

Pumped

Authors:

Mya Camacho-Hill | Computer Engineer

Simran Aulakh | Electrical Engineer

Alexandra Atuncar | Computer Engineer

Johan Gales | Electrical Engineer

Victoria Rodriguez | Sponsor

Matthew Aberman | Sponsor

Reviewer Committee:

Suboh Suboh | ECE

Dr. Nazanin Rahnavard | ECE

Mark Lundstrom | WUCF

University of Central Florida

Department of Electrical & Computer Engineering



CHAPTER 1 EXECUTIVE SUMMARY.....	1
CHAPTER 2 PROJECT DESCRIPTION.....	2
2.1 PROJECT BACKGROUND AND MOTIVATION	2
2.1 EXISTING PRODUCTS.....	3
2.1.1 <i>Arduino Weighing Scale</i>	3
2.1.2 <i>Zebra Printers</i>	3
2.1.3 <i>Cubtale Smart Milk Inventory Tracking</i>	3
2.2 GOALS AND OBJECTIVES	4
2.3 DESCRIPTION OF FEATURES	5
2.4 ENGINEERING SPECIFICATIONS	6
2.5 HOUSE OF QUALITY	7
2.6 HARDWARE BLOCK DIAGRAM	8
2.7 SOFTWARE FLOWCHART.....	9
2.8 PROTOTYPE ILLUSTRATION.....	10
CHAPTER 3 RESEARCH AND INVESTIGATION	10
3.1 RELEVANT TECHNOLOGY.....	10
3.1.1 <i>Huckleberry</i>	10
3.1.2 <i>Baby Tracker</i>	11
3.1.3 <i>Nara Baby</i>	11
3.2 MICROCONTROLLER.....	11
3.2.1 <i>MSP430</i>	11
3.2.2 <i>ARDUINO UNO R3</i>	12
3.2.3 <i>RTK7RLG220C00000BJ</i>	12
3.2.4 <i>ESP32</i>	12
3.2.5 <i>STM32F411</i>	12
3.2.6 <i>Raspberry Pi Pico</i>	13
3.2.7 <i>Microcontroller Part Selection</i>	14
3.3 SCALE ARCHITECTURE.....	14
3.3.1 <i>Strain Gauge Load Cell Scales</i>	14
3.3.2 <i>Electromagnetic Force Restoration Scales</i>	14
3.3.3 <i>Capacitive Digital Scales</i>	15
3.3.4 <i>Hydraulic or Pneumatic Scales</i>	15
3.3.5 <i>Piezoelectric Digital Scales</i>	15
3.3.6 <i>Smart and IoT Connected Scales</i>	15
3.3.7 <i>Scale Part Selection</i>	15
3.4 WEIGHT SENSORS	16
3.4.1 <i>Load Cells</i>	16
3.4.1.1 <i>Single-Point Load Cells:</i>	16
3.4.1.2 <i>Bar-Type Load Cells</i>	16
3.4.1.3 <i>S-Type Load Cells</i>	17
3.4.1.4 <i>Compression Load Cells</i>	17
3.4.1.5 <i>Load Cell Part Selection</i>	19
3.4.2 <i>Load Cell Amplifier</i>	19
3.4.2.1 <i>HX711 Load Cell Amplifier</i>	19
3.4.2.2 <i>ADS1232 Load Cell Amplifier</i>	19

3.4.2.3	NAU7802 Load Cell Amplifier	20
3.4.2.4	Analog Load Cell Amplifier	20
3.4.2.5	Load Cell Amplifier Part Selection	21
3.5	PRINTER	22
3.5.1	Zebra ZQ620 / ZQ520.....	22
3.5.2	Brother RJ-2035B.....	22
3.5.3	Bixolon SPP-R310	23
3.5.4	MUNBYN ITPP941	23
3.5.5	Printer Part Selection.....	23
3.6	DISPLAY SCREEN	24
3.6.1	OLED (Organic Light Emitting Diode).....	24
3.6.2	LCD (Liquid Crystal Display).....	24
3.6.3	Electronic Paper Display	25
3.6.4	Multi-Segment LED Displays.....	26
3.6.5	LCD Part Selection.....	27
3.7	LEDs.....	27
3.7.1	Standard Through-Hole LEDs	28
3.7.2	Surface-Mount Device LEDs	28
3.7.3	Bi-color or Tri-color LEDs	28
3.7.4	WS2812B	28
3.7.5	LED Part Selection.....	29
3.8	POWER SUPPLY.....	30
3.8.1	Primary (Non-Rechargeable) Batteries.....	30
3.8.2	Alkaline Batteries	30
3.8.3	Zinc-Carbon Batteries.....	30
3.8.4	Silver Oxide Batteries	30
3.8.5	Secondary (Rechargeable) Batteries.....	31
3.8.6	Lithium-Ion Batteries.....	31
3.8.7	Lead-Acid Batteries.....	31
3.8.8	Nickle-Metal Hydride Batteries	31
3.8.9	Solar Power	32
3.8.10	Supercapacitors.....	32
3.8.11	Battery Part Selection	33
3.8.12	Part Selection Battery Charging IC/Module.....	34
3.9	VOLTAGE REGULATORS.....	34
3.9.1	LM2576	35
3.9.2	LM2679	35
3.9.3	LM2596	35
3.9.4	MP1584	35
3.9.5	XL4015.....	36
3.9.6	XL6009.....	36
3.9.7	MT3608.....	36
3.9.8	AMS1117.....	36
3.9.9	LM7805	36
3.9.10	Voltage Regulator Selection	37
3.10	WIRELESS COMMUNICATION.....	38
3.10.1	Bluetooth	38
3.10.2	Wi-Fi	38

3.10.3 Zigbee	38
3.10.4 Ultra-Wideband	38
3.10.5 Near Field Communication (NFC)	38
3.11 MICROCONTROLLER INTERFACES	39
3.11.1 PWM	39
3.11.2 ADC/DAC	39
3.11.3 Serial Communication Interfaces	39
3.11.4 UART (Universal Asynchronous Receiver Transmitter)	39
3.11.5 SPI (Serial Peripheral Interface)	39
3.11.6 I2C (Inter-Integrated Circuit)	40
3.11.7 Two Wire Automotive Interface	40
3.12 SOFTWARE RESEARCH	41
3.12.1 Mobile Operating Systems	41
3.12.2 Apple iOS	41
3.12.3 Google Android	42
3.12.4 HarmonyOS	42
3.12.5 KaiOS	42
3.13 PROGRAMMING LANGUAGES	43
3.13.1 C	43
3.13.2 Python (MicroPython)	43
3.14 MOBILE APPLICATION PROGRAMMING LANGUAGES	43
3.14.1 Java/Kotlin	43
3.14.2 Swift	44
3.14.3 React Native (JavaScript/Typescript)	44
3.14.4 Flutter (Dart)	44
3.15 CODE REPOSITORIES	45
3.15.1 GitHub	45
3.15.2 Bitbucket	46
3.16 DATABASES	46
3.16.1 SQLite	46
3.16.2 MySQL Database	46
3.16.3 Firebase Realtime Database	46
3.16.4 MongoDB	47
3.16.5 PostgreSQL	47
3.16.6 Database Selection	48
3.17 3D MODELING SOFTWARE	48
3.17.1 Tinkercad	48
3.17.2 Fusion 360 (Autodesk)	48
3.17.3 SolidWorks	49
3.17.4 FreeCad	49
3.17.5 OnShape	49
CHAPTER 4 STANDARDS AND DESIGN CONSTRAINTS	49
4.1 STANDARDS	49
4.1.1 Electrical and Power Standards	49
4.1.1.1 PCB Standards	50
4.1.2 Battery and Power Standards	51
4.1.3 LED Standards	52

4.1.3.1 IEC 62031	52
4.1.4 Wireless Communication Standards	52
4.1.5 Breastmilk Storage Standards	53
4.1.6 Security and Software Protocols	54
4.1.6.1 ISO/IEC 27001 (Information Security Management)	54
4.1.6.2 ISTQB Software Testing Standard	54
4.2 CONSTRAINTS	56
4.2.1 Time Constraints	56
4.2.2 Ethical and Social Constraints	56
4.2.3 Economic Constraints	57
4.2.4 Manufacturability Constraints	58
4.2.5 Realistic Design Constraints	58
CHAPTER 5 COMPARISON OF CHATGPT WITH OTHER SIMILAR PLATFORMS.....	58
5.1 BENEFITS	59
5.1.1 Specificity	59
5.1.2 Inspiration	59
5.1.3 Multimedia	59
5.2 DISADVANTAGES	60
5.2.1 Environmental Concerns	60
5.2.2 Inaccuracy	60
5.2.3 Academic Misuse	60
5.3 EXAMINING AI USAGE FOR OUR PROJECT	60
5.3.1 Case Study 1	60
5.3.2 Case Study 2	61
5.3.3 Case Study 3	62
CHAPTER 6 HARDWARE DESIGN.....	62
6.1 BREADBOARD DEMONSTRATION	62
6.2 SUBSYSTEM BLOCK DIAGRAM	63
6.3 CONVERTER SCHEMATICS	63
6.3.1 AC/DC Converter	64
6.3.2 3.3V DC/DC Buck Converter Schematic	65
6.4 MCU SCHEMATIC (USB TO UART BRIDGE)	67
6.5 TOUCHSCREEN SCHEMATIC	68
6.6 WEIGHT SENSOR SCHEMATICS	69
6.6.1 Load Cell Amplifier Schematic	69
6.6.2 Load Cell Schematic	70
6.7 LED SCHEMATIC	71
.....	71
6.8 PRINTER	71
CHAPTER 7 SOFTWARE DESIGN	72
7.1 MOBILE APPLICATION DEVELOPMENT	72
7.1.1 Mobile Application	73
7.2 DEVICE SOFTWARE	78
7.2.1 Load Cell Subsystem	78
7.2.2 LED Subsystem	79

