TONIQ Final Report

CODY RUPP, SAMIR RASHID, KYLE TRINH, and ANTHONY TARBINIAN, University of California, San Diego, USA

TONIQ is a smart water bottle that improves the well-being of its users by tracking critical health metrics such as water consumption and water quality, and presents them on an integrated e-ink display. Additionally, the lid of the bottle (containing the electronics) is fully independent of the bottle itself, and fit to the exact specifications of a standard wide-mouth water bottle opening. Existing smart water bottles have many flaws: they are expensive, inflexible, reliant on a mobile app for tracking, and may require nonstandard charging cables prone to failure. We aim to solve these key issues while maintaining an elegant and user-friendly design in one container: *TONIQ*.

CCS Concepts: • **Hardware → PCB design and layout**; • **Computer systems organization → Embedded systems**.

Additional Key Words and Phrases: Industrial Design, Manufacturing, PCB Design, Embedded Firmware

ACM Reference Format:
Cody Rupp, Samir Rashid, Kyle Trinh, and Anthony Tarbinian. 2025. TONIQ Final Report. *Proc. ACM Meas. Anal. Comput. Syst.* 37, 4, Article 111 (August 2025), 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

The proliferation of smart, connected devices has extended into nearly every aspect of daily life, including personal health and wellness. In this space, the smart water bottle has emerged as a popular consumer product, aiming to help users monitor hydration and maintain healthy habits. However, the current market is saturated with products that, while innovative, often introduce significant user friction. Many existing smart water bottles are expensive, lock users into a proprietary ecosystem with a single, non-interchangeable bottle, and rely heavily on a companion mobile application for core functionality. This dependency on a smartphone not only complicates the user experience but also introduces another device to be managed and charged, often with nonstandard cables prone to failure.

This project, TONIQ, was conceived to address these deficiencies by fundamentally rethinking the architecture of a smart water bottle. Our approach inverts the conventional design: instead of embedding technology into the bottle itself, we have engineered a self-contained, intelligent lid that is fully independent of the container. This lid is designed to the exact specifications of a standard wide-mouth opening, enabling it to be used with a wide variety of existing, popular water bottles, such as a 32 oz. Hydro Flask. By decoupling the technology from the vessel, TONIQ offers users unparalleled flexibility and freedom from a single proprietary ecosystem.

TONIQ improves its user's well-being by tracking critical health metrics and presenting them on an integrated, low-power e-ink display. This removes the need for a mobile app to check status, aligning with our goal of an elegant and self-sufficient user experience. Its core features include water consumption tracking via an ultrasonic sensor, water quality measurement in parts per million (ppm) using a Total Dissolved Solids (TDS) sensor, and

Authors' Contact Information: Cody Rupp, crupp@ucsd.edu; Samir Rashid, s3rashid@ucsd.edu; Kyle Trinh, kytrinh@ucsd.edu; Anthony Tarbinian, atarbini@ucsd.edu, University of California, San Diego, La Jolla, California, USA.

ACM 2476-1249/2025/8-ART111
https://doi.org/XXXXXXX.XXXXXXX

an automated UV-C sterilization system to kill harmful microorganisms present in the liquid [7, 12]. To ensure accurate readings, an onboard Inertial Measurement Unit (IMU) allows the system to take measurements only when the bottle is stationary and upright [11].

The entire project was guided by several core design principles. First and foremost was power efficiency, a critical requirement for a device intended to be used for long periods without frequent charging. This was achieved through the careful selection of a Nordic MCU capable of ultra-low-power sleep states, the use of an e-ink display that consumes effectively zero power when not actively refreshing, and a replaceable CR2450 coin cell battery chosen for its high capacity and ease of replacement by the user. The project was executed by dividing work into four parallel tracks—Mechanical, Electrical, Firmware, and Manufacturing—allowing for concurrent progress on all fronts.

