady);
  } while (!NewDataReady);
```

Figure 2. Screenshot of the code 1

This code is used to initialize the VL53L4CD sensor on the Particle Boron device. It ensures the sensor has optimal accuracy and starts the ranging process. The #include <vl5314cd_class.h> statement includes the header file for the VL53L4CD class, providing functions to interact with the sensor. The <#define DEV_I2C> Wire line then defines the I2C bus to be used for communication with the sensor. After that, Serial.begin(115200) sets up serial communication at a baud rate of 115200, allowing for debugging and data output, DEV_I2C.begin() then starts the I2C bus. The line sensor_v15314cd_sat.begin() initializes the VL53L4CD sensor, while the line sensor_v15314cd_sat.VL53L4CD_0ff() temporarily turns off the sensor. After that, sensor_v15314cd_sat.InitSensor() re-initializesthe sensor and sensor_v15314cd_sat.VL53L4CD_SetRangeTiming(200, 0) sets the range timing for the sensor to 200 milliseconds, providing the best accuracy without enabling low power mode. Last but not least, sensor_v15314cd_sat.VL53L4CD_StartRanging(); starts the ranging process. A large part of the code is for it to continuously check for new measurement data and processes it, having the use of loops necessary. The do...while loop continuously checks if new data is available from the sensor using the code, sensor_vl5314cd_sat.VL53L4CD_CheckForDataReady(&NewDataReady), and once new data is available, the sensor_vl5314cd_sat.VL53L4CD_GetRangingMeasurement(&results) function reads the measurement results into the results structure. Hence, the measurement results are then processed (e.g., distance, timing) and printed to the serial monitor using the Serial.print() and Serial.println() functions. In summary, this code sets up the VL53L4CD sensor on the Particle Boron, starts the ranging process, and continuously reads and processes the measurement data.

A main part of this project is the UV light. Throughout the day, the water the user consumes can get exposed to many different sorts of bacteria. Hence, this component plays an essential part in disinfecting the water in the water bottle, ensuring that while the user is staying hydrated, they are also consuming clean water.

Another major part of my project are the notification sent to the user to make ensure their hydration. As our user has an initial goal, our app is responsible to ensure they fufill that goal. Hence once our distance sensor calculates that our user needs to consume water, it will trigger the notification and send that notification to our user, reminding them to drink water.

Figure 3. Screenshot of smartflask

4. EXPERIMENT

4.1. Experiment 1

One of the possible blind spots of this project can be the water resistance of the sensor and electronics from the water itself to allow for measurements to take place. This part is important to this program because the distance sensor has the major role of calculating the correct amount of water our user has drunk. Hence, it needs to be placed at the top of the waterbottle cap. However, we have faced a difficulty regarding what product can be used in order to not only keep it stable, but also ensure that it is edible/drinking safe.

The experiment involves testing the water resistance and measurement accuracy of the VL54L4CD sensor when exposed to water. The sensor is mounted inside the water bottle's cap, covered with various sealing materials like silicone, Kapton tape, and clear washers to ensure watertight integrity. The setup aims to evaluate the effectiveness of each material in maintaining sensor functionality while preventing water ingress. Control data is sourced from a dry environment where the sensor's distance readings are unaffected by water. By comparing these readings to those taken in the sealed environment, we can assess each material's impact on accuracy.

Material	Data Recorded	Water Resistant
Kapton Tape	Yes	Yes
Silicone	No	Yes
Clear Washer	No	Yes
No Seal	Yes	No

Figure 4. Figure of experiment 1

The experiment showed that Kapton tape allowed the sensor to record data, though the readings were occasionally finicky. This suggests that while the tape's thinness and transparency were generally effective, its adhesive properties or application method might have caused sporadic inconsistencies.

In contrast, silicone and clear washers completely prevented the sensor from recording any data. Silicone's thickness and opacity likely obstructed the sensor's ability to function, while the clear washers may have introduced physical barriers or refracted the sensor’s signal in a way that blocked accurate readings.