7.2.3 Touchscreen Subsystem	79
7.2.4 Printer Subsystem.....	81
CHAPTER 10 ADMINISTRATIVE CONTENT	82
10.1 BUDGET	82
10.2 BILL OF MATERIALS	83
10.3 DISTRIBUTION OF WORK	83
10.4 MILESTONES	84
APPENDIX A: REFERENCES.....	86
APPENDIX E: CHATGPT PROMPTS & OUTCOMES.....	89

Chapter 1 Executive Summary

Breastfeeding mothers face many challenges including ensuring proper labeling of milk storage to having to keep track of milk inventory to avoid waste. The traditional method of manually labeling milk bags with a sharpie is both time-consuming and subject to errors, especially after late night pumping sessions. This senior design project aims to transform breastmilk management for mothers by developing a breastmilk scale and labeler that simplifies the tracking and storage process. Our device integrates an accurate digital scale, automatic thermal label printer, and an accompanying mobile application, offering an innovative solution for mothers, caregivers and healthcare professionals.

What sets our project apart from existing solutions is its integration of hardware and software, designed with user convenience in mind. The device will accurately measure the volume of milk in the storage bags and print a label containing essential details such as date, time, volume, and optional dietary specifications. The device further enhances user convenience by syncing the data to the mobile application via Bluetooth after the printing of the label is completed. Furthermore, the mobile application increases usability by allowing users to manage milk inventory, expiration dates, and feeding patterns, reducing the risk of wasted milk.

Portability is also a key feature of our design, ensuring mothers can use the device both on-the-go and at home. A battery-powered structure allows for more flexibility in various environments while LED indicators provide real-time status updates, such as low battery alerts. Beyond individual benefits, our system also promotes milk donation by simplifying the labeling of dietary restrictions and allergens. This makes it easier for milk banks and hospitals to distribute donated milk to NICU babies and families in need. This device not only enhances convenience but also strengthens the support system for breastfeeding mothers.

This report documents the full design process, including system objectives, electrical integration, and both firmware and mobile application development. It follows the progression from prototype development to final implementation, incorporating rigorous testing to evaluate performance and refine the system based on results. By utilizing engineering knowledge with user-centered design, our project empowers mothers with an efficient, accurate, and stress-reducing solution for breastmilk management.

Chapter 2 Project Description

2.1 Project Background and Motivation

In the realm of innovative engineering, our senior design project seeks to address a challenge experienced by mothers everywhere. New mothers generally resort to breastfeeding as opposed to formula, as natural breast milk contains more nutrients and correlates to better immune system responses as the baby matures. Mothers aim to build a milk stash so that when they start work or travel, the baby can still receive the same amount of nutrients without reverting to formula. For pumped milk, it is important to label the date, time, and amount collected to ensure that it is consumed in a timely manner and does not expire or waste. While pumping milk for later use can save time and money, the process can be taxing. Currently the only way to label milk bags is with a sharpie and manually checking the number of ounces when pouring into the storage bag as well as checking for the latest date and time. Although it may just seem tedious this process usually occurs late at night or early in the morning when the mother is tired. This can lead to errors when labeling, cause added stress and be an inconvenience for the mother. Our project consists of creating an efficient system for tracking and properly labeling breastmilk storage bags. This system aims to combine practicality with ingenuity, empowering mothers with a solution that saves time, reduces stress and waste, as well as enhances their caregiving experience.

Our sponsors, Victoria Rodriguez and Matthew Aberman, approached our section and presented the idea of creating a device that would help with efficiently tracking breastmilk bags. After learning more about this idea, our group recognized the potential of this project to make a meaningful difference in the lives of mothers everywhere. Being able to save even a few minutes each day allows the mothers to feel some relief knowing that the milk has been accurately labeled, reducing guilt or anxiety when feeding their child. Victoria expressed that being able to keep track of how much milk their baby consumes is key when trying to avoid over and under feeding. She had shared that when many mothers are ready to go back to work after giving birth the process can be more complicated due to their breastmilk schedules overlapping with professional responsibilities. Having a portable device to take on the go and an application that keeps track of her milk supply provides a comforting feeling when a caretaker is with her child. Beyond alleviating a small portion of the stress for mothers, the product aims to call to mothers that are able to produce an excess amount of milk, to consider donating to milk banks or hospitals. With the ability to easily label dietary restrictions and allergens on the milk it can easily be distributed for NICU babies, fosters, and families where the mother cannot produce sufficient amounts of milk.

This present difficulty not only requires technical skills but creativity and empathy to ensure the final product meets the needs of its users seamlessly. We strongly believe in this idea, knowing its immense value to future mothers, and are taking on this project to create something beneficial for women everywhere. Victoria and Matthew are both engineers which interested us as well knowing they can provide helpful feedback and new insights on the direction of our project.

Our project integrates software and hardware elements which will challenge us to work cohesively together as a team. Guided by the feedback and expertise of our sponsors we aim to not only create a solution that meets expectations but exceeds them. Through this venture we hope to contribute to a brighter, more supportive future for mothers, families, and caretakers.

2.1 Existing Products

2.1.1 Arduino Weighing Scale

For our hardware design our main goal is to create a scale that can be easily integrated with our software needs. After much research we have found a DIY Arduino weighing scale project by a user on the website “Instructables.” It gave us valuable insight into the components we need for our own project as well as their specific usages. In this project they use both an Arduino and an HX711 module. For their Arduino they are using a nano, which they recommend because of its low weight and its built-in USB interface. Our aim is to keep our project portable, so we will likely be using the same Arduino as them. They integrate the HX711 by connecting to the Arduino to share the weight’s data. This processed data shows up on the LCD display of the scale. Since this is similar to our scale concept, we will be using their build as a reference for that portion of our design.

2.1.2 Zebra Printers

Zebra printers are highly versatile in their usage. They have incredible utility and are an immensely popular choice for businesses. In our design, we are aiming to print the date, time, weight, and possible dietary restrictions on our labels. Zebra printers are able to combine pre-printed labels with the aforementioned dynamic data. They also provide their own software development kits (SDKs), which are immensely useful for our own software integration. They provide things such as APIs, libraries, and documentation that we can use for our design. This would make it easier to program our microcontroller of choice to transmit commands to the printer.

Zebra Technologies also offers breast milk tracking barcode labels. The details include date, time, volume, and expiration. This form of labeling accommodates other caregivers by letting them access this information with a simple scan. It also ensures that the milk is handled properly based on its specifications. Barcode labeling is an element we are considering, especially because of its accessibility to caregivers.

2.1.3 Cubtale Smart Milk Inventory Tracking

Cubtale is a company that offers innovative solutions to inconveniences mothers face on a regular basis. One of such solutions is a product called “Smart Milk Inventory Tracking – Labels”. These labels include a barcode along with manually recorded details such as the date, time, volume, name, and milk bag number. Users are able to enter these details into the accompanying app using the corresponding label number. The app offers features such as feed reminders, which tell the user which milk bag to use, as well as expiration date notifications. Within the application, there is information on breastmilk storage guidelines which tells users how to properly store their breastmilk to prevent premature expiration. The user can also track how much milk is available in their inventory. There are also instructions that provide guidance to caregivers by informing of how to feed the baby, including warming instructions for the milk.

Beyond milk tracking, Cubtale also offers 'Cubtale Cubs,' smart devices that are designed to track daily activities. They can track sleep, diaper changes, personal care, medicine, and pump use. When you press a specific button on the Cub, it then logs the activity. If there is no Wi-Fi, the Cubs can store the latest five records in its memory. This is a possible function we can

implement in our scale in case our Bluetooth connection fails. Another positive feature of this device is that they only connect to Wi-Fi for a few seconds, which lessens radiation exposure significantly for the child. We would also like to limit this exposure by implementing this aspect. After the button is pressed, the information is sent to the application, and the user is notified of the activity being completed. This is an excellent feature for users who have other caregivers looking after their child, and it gives them peace of mind knowing their routine is being followed.