The main contributions of this work are summarized as follows:

- The design and implementation of a fully integrated, standalone smart water bottle lid containing all sensor, processing, and display components.
- A modular mechanical design that is cross-compatible with standard wide-mouth water bottles, freeing the user from a single proprietary system.
- An app-independent user experience centered on an onboard e-ink display, which presents all relevant health data directly on the device.
- The development of a custom-designed Printed Circuit Board (PCB) and associated embedded firmware, highly optimized to operate within the severe spatial and power constraints of a bottle lid.
- This paper details the design, development, and implementation of the TONIQ prototype. Section 2 reviews related works in the smart water bottle market. Section 3 provides a deep dive into the technical material, covering the mechanical design, PCB development, and firmware architecture. Section 4 discusses the project milestones and challenges encountered. Finally, Section 5 concludes with a summary of the work and directions for future development.

2 Related Works

Many smart and health-focused water bottles have come and gone. We performed market research to examine the existing smart water bottles and where they fall short. In addition, it provided invaluable knowledge to our mechanical design.

SGUAI is an insulated smart mug which boasts drink temperature measurement, a LED matrix display, and wireless charging [1]. SGUAI also comes with a mobile application which allows users to reprogram the LED matrix from a plethora of premade graphics or draw their own graphics right on their smartphone. SGUAI was useful during our development as a reference for how electronics were routed through the body of the water bottle. The SGUAI bottle has an inner insulated layer which touches the actual liquids and maintains their temperature. On the outside is a plastic layer which is visible to the user. Between the two layers is where the electronic routing was located. Since the SGUAI's lid has no electronics in it, there's no electronic routing between the body of the bottle. The wiring goes across the body of the bottle to the bottom where

Trago is a smart water bottle startup founded in 2015 [9]. They launched a Kickstarter which raised over $70,000 [5]. Trago is centered around health and tracking water consumption primarily [10]. The Kickstarter page describes it as: "Trago is a patent-pending water bottle that accurately measures how much you drink. It connects to your smart phone, and other wearable devices, and allows you to set, and measure, intelligent hydration goals. Optimal hydration is both art and science, and it's different for everyone. Trago helps you pinpoint the perfect amount for you." [5]. Trago also boasts compatibility with any wide-mouth water bottle, allowing it to be used with the consumer's preferred bottle they might already own.

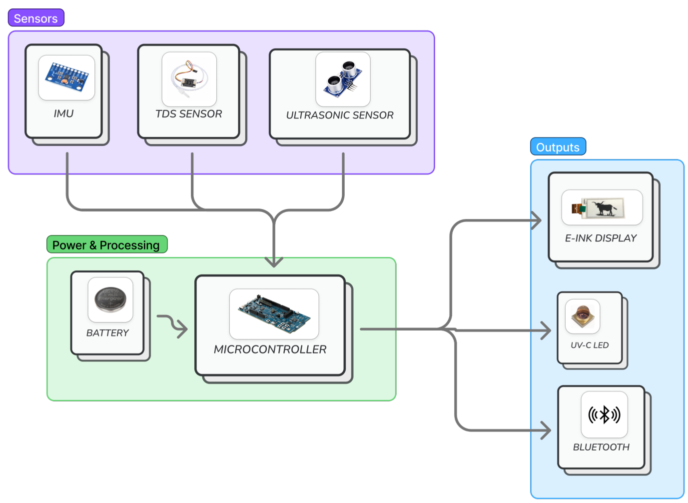

Fig. 1. High-Level System Diagram

SmartStuff is another smart water bottle targeting increased water intake [13]. Interestingly, SmartStuff combined sensor data from the bottle itself and an accompanying smart watch to put together meaningful takeaways about the user's water drinking patterns. Outside of their sensor implementation and system design, they investigated multiple implementations with their own networking strategy. One approach opted to upload data directly to the cloud while the other one involved communication to a personal device such as a smartphone.

For developing the firmware for the project, we elected to use the nRF52840dk alongside the Zephyr embedded operating system for its strong support for BLE which was initially planned as a more integral part of the project [4, 8]. The development environment was comprised of the nRF Connect VS Code extension alongside Zephyr's build tool: `west` [3, 6].

3 Technical Material

3.1 Mechanical Design and Manufacturing

3.1.1 Design Constraints. In designing the mechanical components of our system, we identified three key constraints: perfecting the threading tolerance, fitting all the necessary electronics in the lid in a functional and user-friendly manner, and waterproofing said electronics. One particular design challenge that we ran into early on was the placement of the e-ink display. Since we initially had planned on using Waveshare's 2.13 inch flexible display, we wanted to place it curved on the outside, flush with the lid so that users could easily view the display interface. We also briefly explored placing the display on the top, but ultimately decided against that due to worries that the non-circular shape of the display would look ugly on the circular lid, as well as complicating future battery placement options.