Without any seal, the sensor was able to record data, but this setup left the electronics exposed to potential water interference, highlighting the necessity of a watertight seal.

Overall, while Kapton tape proved to be the most viable solution, further refinements are needed to enhance its consistency and reliability. Future iterations might explore alternative materials or sealing techniques to achieve optimal performance without compromising water resistance.

4.2. Experiment 2

Another potential blind spot of this project is the data refresh speed within the application. Quick data refresh and loading times are crucial for a seamless user experience.

Server Configuration	Data Refresh Speed (ms)	Data Load Time (ms)
Local Server	100	200
Cloud Server (deployed on Render)	150	250
Cloud Server (Ran on Firebase App hosting)	120	220
No Optimization	300	500

Figure 9. Figure of experiment 2

The experiment revealed that implementing a web server significantly improved data refresh and loading times. The local server configuration provided the fastest refresh speed at 100 ms and a loading time of 200 ms, making it the most responsive option. However, cloud server configurations also performed well, with Cloud Server B showing a slight advantage over Cloud Server A due to better latency management.

Without optimization, the app experienced much slower refresh (300 ms) and load times (500 ms), leading to a suboptimal user experience. The experiment highlights the importance of server setup in enhancing app performance. While local servers offer the best speed, they may not be feasible for all users due to accessibility concerns. Cloud servers provide a good balance between speed and accessibility, making them a viable option for most scenarios. Future improvements might focus on further optimizing server-side code or exploring faster cloud hosting solutions to ensure the best user experience.

5. RELATED WORK

Researchers Griffith, Shi, and Biswas have proposed to measure hydration levels by tracking the fill level of a refillable water bottle using an accelerometer. The sensor module continuously gathers acceleration data, processes it to determine the bottle's orientation, and estimates the remaining water level based on the tilt. While the article claims that the sensor module can accurately estimate the fill level in real-time, the effectiveness of this approach depends on several factors, including bottle design, sensor placement, environmental factors, and user behavior. The method has limitations, such as indirect measurement of hydration and user compliance. Our method contains the use of a ToF sensor. This method provides a much more direct and accurate measurement, as it doesn't require complex calculations from the tilt, instead simply just provides a direct measurement of the distance to the liquid surface. As the sensor is mounted on the cap, its readings are directly correlated to the liquid level. This reduces the complexity of the system compared to accelerometer-based approaches.

Researches Rodin, Shapiro, Pinhasov, Kreinin, and Kirby explores the use of ultrasonic sensors for liquid level measurement. The solution involves mounting a sensor above the liquid, emitting ultrasonic pulses, measuring the time difference between emitted and received echoes, calculating distance, and determining the water level. This approach is effective for large-scale storage tanks but can be influenced by sensor positioning, environmental factors, and noise. Adapting this solution to a water bottle might require adjustments, and ultrasonic sensors might have limitations in accuracy, cost, and sensitivity to water clarity. ToF distance sensors, like the VL53L4CD, offer advantages in accuracy, range, speed, and resistance to environmental factors, making them more suitable for water bottle applications.

Researches Sabry, Eltaras, Labda, Hazma, Alzoubi, and Malluhi focuses on a different aspect of hydration monitoring combine machine learning with hydration monitoring using wearable devices. In this study they use Accelerometer, magnetometer, gyroscope, galvanic skin response sensor, Photoplethysmography sensor, temperature sensor and barometric sensor to tell the user when its time to drink water. In this approach they are measuring dehydration rather than the amount of water that the user has consumed. Although, this can be a very accurate approach, using many sensors will have complications with both power efficiency and user experience. Having the ability to track the amount of water that is consumed everyday and build habits on when and how much to drink water, based on personilized information such as the weight, diet, and their BMI would be a more suitable solution for most users.