Our project will incorporate several of these key functionalities. We intend to develop software that provides expiration date reminders and suggests which milk bag to use based on past data. Labeling is also a critical aspect of our design, and incorporating a barcode system for seamless data logging is a strong consideration. Its capabilities will serve as a strong reference for our own design.

2.2 Goals and Objectives

Our goals are organized into three tiers: basic, advanced, and stretch. The basic goals focus on fundamental functions to assist mothers in breastmilk management. The advanced goals expand on this foundation by adding more interactive features to the device. Stretch goals further enhance the system by integrating data-driven insights and multi-user accessibility.

Goals:

Basic Goals:

- To help mothers manage pumped breastmilk
- Design a device that measures pumped breastmilk
- Have the device print a label for breastmilk
- Creating a user-friendly interface for mothers to access information

Advanced Goals:

- Optimize design for portability
- Offer customizable labeling
- Help monitor milk inventory
- Design a physical guide system to minimize milk spillage.
- Provide visual feedback on system status

Stretch Goals:

- Provides insights to help mothers make informed dietary decisions.
- Implement multi-user access with varying levels of permission.
- Enhance software for integration with milk banks.

Objectives:

To enable accurate tracking and management of breast milk storage, the device will be used to measure and log milk volume after each pumping session. A built-in sensor will determine the exact volume of milk with consideration for the weight that may come from the packaging used for the milk. Labels will include date, time, & weight received from the scale, with optional input for allergens or dietary restrictions. Using a pre-built printer, a label will be printed with the data collected from the scale. Advanced labeling options will allow users to enter additional details such as donation specific fields for more personalization. The system will utilize labels that

adhere to milk bank and hospital requirements. Then the user-interface will allow users to manually edit details in the mobile application, if needed.

To prioritize portability, the device will be designed with compact and lightweight hardware components, which will enable mothers to carry it while traveling or pumping on the go. In addition, the shape of the scale will be ergonomic, allowing it to support the bag of breastmilk to minimize spillage. A battery-operated structure ensures this convenience without sacrificing its functionality, and the implementation of LEDs will provide visual feedback in the case of low battery.

The software that will be accompanying this device will manage inventory, track expiration dates, and notify users of inventory issues; this ensures the efficient and safe use of breastmilk. These features will be provided through a user-friendly app, allowing mothers and caregivers to make informed decisions regarding the child's intake. Additionally, the app will analyze patterns from the data entered such as feeding frequency, baby's age, and amount stored in inventory to help mothers anticipate future milk needs. By integrating accurate hardware measurements with software capabilities, this system aims to provide a solution that meets the needs of breastfeeding mothers and their support networks.

2.3 Description of Features

Our sponsor has provided us with a list of primary user jobs and requirements for this project. In our design, we are aiming to solve the problems presented to us for breastfeeding moms as well as caregivers and healthcare professionals.

Breastfeeding moms must not only mentally track but label all of the milk they produce for their child. This can become tricky, as sometimes mothers lose track of when they have produced the milk and how much of it has been produced as well. With the scale feature, we aim to fix this problem. In our design we would like to create a scale that can weigh the volume of the milk, and then print the date, time, and weight on a label. Sometimes mothers must make sure they label the bags based on allergy specifications, such as writing if it is dairy-free. In our design, we want to add a feature that prints these specifications on the same label.

In collaboration with the device, we are looking to develop an app that will make keeping track of milk inventory seamless. Utilizing the device after entering the proper information and printing the label, the device will then log the new quantity of milk into the inventory stored in the app. Through this app users can edit the details of the milk inventory as well as delete from it as the milk is being used. Other features of this app would include the ability to receive notifications when milk is about to expire and check-ins for if any milk was properly logged out of inventory. This application will be accessible at the mother's discretion where she can create secondary accounts for caregivers that will allow them access to her data but provide restrictions for that user's access too. Since this app is not only being designed for mothers as users it will have a feature that will calculate the necessary amount of milk needed for feeding based on user input. Utilizing this information, it will also be able to determine when more milk is necessary to meet feeding needs. The user would be able to set a threshold of when they want to be notified of when inventory is getting low. Every baby's feeding schedule is different so this feature will help to account for that.

2.4 Engineering Specifications

At this time, we are currently making informed estimates on our values based on research of previous projects/products. For our testable specifications, we are planning on testing weighing accuracy, LEDs, and display.

Table 1: Prioritized Specifications

Parameter	Description	Value
Weight	Weight of product	1-3 lbs.
Dimension	Total volume of product must adhere to portability requirements	10 x 5 x 3 inches
Storage Capacity	Total logs of milk bags that can be stored in data inventory	1200 Entries
Battery Life	Rechargeable battery must charge scale and printer	6-8 Hours
Weighing Accuracy	Feedback from scale must be accurate	$\pm 0.005-0.02$ oz
Maximum Weight Capacity	Max amount of weight the scale can hold, must account for breastmilk bags of larger volumes	1-3 lbs.
Dimension of Labels	FDA compliant, has date/time/dietary restrictions	2 x 3 inch labels
Wireless Connection Range	Range of Bluetooth connection between product and mobile device	30 feet
Software Response Time	Response time to see data from hardware device to mobile application	< 5 seconds
Printing Time	Time it takes to print one label	< 5 seconds