Another challenge we ran into was the placement and waterproofing of the TDS and ultrasonic sensors, which needed to protrude from the bottom of the lid and be able to contact the water safely. Our eventual choice for

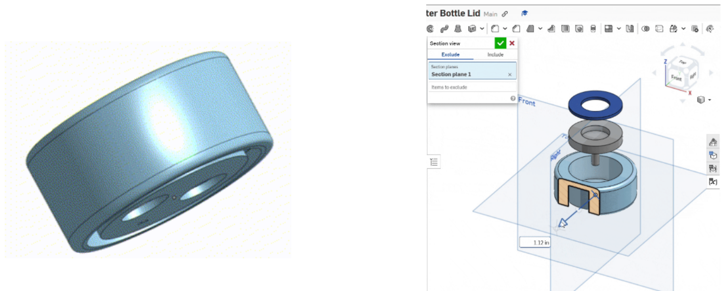

Fig. 2. Lid 3D Design

these components was to leave holes for just the probes of the TDS and just the transducers of the ultrasonic sensor such that the holes had a very tight fit with the components in. In an ideal design, we would have liked to implant a gasket within these holes, and additionally sealed the gasket from the inside with a food-safe material like an epoxy.

In order to fit all the electronics within the roughly 6 cubic inches of space inside the lid, we referenced the designs from the existing products that we tore down and inspected earlier. These teardowns heavily influenced our eventual PCB design which effectively integrated all the electronics while minimizing the amount of space used.

3.1.2 Prototype Manufacturing. To manufacture the prototypes for the bottle, we used a combination of 3-D printing, hot glue, and friction-fit components to integrate as many of the electronics as we could while still maintaining the bottle's usability. The mechanical lid prototypes were all 3-D printed with gray PLA using a Bambu Labs A1 Mini printer. We found that starting the print base from different sides of the lid yielded different manufacturing pros and cons. For instance, printing with the bottom of the lid (the side that would contact the water) facing down would result in the most structurally sound prototypes, but due to the deep threading the supports would be near impossible to remove. When printed the other way (with the top of the lid facing down), the number of supports were minimized making it much easier and cleaner to assemble a prototype lid. However, it tended to come with defects along the top side of the lid due to the curvature along the edge. Thus, we chose to print with the top side down when we needed quick prototypes to test tolerances, and we chose to print with the bottom side down when we needed a more polished prototype for testing and demoing.

Our approach to electronics prototyping was methodical, centered on a "divide and conquer" strategy to mitigate integration risks. Before attempting to combine all components, we focused on validating each piece of hardware individually. We separated the project into distinct tasks, with team members assigned to write specific firmware drivers for each hardware component and verify its functionality on its own dedicated breadboard and development kit. This independent validation of parts like the ultrasonic sensor and TDS module ensured the hardware performed as expected and allowed us to parallelize the firmware development process effectively. The ultimate goal of this phase was to produce a Minimum Viable Product (MVP) using development kits, which would then serve as the blueprint for the final, integrated PCB design.

In practice, this meant tackling each sensor's unique requirements. Prototyping the Total Dissolved Solids (TDS) sensor involved writing firmware to sample analog data from the sensor's probes using the microcontroller's Analog-to-Digital Converter (ADC). For the Inertial Measurement Unit (IMU), we used a breakout board to

Fig. 3. Several Iterations of Lid Prototypes

establish communication over I2C and successfully read gyroscope data to determine the bottle's orientation. The ultrasonic sensor required a more custom approach; after determining that existing modules were too large, we prototyped a solution that involved boosting the voltage to the transducer to allow it to read the water level through the plastic lid. To test that the readings were coherent, we tried various water dispensers around UCSD CSE department building.