6. CONCLUSIONS

The smart water bottle project, while promising, faces certain limitations that could be addressed with further development. One such limitation is battery life. The integration of various features, such as real-time water level tracking, LED displays, and wireless connectivity, can significantly drain the battery [14]. To overcome this, future iterations could explore energy-efficient components, optimize software algorithms, and implement intelligent power management strategies [15]. Another area for improvement is sensor accuracy. While the chosen sensor provides reasonable accuracy, factors like water temperature, impurities, and sensor placement can influence readings. Implementing more robust calibration methods and exploring alternative sensor technologies could enhance accuracy. Additionally, the user interface might require further refinement to ensure a seamless and intuitive experience. Gathering user feedback and conducting usability testing can help identify areas for improvement and optimize the overall interaction. Finally, the cost of implementing a smart water bottle might be a barrier for some consumers. Exploring cost-effective components, streamlining manufacturing processes, and offering budget-friendly versions could increase accessibility.

This project has demonstrated the potential of smart technology to enhance hydration habits. By incorporating features like water level tracking, hydration reminders, and personalized recommendations, the smart water bottle can motivate users to stay hydrated and improve their overall health. Future improvements in battery life, sensor accuracy, and user interface design could further enhance the product's appeal and effectiveness.

REFERENCES

[1] Baek, Chang-Hwan, et al. "The analysis of fatigueness and preference for three appearance attributes of LED light color." Journal of the Korean Institute of Illuminating and Electrical Installation Engineers 25.1 (2011): 18-29.

[2] Crandall, Lathan A., and Milford X. Anderson. "Estimation of the state of hydration of the body by the amount of water available for the solution of sodium thiocyanate." The American Journal of Digestive Diseases 1.2 (1934): 126-131.

[3] Thomas, David R., et al. "Understanding clinical dehydration and its treatment." Journal of the American Medical Directors Association 9.5 (2008): 292-301.

[4] Barrow, Michael W., and Katherine A. Clark. "Heat-related illnesses." American family physician 58.3 (1998): 749-756.

[5] Darrah, Marjorie, et al. "Are virtual labs as effective as hands-on labs for undergraduate physics? A comparative study at two major universities." Journal of science education and technology 23 (2014): 803-814.

[6] Balcı, Arif Kadri, et al. "General characteristics of patients with electrolyte imbalance admitted to emergency department." World journal of emergency medicine 4.2 (2013): 113.

[7] Lindner, Marvin, et al. "Time-of-flight sensor calibration for accurate range sensing." Computer Vision and Image Understanding 114.12 (2010): 1318-1328.

[8] Brunner, P., et al. "Current trends in hardware and software for brain–computer interfaces (BCIs)." Journal of neural engineering 8.2 (2011): 025001.

[9] Rosa, Claudio, et al. "Dual connectivity for LTE small cell evolution: Functionality and performance aspects." IEEE Communications Magazine 54.6 (2016): 137-143.

[10] Yeh, Chun-Liang, and K. K. Kuo. "Ignition and combustion of boron particles." Progress in Energy and Combustion Science 22.6 (1996): 511-541.

[11] Griffith, Henry, Yan Shi, and Subir Biswas. "A container-attachable inertial sensor for real-time hydration tracking." Sensors 19.18 (2019): 4008.

[12] Rodin, Dmitry, et al. "An accurate wearable hydration sensor: Real-world evaluation of practical use." Plos one 17.8 (2022): e0272646.

[13] Sabry, Farida, et al. "Towards on-device dehydration monitoring using machine learning from wearable device's data." Sensors 22.5 (2022): 1887.

[14] Huang, Yuge, et al. "Prospects and challenges of mini-LED and micro-LED displays." Journal of the Society for Information Display 27.7 (2019): 387-401.

[15] Shestakofsky, Benjamin. "Working algorithms: Software automation and the future of work." Work and Occupations 44.4 (2017): 376-423.