2.5 House of Quality

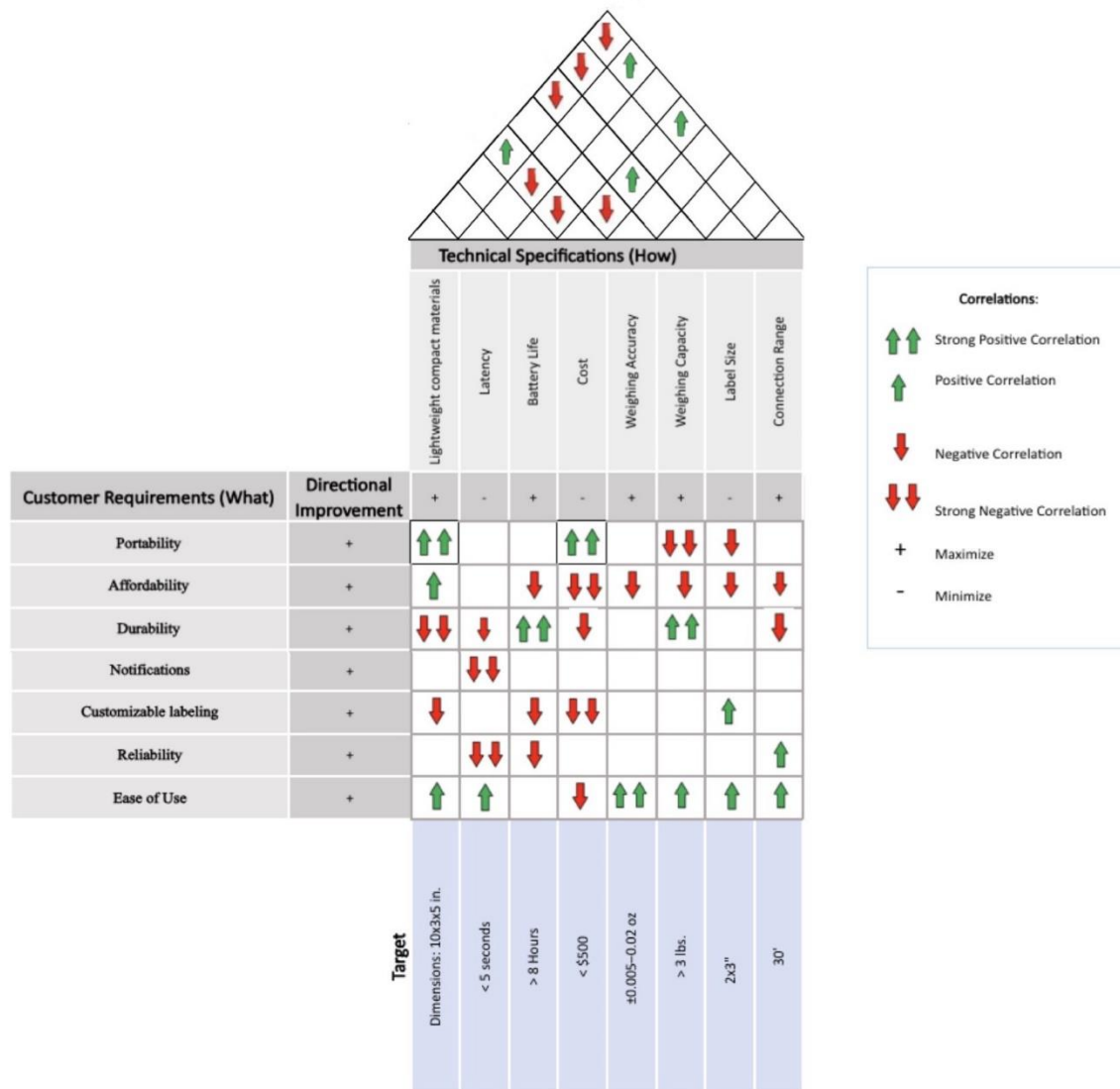


Figure 3: House of Quality

2.6 Hardware Block Diagram

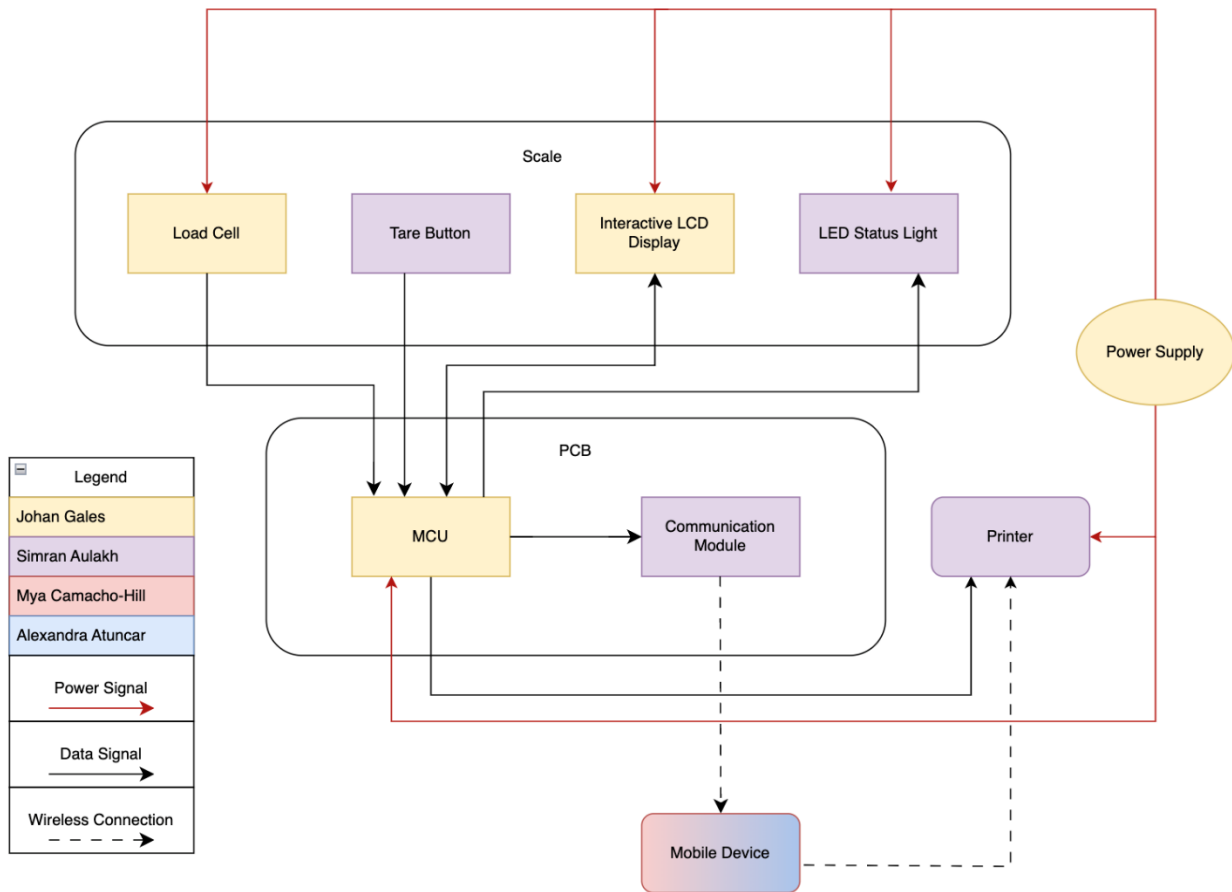


Figure 1: Hardware Block Diagram

For our prototype, we plan to implement all circuitry on a custom-designed printed circuit board (PCB) to streamline the integration and reliability of our device. The PCB houses key components including the microcontroller (MCU), load cells for weight sensing, a tare button, and the communication interface that connects to the label printer. The hardware block diagram shown here provides a high-level overview of how major subsystems interact. Each component is visually grouped to reflect its role in the overall system. This block-based framework helps clarify the flow of power and data signals throughout the device, making it easier to conceptualize, troubleshoot, and refine each subsystem independently before moving on to physical PCB design.

2.7 Software Flowchart

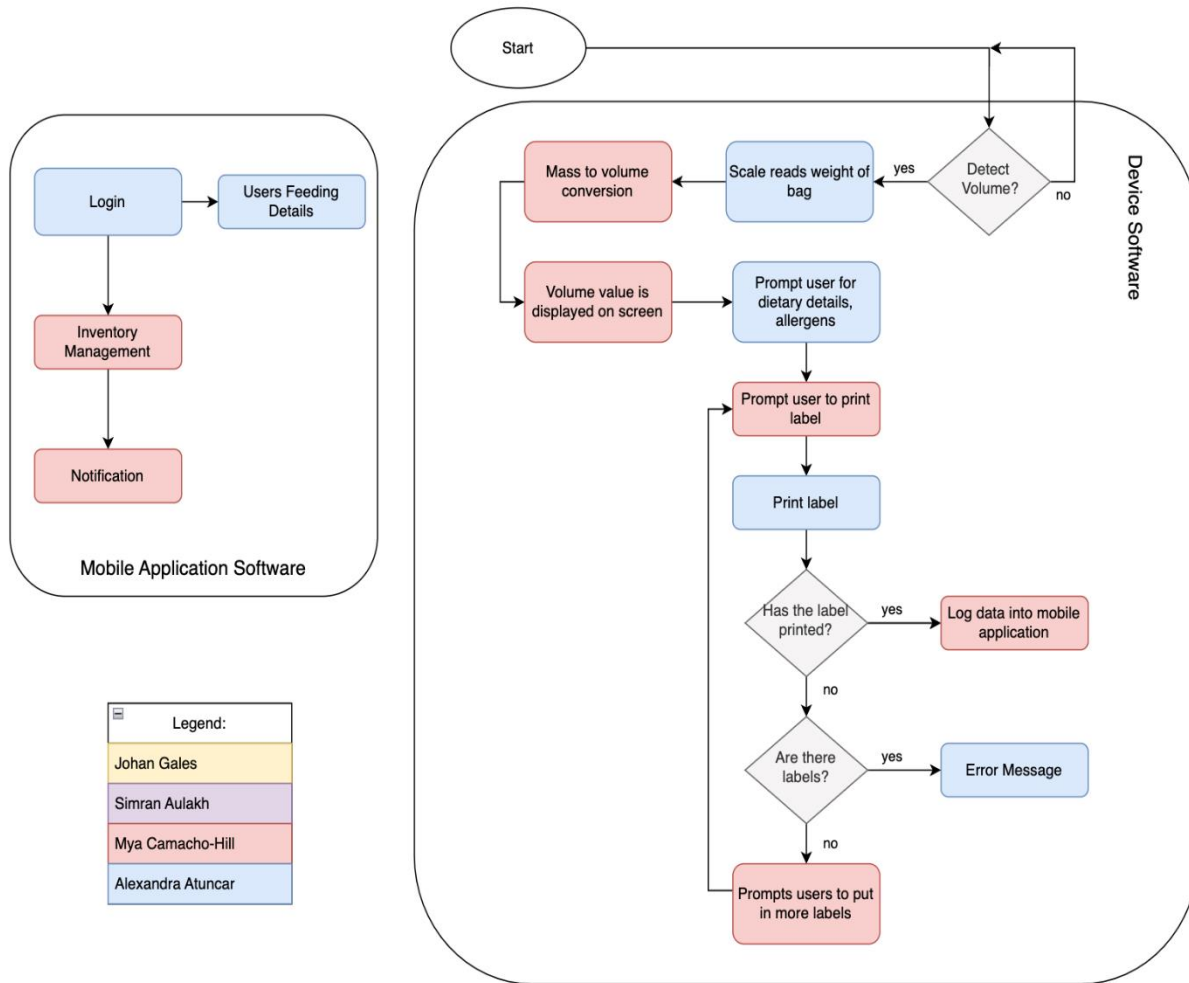


Figure 2: Software Flowchart

Our software flowchart illustrates the process by which the device captures, labels, and logs breastmilk data. The device's software begins by detecting the presence of a milk bag and reading its weight. It then converts this weight into volume, prompting the user for any additional dietary information or allergen notes, and generates a label for printing. Once the label is successfully printed, the data is logged into the mobile application. The mobile app enables users to manage their milk inventory, track storage and expiration, and enter child specific feeding preferences. Additionally, it provides notifications to remind users of important actions, such as low inventory, or expired milk. This integrated system supports both the accurate tracking of breastmilk and the personalization of care for each child's dietary needs.