To manufacture the prototypes for the bottle, we used a combination of 3-D printing, hot glue, and friction-fit components to integrate as many of the electronics as we could while still maintaining the bottle's usability. The mechanical lid prototypes were all 3-D printed with gray PLA using a Bambu Labs A1 Mini printer. We found that starting the print base from different sides of the lid yielded different manufacturing pros and cons. For instance, printing with the bottom of the lid (the side that would contact the water) facing down would result in the most structurally sound prototypes, but due to the deep threading the supports would be near impossible to remove. When printed the other way (with the top of the lid facing down), the number of supports were minimized making it much easier and cleaner to assemble a prototype lid. However, it tended to come with defects along the top side of the lid due to the curvature along the edge. Thus, we chose to print with the top side down when we needed quick prototypes to test tolerances, and we chose to print with the bottom side down when we needed a more polished prototype for testing and demoing.

The e-ink display proved to be the most complex component to prototype for the bottle lid. Originally, we chose to place the e-ink display on the curved side of the lid, but after discovering that the "flexible" e-ink display was inoperable while bent (at any nontrivial angle), we had scrap our entire design and move the display to the top of the lid.

3.1.3 Food Safety. Although we were not able to implement a fully food safe design in the time frame allotted for this class, we had plans for how exactly we would do it given the time. First we would need to contact a bottle manufacturer, and make a standard wide mouth water bottle body out of double-insulated stainless steel (much like a Hydro Flask). Then, for the lid material, we would injection mold a BPA-free polypropylene plastic version of our existing CAD model. The electronics would be housed inside this, and most would not contact the water. The only components that came in direct contact with the water were the TDS and the ultrasonic sensors. The probes of the TDS module are already food-safe stainless steel, and so only the holes which the probes came through would need to be sealed with some food-safe sealant like epoxy. For the ultrasonic sensor, we could

Item	Cost
E-ink display	$20
E-ink driver board	incl.
Lid polypropylene	$0.10
Lid gasket	$0.10
Coin cell battery	$1
PCB	$5
Ultrasonic sensor	$5
UV-C LED	$2
Vacuum sealed bottle	$2
SUM	**$35.20**

Table 1. Bill of materials for initial prototype

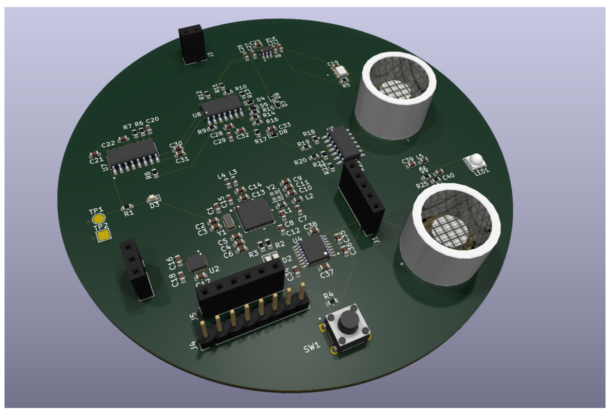

Fig. 4. PCB Render

either use waterproof transducers made of aluminum, or we could choose to leave the transducers behind a wall of acoustically transparent material so that they were still able to maintain a stable signal while being shielded from contact with the water. In the end, we decided that the acoustically-transparent design would be the most reliable, as we believed that the aluminum could corrode over time or alter the taste of the liquid in the container.

3.2 PCB Design and Manufacturing

To fully integrate all system electronic components in a space-efficient manner, we decided to develop a Printed Circuit Board (PCB) for our application. At its core, the TONIQ PCB is an nRF52840-equipped sensor board programmed over SWD with expansion headers for an external e-ink display, as well as an antenna for Bluetooth connectivity.