2.8 Prototype Illustration



Figure 4: Prototype Illustration

Chapter 3 Research and Investigation

3.1 Relevant Technology

Currently there are a number of apps that act as data trackers for mothers to use in various aspects of their lives. In relation to our project specifically, some of these apps include inventories that can be used to keep tabs on available stored breastmilk as well as pumping times. A few of them are described below:

3.1.1 Huckleberry

Huckleberry is an all-around tracker app for various aspects relating to babies/mothers including sleep, feeding, bathroom use, pumping, medicine, growth, and activities. Under each of these sections, mothers can record the times of the day that the baby has done the category (slept, peed, ate, etc.), with additional prompts for the quality of these actions, other descriptors, and personal notes. For every time-based activity, the data recorded is then displayed in a weekly schedule with blocks for the duration of each activity. There are also tabs for activity overviews by day, in a list, or as a summary.

Relevant to our project is the pumping category that allows users to log how much milk is being pumped (able to be distinguished further between left and right breast or as a total), by ounce or by milliliter, the time it began being pumped, and the duration of the pump session. This data is then recorded on the same schedule as previously mentioned, however it cannot access or track specific bags of the milk being pumped – only the time and amount that was pumped. While these features are undoubtedly useful to mothers, the more important aspect for our project is keeping track of how much milk is available to the mother, as well as when it can be used by and

stored. The Huckleberry program is also only available through a monthly or annual subscription that begins with a 14-day free trial.

3.1.2 Baby Tracker

Baby Tracker holds many similarities to Huckleberry in its ability to track and monitor several aspects of parenting, including feeding (nursing, formula, supplement), diaper use, sleep schedule, physical growth, milestones, and more. As it relates to our project, the app allows mothers to track their pumped breastmilk: the feature includes a toggle that switches between live timed logging or retroactive logging, as well as the ability to distinguish between left and right breast pumping. The collected milk is logged in ounces, and each log automatically updates the overall amount in the inventory. In another tab, the activity logs can be viewed by day, with options to make a note about each activity (or pump) and include facts like allergens or other warnings. The app is completely free and utilizes a very minimalist style that is easy to follow and understand.

3.1.3 Nara Baby

Nara Baby holds many of the same capabilities of the previously mentioned tracker apps with a few notable differences: firstly, the specific fields of activity tracking can be toggled on or off, curating the app experience to better suit the parent's needs. This allowed us to fixate purely on the breastmilk aspect, ignoring all the other activities irrelevant to our project. The second notable difference from the previous two apps was the ability to add caretakers & join existing family accounts, a feature that is highly relevant to our goals with the project.

While these apps have an overlap with the software aspect of our project, we aim to further integrate the physical labelling and weighing of the breastmilk being collected through our hardware, creating a device that will further facilitate the pumping process for mothers in conjunction with an app for inventory. At the very least, these apps can help inform the structure of our app and software.

3.2 Microcontroller

3.2.1 MSP430

The MSP430 is a 16-bit microcontroller from Texas Instruments, known for its low power consumption, making it ideal for battery-operated and portable devices. It features multiple low-power modes for enhanced energy efficiency. Additionally, it includes a range of integrated peripherals, such as timers, UART, SPI, and I2C, along with analog-to-digital converters (ADC) that make it highly useful for sensor interfacing.

However, the MSP430 has lower processing power compared to more modern microcontrollers, which is an important consideration. Its programming complexity is also higher than some other options. While it would be well-suited for our project's power efficiency and weight measurement requirements, it may struggle with handling a touchscreen interface or controlling a label printer due to its limited processing capabilities and increased development complexity.

3.2.2 Arduino UNO R3

The Arduino UNO R3 is an 8-bit microcontroller based on the ATmega328P. It is widely used for prototyping due to its large community support and ease of use. Its Integrated Development Environment (IDE) is beginner-friendly, making development and troubleshooting more accessible. Additionally, it offers a large number of I/O pins, supporting UART, SPI, and I2C for peripheral communication.

That being said, as an 8-bit microcontroller, it has limited processing power. It also lacks built-in Wi-Fi or Bluetooth, which may require external modules for wireless communication. With only 2KB of RAM and 32KB of Flash memory, it is unsuitable for running a touchscreen interface or handling complex tasks. While it could be a viable option for prototyping, it is unlikely to be suitable for the final design of our project.

3.2.3 RTK7RLG220C00000BJ

The RTK7RLG220C00000BJ (Renesas RL78/G22 Kit) is a 16-bit microcontroller development kit designed for ultra-low power applications, making it highly suitable for portable devices. It features integrated capacitive touch sensing and supports multiple communication interfaces, including I2C, SPI, and UART. Additionally, its 16-bit CPU provides high efficiency, making it a strong candidate for power-sensitive applications.

On the other hand, it comes with increased complexity when integrating peripherals, which may require additional development effort. Additionally, it lacks built-in Wi-Fi or Bluetooth, meaning external modules would be needed for wireless connectivity.

3.2.4 ESP32

The ESP32 is a 32-bit microcontroller with built-in Wi-Fi and Bluetooth, making it a strong choice for applications requiring wireless communication. Its dual-core processor enables it to efficiently handle tasks such as weight measurements, touchscreen UI, and data transmission. Additionally, it offers a large number of GPIO pins, allowing for extensive peripheral integration.

Compared to the MSP430 or RL78/G22, the ESP32 consumes more power, which may be a consideration for battery-operated designs. However, its processing power and connectivity features make it a well-rounded option for our project.

3.2.5 STM32F411

The STM32F411 is a high-performance microcontroller that provides greater processing power compared to options like the MSP430 or Arduino UNO, while maintaining lower power consumption than the ESP32. It is capable of supporting touchscreen displays and is well-suited for handling sensor inputs and label printing.

However, it does not have built-in Wi-Fi or Bluetooth, meaning a separate wireless module would be required for connectivity. This approach can offer better power efficiency, making it a strong candidate for our project if wireless communication is needed without significantly increasing energy consumption.

3.2.6 Raspberry Pi Pico

The Raspberry Pi Pico is a high-performance microcontroller that offers good processing power for handling various tasks, making it an excellent choice for projects that require efficient computation. It has easy compatibility with touchscreen displays, allowing for intuitive user interfaces, which could be ideal for a digital scale with touchscreen input. The RP2040 chip at its core provides dual-core processing, enabling it to manage multiple tasks effectively, including handling sensor data and driving a display.

However, the Raspberry Pi Pico does not come with built-in Wi-Fi or Bluetooth, which limits its ability to connect wirelessly out of the box. If wireless communication is needed, external modules would need to be integrated, such as the ESP8266 or ESP32 for Wi-Fi, or a Bluetooth module. This adds complexity to the design but can be managed effectively, especially when power efficiency is not the primary concern. Despite this limitation, the Raspberry Pi Pico remains a strong choice for projects that need processing power and display integration while allowing for flexibility with external wireless modules.

Table 2: Microcontroller Comparison

Microcontroller	MSP430(TI)	Arduino UNO R3	RTK7RLG220C00000BJ (Renesas RL78/G22)	ESP32	STM32F411	Raspberry Pi Pico
Architecture	16-bit	8-bit	16-bit	32-bit	32-bit	32-bit
Clock Speed	Up to 16 MHz	16 MHz	32 MHz	Up to 240 MHz	100 MHz	133 MHz
RAM	2-8 KB	2 KB	4-8 KB	520 KB	128 KB	264 KB
Flash Memory	16-128 KB	32 KB	64 KB	4 MB	512 KB	2 MB
Connectivity	None	None	None	Wi-Fi, Bluetooth	None	None
I/O Interfaces	UART, SPI, I2C, ADC	UART, SPI, I2C, ADC	UART, SPI, I2C, ADC	UART, SPI, I2C, ADC, DAC, PWM	UART, SPI, I2C, ADC, DAC, PWM	UART, SPI, I2C, ADC, PWM

Power Consumption	Ultra-low power	Moderate	Ultra-low	Higher power consumption	Low Power	Low Power
Operating Voltage	1.8-3.6V	5V	1.6-5.5V	2.2-3.6V	1.7-3.6V	1.8-3.3V
Temperature Range	-40-85°C	0-70°C	-40-85°C	-40-125°C	-40-105°C	-20-85°C

3.2.7 Microcontroller Part Selection

The microcontroller selected for our project is the ESP32, specifically the ESP32-WROOM-32UE, chosen for its robust features and versatility. One of the primary reasons for choosing the ESP32 is its built-in wireless and Bluetooth connectivity, which is crucial for our project. This capability will allow the milk data logged on the device to be wirelessly transferred to the app via Bluetooth. Additionally, the ESP32 supports a wide range of communication interfaces, including SPI and I2C, which is beneficial given the variety of hardware components we plan to integrate into the system. The flexibility to use multiple communication protocols ensures that we can easily interface with different components as needed. Finally, the ESP32 operates at a voltage level compatible with our chosen power supply, making it a perfect fit for the overall design.

3.3 Scale Architecture

3.3.1 Strain Gauge Load Cell Scales

This type of scale uses strain gauge load cells, which are sensors that measure strain in response to the applied weight. When weight is applied, the metal structure deforms, causing a change in the electrical resistance of the strain gauge. This change in resistance is converted into an electrical signal, which is then processed by an Analog-to-Digital Converter and displayed as weight. Strain gauge load cell scales are commonly used in retail settings, warehouses, factories, and for personal weighing. Some advantages of this scale include its wide availability, reliable accuracy, and ability to handle a broad range of weights. However, some drawbacks include its sensitivity to temperature fluctuations and the potential for damage to the load cells if mishandled.

3.3.2 Electromagnetic Force Restoration Scales

Electromagnetic force restoration scales use a magnet and coil system to generate a force that counteracts the weight of an object. The current required to balance this force is measured and converted into the corresponding weight. These scales are commonly used in pharmaceutical and scientific laboratories, medical research, and jewelry weighing due to their high precision. The primary advantage of this type of scale is its extremely high accuracy, even down to micrograms. Additionally, the lack of mechanical wear contributes to its reliability. However, these scales are very expensive and are sensitive to vibrations and environmental factors, making them

unsuitable for portable use. They also have a limited weight capacity compared to other scales, though this is less of a concern for our design, as we are not dealing with heavy objects.

3.3.3 Capacitive Digital Scales

Capacitive digital scales measure changes in capacitance when weight is applied to a conductive plate. The force alters the dielectric properties, causing a shift in capacitance, which is then converted into an electrical signal representing the weight. Capacitive scales are commonly used for kitchen applications, postal services, and other compact weighing needs. They require fewer components, making them ideal for portable designs, and they consume minimal power. However, they are generally less accurate than other types of scales, and their reliability can be affected by temperature fluctuations.

3.3.4 Hydraulic or Pneumatic Scales

Hydraulic scales determine weight by measuring changes in fluid or air pressure. When weight is applied, the pressure increases, and this change is converted into a weight measurement. These scales are commonly used in construction, harsh environments, and for weighing extremely large objects. Their main advantage is their resistance to shock and overload, making them highly durable. Additionally, the absence of electrical components reduces safety risks in hazardous conditions. However, hydraulic scales are not suitable for portable designs, which is a key requirement for our project.

3.3.5 Piezoelectric Digital Scales

Piezoelectric digital scales utilize piezoelectric crystals, which generate an electrical charge when subjected to mechanical stress. The weight is determined by analyzing this charge output, as it is proportional to the applied force. These scales are commonly used in nanotechnology, aerospace, and medical applications, where extreme precision and stability are required. While their accuracy is a major advantage, they are impractical for our project due to their high cost and the constraints of our budget.

3.3.6 Smart and IoT Connected Scales

Smart scales use communication technologies like Bluetooth or Wi-Fi to send weight data to apps or connected software. They can integrate with Point of Sale (POS) systems to perform tasks such as syncing inventory, tracking health data, and logging information. Key advantages of smart scales include remote monitoring, data tracking, and integration with smart devices and automation, all of which enhance efficiency. However, they are typically more expensive than traditional digital scales and require ongoing software support, which can be a disadvantage. Additionally, the need for internet connectivity can interfere with portability, making them less suitable for mobile use.

3.3.7 Scale Part Selection

The scale we have decided to build is the Strain Gauge Load Cell. One of the reasons we chose this scale architecture is because of its compatibility with the ESP32 microcontroller we will be using. Another reason is due to its low voltage which allows it to be easily integrated with battery powered systems. Lastly, the portability of this scale compared to the others was a huge deciding factor since it is a major requirement of our project.

3.4 Weight Sensors

Due to the specific design requirements of our project, we will likely be utilizing a load cell as our primary weight sensor. Load cells offer high accuracy, reliability, and ease of integration with microcontrollers, making them well-suited for our application. Rather than relying on mechanical or non-electrical weight measurement methods, we will incorporate the load cell directly into our PCB design. This approach will allow for a more compact, efficient, and streamlined system, ensuring precise weight measurements while maintaining the overall portability of our device. By implementing the load cell on a PCB, we can also optimize signal integrity, reduce noise, and improve overall system stability, leading to more consistent and accurate readings.

3.4.1 Load Cells

Load cells are essential components in weight measurement systems, converting force into an electrical signal that can be processed and displayed. They come in various types, each designed for specific applications based on factors such as weight capacity, accuracy, and structural integrity.

3.4.1.1 *Single-Point Load Cells:*

Single-point load cells are among the most commonly used in small-scale weighing applications due to their lightweight design and high accuracy. One of their most significant advantages is their ability to measure weight accurately even when the load is not perfectly centered, making them ideal for compact digital scales. These load cells are widely used in kitchen scales, postal scales, and laboratory balances.

A well-known model in this category is the SparkFun SEN-13329, which has a maximum capacity of 5 kg. Made of aluminum, this load cell has an output sensitivity of approximately 0.9 to 1.1 mV/V and requires an excitation voltage of 5V. Its lightweight nature, affordability, and ease of integration with a load cell amplifier make it a popular choice for precision weighing applications. However, its relatively low weight capacity limits its use to light objects. Another widely used single-point load cell is the CZL635, which has a capacity of 10 kg. Like the SparkFun model, it is also made of aluminum but offers a slightly broader output sensitivity range of 0.85 to 1.15 mV/V, with an excitation voltage between 5V and 10V. This model is preferred for laboratory balances and lightweight industrial applications due to its enhanced durability, though it still shares the limitation of being restricted to low-weight measurements.

3.4.1.2 *Bar-Type Load Cells*

For applications requiring higher weight capacities, bar-type load cells are often the preferred choice. These load cells are commonly used in medium-scale weighing systems such as luggage scales, bench-top industrial scales, and larger digital weighing platforms. Compared to single-point load cells, bar-type load cells are structurally more robust but require an evenly distributed load to maintain measurement accuracy.

A widely used model in this category is the HX711-Compatible 50 kg load cell, which is constructed from aluminum alloy and offers an output sensitivity between 0.85 and 1.15 mV/V. It operates with an excitation voltage of 5V to 10V and integrates seamlessly with the HX711 analog-to-digital converter (ADC) module, making it an affordable and accessible choice for

digital platform scales. However, its primary drawback is its limited ability to handle uneven weight distribution. Another high-quality bar-type load cell is the Bosche H30A, which provides a maximum capacity of 60 kg and is constructed from stainless steel, giving it superior durability. This model features an output sensitivity of 2.0 mV/V and an excitation voltage range of 5V to 12V. While its stainless-steel build ensures longevity and resistance to environmental factors, it is more expensive than aluminum alternatives. The Vishay Tedeo-Huntleigh 1042 is another widely recognized bar-type load cell that is available in maximum capacities of 50 kg, 100 kg, and 200 kg. Made from stainless steel, it delivers an output sensitivity of 2.0 mV/V and operates within an excitation voltage range of 5V to 12V. Due to its high accuracy, it is commonly used in larger digital weighing systems and medical scales. However, precise mounting is crucial for optimal performance, as improper installation can impact measurement accuracy.

3.4.1.3 S-Type Load Cells

S-type load cells offer even higher weight capacities while maintaining precision, making them ideal for force measurement and industrial weighing applications. These load cells are distinguished by their "S" shape, which allows them to measure both tension and compression forces with high accuracy.

One notable model is the Vishay Tedeo-Huntleigh 614, available in maximum capacities of 100 kg, 200 kg, and 500 kg. Constructed from nickel-plated alloy steel, it has an output sensitivity of 3.0 mV/V and an excitation voltage range of 5V to 15V. Due to its durability and precision, it is widely used in force measurement systems and industrial applications where highly accurate load measurements are required. However, it is less commonly used for personal weighing devices due to its high weight capacity and structural complexity. Another reliable model in this category is the Zemic H3-C3-100kg-3B, which has a maximum capacity of 100 kg and is made of stainless steel. Operating with an excitation voltage of 5V to 15V, this load cell is commonly used for suspended weighing applications and dynamic force measurement. It features excellent overload protection, making it suitable for environments where unexpected force fluctuations may occur. However, its enhanced durability and advanced features make it more expensive than standard bar-type or single-point load cells.

3.4.1.4 Compression Load Cells

For large-scale industrial and heavy-duty weighing applications, compression load cells are the preferred choice. These load cells are specifically designed to measure compressive forces and are used in applications such as truck scales, weighbridges, and bulk material measurement systems.

The Vishay Celtron CLB is a well-regarded compression load cell with maximum capacities ranging from 500 kg to 10,000 kg. Made of stainless steel, it offers an output sensitivity of 2.0 mV/V and operates within an excitation voltage range of 5V to 20V. This model is widely used in high-capacity industrial weighing systems where precise and reliable weight measurement is critical. While it provides exceptional durability and accuracy for large-scale applications, it is not suitable for portable digital scales due to its high weight capacity and structural bulk.