3.2.1 Schematic Design. We designed the schematic for our PCB largely by integrating existing schematic designs for each of the different features into one cohesive system. For example, we started by building the nRF52840 example schematic shown in the product specification document, and then making the necessary peripheral

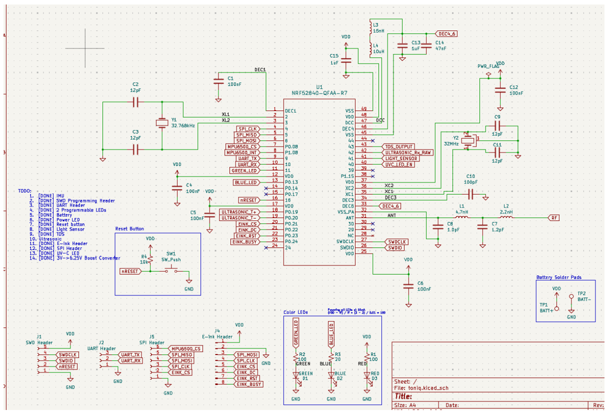

Fig. 5. MCU Schematic

connections to other sensors where needed (e.g. I2C, SPI, etc.). We did this for nearly all components except for the ultrasonic transceiver, where we designed the circuit from scratch.

3.2.2 Ultrasonic Transceiver Circuit. To implement our ultrasonic transceiver design, we created a custom circuit that amplified and rectified the received signal from the transducers. This is not something commonly implemented in ultrasonic sensor applications, as they usually have a clear line of sight and/or dry environment that would not warrant the need for such signal justification. However, since the sensor in our application is constrained by space, food safety, water proofing, and power supply requirements, the choice to develop our own circuit rather than use an existing one found online was critical.

3.3 Firmware

E-ink Display

The e-ink display firmware development was one of the most challenging aspects of the project. To provide a power-efficient, informative, and ergonomic medium to display information about the water bottle, we opted for a E-Ink display (or e-paper) to present users with water bottle data. Initial efforts were slowed because the manufacturer's sample code was not functional, forcing us to use third-party Arduino libraries. In particular, a 3rd party library, GxEPD2, was vital in making the display functional. Once the display was able to flash any kind of text or graphics, we discovered that the "flexible" e-ink display we had selected could not actually be refreshed while bent at any angle, a critical flaw that was not clear from the product's advertising. This discovery forced a complete, last-minute redesign of the mechanical enclosure to accommodate a new, rigid display. Towards the end of the project, we were able to translate most of the GxEPD2 library to the NRF52840, where the display refreshed/flashed a noisy image.

The initial bring-up of the display required extensive low-level debugging. Communication with the e-ink display driver happens over an SPI bus, but it also requires several GPIO lines to function: a Chip Select (CS) pin to initiate and terminate transactions, a Data/Command (DC) pin to differentiate between sending command bytes and data bytes, a Reset (RST) pin to initialize the display, and a BUSY pin that the display asserts to signal it is in the middle of a refresh cycle and cannot accept new commands. Our first major hurdle was getting simple SPI transactions to work. We used a logic analyzer to probe the physical lines, ensuring the voltage levels and timings of our MCU's transmissions matched the specifications in the display's datasheet. This process was

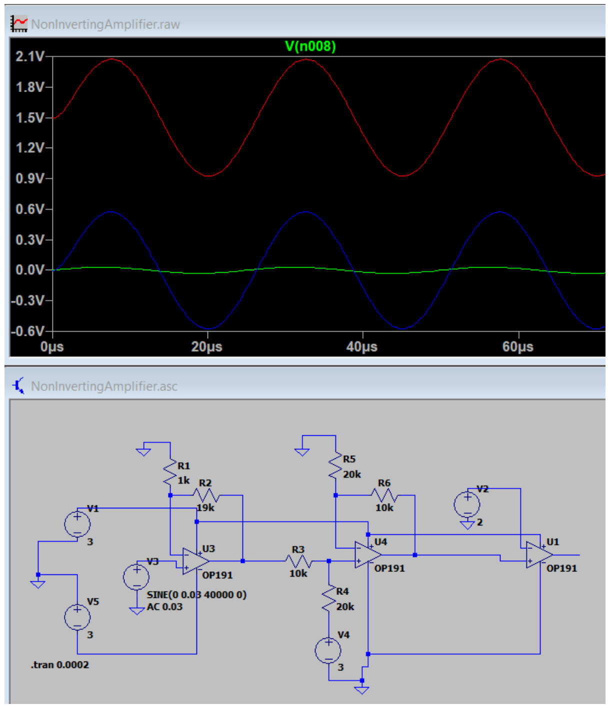

Fig. 6. Ultrasonic Transceiver Circuit, designed in LTSpice

tedious but essential for verifying that we were sending the correct sequence of commands to registers on the display controller.

The choice of development platform profoundly impacted the bring-up process. We found working example code for the display on an Arduino, which allowed us to get the screen showing text within a couple of hours. The Arduino environment and libraries like GxEPD2 abstract away much of the low-level hardware configuration. In contrast, porting this functionality to our power-efficient Nordic nRF52840 MCU running Zephyr RTOS was significantly more complex.

This complexity was compounded by a lack of clear documentation for our specific hardware combination within the Zephyr ecosystem. Zephyr uses a device tree (.overlay files) to define hardware, abstracting pin numbers and peripherals from the application code. While powerful, this abstraction layer was difficult to navigate without specific examples. The official documentation for the display was insufficient, and documentation for connecting it via Zephyr's device tree was sparse. This forced a heavy reliance on the GxEPD2 third-party library, which fortunately had internal support for the nRF architecture. Despite these hurdles, the Zephyr build system itself was sufficient for our needs. Using the west tool, we could compile the firmware with a single command, and project-specific configurations were easily managed in the prj.conf file, streamlining the overall development workflow. Towards the end of the project, we had successfully ported most of the GxEPD2 library, enabling the display to refresh, although the output remained a noisy, unrefined image.

TDS Sensor The TDS firmware performs an analog read to get the millivolts of resistance on the pin. The conductivity data can be converted using a precalibrated formula to produce the parts per million (ppm) count of particulate matter in the measured liquid.

Fig. 7. Prototype with 3D-print of custom designed lid, TDS sensor, IMU, nrf52840dk and original non-flexible E-ink display

Ultrasonic Ultrasonic sensors work by emitting inaudible chirps and measuring the time it takes to hear the sound return. Our firmware currently integrates the provided driver board with our ultrasonic sensor. We trigger a 40KHz pulse and measure how many CPU cycles elapse until the return pulse is received. We convert this distance to centimeters using a formula from the manufacturer.

IMU An IMU was required for this project to work in conjunction with the TDS sensor. Since the TDS probe was placed in the lid, the only time where the TDS probe is in contact with the liquid is while the water bottle is flipped upside down. As a result, an IMU was used to detect when the TDS sensor is in contact with the liquids. The accelerometer readings were read off the IMU via I2C transactions [2]. When the accelerometer recorded an acceleration due to gravity in the opposite direction of upright, the firmware deduced that the bottle was currently upside down.

BLE Advertisements Initially, the project scope included full Bluetooth Low Energy (BLE) connectivity to a companion smartphone application for health data synchronization. We ultimately decided the user experience and battery life would be diminished by forcing users to use a companion app in addition to the integrated screen.

We found that Bluetooth support was a useful feature for integration testing. The Bluetooth firmware uses the Zephyr APIs to create a Bluetooth service. We use the NRF Connect mobile app to read the advertised Generic ATTribute Profile (GATT) characteristics. We implement a Bluetooth service with characteristics to read debug output of sensor data. These Bluetooth characteristics were useful for wirelessly reading sensor data, enabling us to use our battery powered development board without tethering the board to a laptop to read UART output.

4 Milestones

MILESTONE	**STATUS**
Completed Milestones	
Initial Bottle Lid Designed in CAD + 3D Printed	Complete
MCU & Sensors Selected and Sourced	Complete
Firmware Drivers Prototyped (TDS, IMU)	Complete
E-ink Display Working (with text)	Complete
Display UI Designed & Sketched	Complete
MVP Complete (using dev kits)	Complete
Final 3D Print Model Finalized	Complete
PCB Designed & Ordered	Complete
Final Prototype Complete & Tested	Complete
Final Video, Presentation, & Report Complete	Complete
Incomplete & Abandoned Milestones	
Firmware for BLE Connection to Smartphone	Abandoned
PCB Assembled & Integrated into Bottle	Incomplete
Food Safe Bottle Version Complete	Incomplete
Full UI Graphics	Incomplete

The TONIQ project successfully navigated a complete development cycle, from initial conception to a fully integrated hardware prototype. All primary objectives for creating a functional device were met. This included the initial mechanical design of the bottle lid in CAD software and its subsequent 3D printing, the selection and sourcing of all microcontrollers and sensors, and the creation of a bill of materials. On the firmware side, drivers for the core sensors, including the TDS and IMU, were written and prototyped. A significant effort was dedicated to the e-ink display, successfully progressing from initial bring-up to displaying text on screen. The overall user interface was also designed and sketched out. These individual components were brought together to create a minimum viable product using development kits, which proved the concept and led to the design and ordering of a custom PCB. Ultimately, the project culminated in a finalized 3D printed model with the assembled PCB integrated inside, and all final course deliverables were completed.