Each type of load cell serves a specific purpose, with single-point and bar-type load cells being the most appropriate for compact and portable digital scale designs. The choice of load cell

ultimately depends on the weight capacity requirements, environmental conditions, and level of accuracy needed for a given application.

Table 3: Load Cell Comparison

Load Cell Type	Model	Max Capacity	Material	Output Sensitivity	Excitation Voltage	Key Features
Single-Point	SparkFun SEN-13329	5 kg	Aluminum	0.9 - 1.1 mV/V	5V	Lightweight, high accuracy, ideal for compact digital scales
Single-Point	CZL635	10 kg	Aluminum	0.85 - 1.15 mV/V	5V - 10V	Slightly higher capacity, used in laboratory and small industrial scales
Bar-Type	HX711 – Compatible 50kg Load Cell	50 kg	Aluminum Alloy	0.85 - 1.15 mV/V	5V - 10V	Affordable, pairs well with HX711 ADC, best for platform scales
Bar-Type	Bosche H30A	60 kg	Stainless Steel	2.0 mV/V	5V - 12V	Superior durability, used in bench-top industrial scales
Bar-Type	Vishay Teda – Huntleigh 1042	50 kg, 100 kg, 200 kg	Stainless Steel	2.0 mV/V	5V - 12V	High accuracy, commonly used in medical and industrial scales
S-Type	Vishay Teadea – Huntleigh 614	100 kg, 200 kg, 500 kg	Nickel-Plated Alloy Steel	3.0 mV/V	5V - 15V	Measures both tension and compression, used in force measurement systems
S-Type	Zemic H3-C3-100kg-3B	100 kg	Stainless Steel	3.0 mV/V	5V - 15V	High overload protection, ideal for suspended weighing applications

Compression	Vishay Celtron CLB	500 kg – 10,000 kg	Stainless Steel	2.0 mV/V	5V - 20V	Heavy-duty industrial applications, ideal for truck scales and bulk material measurement
-------------	--------------------------	--------------------------	--------------------	----------	----------	--

3.4.1.5 Load Cell Part Selection

For our load cell selection, we have chosen the SparkFun SEN-13329 due to its compact, lightweight design and high accuracy, which is essential for precise breastmilk labeling. Its 5 kg maximum capacity makes it well-suited for small-scale applications like our digital scale. Additionally, its 5V excitation voltage is easily supported by our system, simplifying integration. The SEN-13329 is also affordable and readily available, making it an ideal choice for both prototyping and final implementation.

3.4.2 Load Cell Amplifier

In our design, we will be using load cell amplifiers to properly translate the small and often noisy signals from our load cells into a form that is readable by our microcontroller. These amplifiers are essential in providing signal conditioning, filtering, and amplification to ensure accurate and stable weight measurements. By carefully selecting the right amplifier, we can optimize the performance of our weighing system and minimize errors due to noise, temperature changes, and electrical interference.

3.4.2.1 HX711 Load Cell Amplifier

One of the most used load cell amplifiers is the HX711. It is a low-cost, 24-bit analog-to-digital converter (ADC) specifically designed for load cell applications. With built-in gain settings and a simple two-wire communication interface, the HX711 makes integration with microcontrollers straightforward. This amplifier is widely used in digital scales and small weighing systems due to its ease of use and affordability.

One well-known module featuring the HX711 is the SparkFun Load Cell Amplifier – HX711, which offers selectable gain settings of 32x, 64x, and 128x. It operates on both 5V and 3.3V power supplies, making it highly compatible with different microcontrollers, including the ESP32.

Another widely available module based on this amplifier is the DFRobot Gravity HX711, which features onboard screw terminals for easy wiring and connection to load cells. Additionally, there are numerous generic HX711 modules available from various manufacturers. These lower-cost alternatives can be useful for prototyping, but their quality and performance can vary, requiring additional calibration and testing.

3.4.2.2 ADS1232 Load Cell Amplifier

For applications requiring higher precision and better noise filtering, the ADS1232 is an excellent alternative. Manufactured by Texas Instruments, this 24-bit high-resolution ADC offers superior noise reduction and signal stability compared to the HX711. It includes onboard low-

noise filtering and is designed to provide more accurate weight readings in environments where electrical interference could be an issue.

A closely related model, the ADS1234, shares the same core architecture but includes four differential input channels. This allows for multiple load cells to be connected simultaneously, making it ideal for applications where weight distribution needs to be measured at multiple points.

3.4.2.3 NAU7802 Load Cell Amplifier

Another high-precision load cell amplifier that stands out for its ease of integration is the NAU7802. Unlike the ADS1232, which uses an SPI communication protocol, the NAU7802 utilizes an I²C interface, simplifying communication with microcontrollers and allowing multiple sensors to be connected on the same data bus.

One of the most widely used modules featuring this amplifier is the SparkFun Qwiic NAU7802, which is designed for easy integration into embedded systems. The I²C communication capability makes this amplifier particularly suitable for applications where multiple weight sensors or other I²C peripherals, such as a touchscreen or additional sensor modules, need to operate in parallel.

3.4.2.4 Analog Load Cell Amplifier

For more custom weight measurement systems that require greater flexibility, analog load cell amplifiers can be used instead of dedicated digital ADC modules. These amplifiers take the small voltage signals generated by a load cell and amplify them before sending them to an external ADC for digitization.

One example of this type of amplifier is the Texas Instruments INA125, a high-precision instrumentation amplifier with a built-in voltage reference. Unlike dedicated ADC-based modules like the HX711 or NAU7802, the INA125 requires external circuitry for gain adjustment and signal conditioning. However, it offers greater customization and can be fine-tuned for applications requiring specific signal amplification characteristics.

Another similar option is the Analog Devices AD620, which also provides excellent gain control and stability for weight measurement applications but requires additional external components for proper operation.

The selection of the right load cell amplifier for our project depends on factors such as precision requirements, ease of integration, power consumption, and communication protocols. The HX711 is a reliable choice for low-cost, simple weighing systems, while the ADS1232 and NAU7802 offer superior accuracy for more demanding applications. Analog amplifiers such as the INA125 provide a more flexible but complex solution that can be tailored to specific measurement needs. By evaluating the trade-offs between these amplifiers, we can determine the best fit for our scale's performance and functionality.

Table 4: Load Cell Amplifier Comparison

Amplifier	Type	Resolution	Interface	Power Supply	Key Features	Common Modules
-----------	------	------------	-----------	--------------	--------------	----------------

HX711	Digital ADC (24-bit)	24-bit	Two-wire	5V / 3.3V	Low-cost, built-in gain settings (32x, 64x, 128x), easy integration	SparkFun HX711, DFRobot Gravity HX711, Generic HX711 modules
ADS1232	Digital ADC (24-bit)	24-bit	SPI	5V	High precision, low-noise filtering, better stability than HX711	Texas Instruments ADS1232
ADS1234	Digital ADC (24-bit)	24-bit	SPI	5V	Similar to ADS1232, but supports multiple load cells (4 channels)	Texas Instruments ADS1234
NAU7802	Digital ADC (24-bit)	24-bit	I ² C	3.3V / 5V	High-precision, easy multi-sensor integration, low power	SparkFun Qwiic NAU7802
INA125	Analog Instrumentation Amp	N/A	Analog Output	2.7V - 15V	Customizable gain settings, external ADC required, built-in voltage reference	Texas Instruments INA125
AD620	Analog Instrumentation Amp	N/A	Analog Output	$\pm 2.3V$ - $\pm 18V$	High gain control, external components needed for full operation	Analog Devices AD620

3.4.2.5 Load Cell Amplifier Part Selection

The Load Cell Amplifier we have chosen is the HX711. We have chosen this one due to its affordability, ease of use and built-in features. One of the main reasons we chose it is due to its simple two-wire interface. This will make it easier to integrate with the ESP32 and does not

require external ADC, which simplifies our design. It also has built in gain adjustability allowing it to be compatibility with different load cells and optimizes accuracy.

Module	Supply Voltage	Current Consumption	Interface	Gain Settings	Max Data Rate	Operating Temperature	Other Features
SparkFun SEN-13879	2.7V-5.5V	~1.5mA	Two-wire(clock data)	32x, 64x, 128x	80 SPS	-40°C to 85°C	High-quality PCB, Stable Performance
DFRobot Gravity HX711	2.7V-5.5V	~1.5mA	Two-wire(clock data)	32x, 64x, 128x	80 SPS	-40°C to 85°C	Plug & Play with Gravity ecosystem
Adafruit HX711 Module	2.7V-5.5V	~1.5mA	Two-wire(clock data)	32x, 64x, 128x	80 SPS	-40°C to 85°C	High PCB Quality, Breadboard-friendly
Generic HX711 Module	2.6V-5.5V	~1.5mA	Two-wire(clock data)	32x, 64x, 128x	80 SPS	-40°C to 85°C	Varies in Quality, Low-cost Alternative

The load cell amplifier we chose was the Spark-Fun SEN-13879. The main reason we chose this is because we chose the SparkFun SEN-13329 loadcell which is directly compatible with this amplifier because they come from the same company. Other reasons we chose this amplifier include reliability, wide power supply range and low power consumption. This amplifier is known for its high-quality PCB which ensures low noise and high stability. We were also able to find a lot of schematics and design examples for it. The amplifier is also compatible with our microcontroller (ESP32) and supports the 3.3V it utilizes. Lastly it helps maintain power efficiency which is important for our device that will be battery operated.