In order to achieve these core goals, the project's scope was pragmatically managed, leading to several features being either de-scoped or left incomplete. The plan to write firmware for a Bluetooth LE connection to a smartphone application was explicitly abandoned; this decision was made to focus development efforts on the self-contained functionality of the lid itself, as a companion app was considered outside the scope of the class. Similarly, while a path to manufacturing a food-safe version was planned using materials like injection-molded, BPA-free polypropylene, the final prototype did not meet this standard, leaving this milestone incomplete. Lastly, while the UI was designed and the display could render text, the final implementation of polished graphics integrating all sensor readings was not fully realized in the final prototype.

5 Conclusion

This project successfully produced a functional prototype of TONIQ, a smart water bottle lid designed to address key flaws in existing commercial products. By integrating all sensing and display technology into a modular lid, TONIQ provides a flexible, app-independent user experience that is compatible with standard wide-mouth water bottles. Our primary contributions are the novel integrated design, which combines a custom PCB, multiple sensors, and a low-power display within a compact, 3D-printed enclosure, and the firmware to operate it.

Throughout the project, we navigated significant mechanical, electrical, and software challenges, from iterating on the complex lid design to overcoming unexpected hardware failures like the non-functional flexible display. Our work demonstrates the feasibility of a self-contained smart lid that tracks hydration and water quality.

References

[1] [n. d.]. *Amazon.com: SGUAI Unique Smart Water Bottle - Smart Coffee Cup with LED Temperature Pixel Display by APP Control, Smart Insulated Novelty Mug Water Flask for Unique Presents, 12oz (White): Home Kitchen.* https://www.amazon.com/SGUAI-Unique-Smart-Water-Bottle/dp/B0C16D39LY?th=1

[2] [n. d.]. *MPU6050 6-DoF Accelerometer, Gyro Guide Datasheet.* https://www.digikey.com/en/htmldatasheets/production/5050648/0/0/1/3886#pf6

[3] [n. d.]. *nRF Connect for VS Code.* https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-for-VS-Code

[4] [n. d.]. *nRF52840 DK.* https://www.nordicsemi.com/Products/Development-hardware/nRF52840-DK

[5] [n. d.]. *Trago - The World's First Smart Water Bottle.* https://www.kickstarter.com/projects/905031711/trago-the-worlds-first-smart-water-bottle

[6] [n. d.]. *West.* https://docs.zephyrproject.org/latest/develop/west/index.html

[7] [n. d.]. *What is UVC?* https://silannauv.com/what-is-uv-c/

[8] [n. d.]. *Zephyr Project.* https://www.zephyrproject.org/

[9] 2015. *Introducing Trago the World's First Smart Water Bottle Support this revolutionary hydration product on Kickstarter now.* https://www.prnewswire.com/news-releases/introducing-trago----the-worlds-first-smart-water-bottle----support-this-revolutionary-hydration-product-on-kickstarter-now-300092726.html

[10] 2015. *U.S. Patent for Liquid consumption tracker Patent (Patent 10,161,782 issued December 25, 2018) - Justia Patents Search.* https://patents.justia.com/patent/10161782

[11] 2019. *IMU Sensors: Everything You Need To Know!* https://embeddedinventor.com/what-is-an-imu-sensor-a-complete-guide-for-beginners/

[12] 2023. *How TDS meters work (The Ultimate Guide) - AquaHow.* https://aquahow.com/how-tds-meters-work/

[13] Emil Jovanov, Vindhya R Nallathimmareddygari, and Jonathan E. Pryor. 2016. SmartStuff: A case study of a smart water bottle. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).* 6307–6310. doi:10.1109/EMBC.2016.7592170

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009