3.5 Printer

3.5.1 Zebra ZQ620 / ZQ520

The Zebra ZQ500 series Mobile Printers have a military-grade durability against various materials like concrete & sprayed liquids that could be useful for our product, as it will be thrown into on-the-go bags and must withstand potential bumps against other objects, as well as potential milk spills. The printer also comes with compatibility to an existing Zebra mobile app to configure different elements of the label printing, such as print quality and sizing. The ZQ500 series are all between 3-4 inches, making it compact enough to implement into our design. Despite these useful features, the Zebra ZQ620 sells for a relatively high price compared to our budget, with Google search results showing a range of >\$600.

3.5.2 Brother RJ-2035B

Brother advertises this RJ-2035B printer as being ultra-compact “yet extremely rugged with durable construction”. They claim an IP54 certification rating, protecting the printer from limited

dust & water splashes in any direction. Other specs they advertise are Bluetooth, NFC quick connection, 4 ips (inches per second) print speed, colored LED indicators, support for multiple operating systems, and a long-lasting battery life (although the exact lifetime is unspecified). The selling points are very similar to the Zebra option, but the advantage of the Brother printer is that it retails for \$380 on the official Brother website – a nearly 40% drop in price.

3.5.3 Bixolon SPP-R310

The Bixolon printer boasts wireless and wired connectivity options including Bluetooth® (MFi certified), Dual-band WLAN (5 GHz and 2.4 GHz), USB and Serial; IP54 water & dust proof; long-lasting battery life (24hr standby time); capability of printing both labels and receipts, with print speeds of 100mm/sec for receipts and 80mm/sec for labels; 1.8m drop protection; and finally a 385 g lightweight design that is easy to grip. For our product, the various connectivity options give us more flexibility in label control, and like the previous printers the durability ratings would help in creating a product that can hold up with potential accidents. The price range is also more desirable for us, selling online in the range of \$250-\$400.

3.5.4 MUNBYN ITPP941

The MUNYUN printer deviates the most from the other printer designs with support for various label sizes, a tradeoff for its larger & bulkier size. Since the printer is made more for office use, it does not aim for the same compact use as the other printers, which is its biggest downside for our project. There is also no Bluetooth connectivity, and functions solely through wired connections. However, the biggest advantage is its affordability with a cost that is less than \$200.

Table 5: Printer Comparison

Printer	Price	Print Speed	Power Consumption / Battery Life	Connectivity	Label Size Supported
Zebra ZQ620/ZQ520	>\$600	4.5(114.3mm) inches per sec	Battery powered	Bluetooth, USB	3-4 inches
Brother RJ-2035B	\$380	4(101.6 mm) inches per sec	Long lasting battery	Bluetooth, NFC, USB	2 inches
Bixolon SPP-R310	\$250-\$400	80 mm/sec for labels	24-hr standby battery life	Bluetooth,WLAN, USB	3 inches
MUNBYN ITPP941	<\$200	150 mm/sec	Wired power,no battery	USB	Supports various label sizes
Primuz Micro Thermal Printer	\$46.88	Up to 60 mm/sec	Wired power, no battery	TTL Serial, RS232	57x30 mm

3.5.5 Printer Part Selection

For our printer selection we went with the Primuz Micro Thermal printer. While doing research for the printer the main issue we were running into is finding a thermal printer that can be integrated into our system and supports label printing. We found a lot of printers that would integrate into our system, but they didn't support label printing. Then the options we found that

do support label printing have their own interface which means we would have to deconstruct the printer or use the interface they already have. With the Primuz Micro Thermal Printer we can easily integrate it into our system because it has a power supply wire, serial interface and supports the microcontroller we chose.

3.6 Display Screen

During our research, several options for display screens emerged as viable options. The most notable among them were the OLED, LCD, ePaper and Multi-Segment LED display screens. Each offers unique advantages in different areas such as power consumption, readability, and integration complexity.

3.6.1 OLED (Organic Light Emitting Diode)

The OLED display screen is a flat panel screen commonly used in TVs, smartphones, monitors and tablets. It works by passing an electric current through organic materials, which in turn emit light. By eliminating the need for a backlight, it allows the device to be more lightweight. The organic material used not only emits colors but also allows for better contrast ratios. Additionally, OLED displays are more energy-efficient and require less power consumption due to the absence of a backlight.

There are two types of OLED displays, PMOLED (Passive Matrix Organic Light Emitting Diode) and AMOLED (Active Matrix Light Emitting Diode). The PMOLED display uses an external semiconductor to control which pixels are lit. As a result, the size is limited to around 320 x 160 pixels. On the other hand, the AMOLED display can offer different levels of colors at each pixel rather than being restricted to only a few levels, as seen in PMOLED displays.

There are many advantages to using the OLED display. One key benefit is that, since it generates light directly, it requires less power consumption. This is especially important because we are trying to create a portable, battery-efficient device. Using components with lower power consumption will contribute to longer-lasting battery life. It has several other advantages, its flexibility, thinness, and lightweight design. This makes it particularly ideal for portable devices since it requires fewer layers without compromising performance.

Furthermore, the OLED display screen can support a wide range of temperatures, from -40°C to 80°C. This is important because, as a portable device, we don't know what climate the device will be used in. All components must withstand temperature fluctuations because they may be in different environments like a hot car or cold hospital freezer. The benefit of the display screen being able to handle this will prevent it from freezing or becoming unresponsive at low temperatures and screen degradation at high temperatures. One disadvantage that we must consider is that OLED screens are more expensive. Since we are working within a budget, it may not be cost-effective to buy these in bulk. Another disadvantage of using this type of display is the fact the organic material the lifespan is shorter than other display screens. A variety of factors can lead to the quickened degradation of the display screen including constant high temperatures.

3.6.2 LCD (Liquid Crystal Display)

An LCD is a display screen that uses liquid crystals to modulate light. The screen utilizes two glass plates and sandwiches the liquid crystals in between. The liquid crystals are arranged in

specific patterns that allow them to be manipulated when different electric currents are applied. This manipulation affects how the light passes through, which in turn allows for different images to appear.

There are three different types of LCD display screens: TN (twisted nematic), IPS (in-plane switching), and VA (vertical alignment). Twisted nematic is most commonly found one in LCDs. The liquid crystals twist at 90 degrees when electricity is applied. They are low cost and have high response times but have limited viewing angles and poor color accuracy compared to the other types. IPS works by having the liquid crystals rotate parallel to let light in when voltage is applied to enhance the image quality of the screens. Some advantages of using this type are that it is also cost effective, has a wide range of viewing angles and accurate color accuracy. A disadvantage is that it has a slightly slower response time in comparison to its counterparts, which is something we are focusing on diminishing. The last type is VA which has the liquid crystals align vertically with the glass allowing light to pass through when voltage is applied. Some of the pros include wider viewing angles and improved color/contrast compared to TN. The biggest disadvantage is that it has higher power consumption, something we definitely want to avoid and probably an option we would not choose.

The overall advantages of using an LCD display include low power consumption, thinness, and lightweight design. All these aspects are important since, as mentioned earlier, our device is portable and therefore, we want it to have a long-lasting battery life. Another benefit is that it has good color representation and accuracy on the screen itself. However, LCD displays also have drawbacks. Some disadvantages to using LCD displays are limited viewing angles, slower response time, the requirement of a backlight, and lower contrast than other displays. One of our main concerns with this display is its response time, as we want to minimize latency, which may not be as feasible with an LCD. Additionally, the need for a backlight means more power consumption and adds weight, which is not ideal for our design requirements.

3.6.3 Electronic Paper Display

An ePaper display is a type of screen created to replicate the look and feel of printed paper. This technology is not only energy efficient but also has minimal blue light emission. It functions by utilizing microcapsules filled with charged black and white particles. By applying an electric field, it moves the particles to form text and images. Unlike the LCD it does not require a backlight to do so.

There are three different kinds of ePaper displays: Monochrome, Color, and Flexible. As their names suggest, the first two types represent whether it is able to produce color or not. Monochrome produces black and white displays that need high contrast and low power usage, in devices such as kindles. While color ePaper displays are not as vivid in color as LCD or OLED displays, they are commonly used for digital advertising where power is more of a concern than color variety. The last type, Flexible ePaper, is built on bendable materials that allows it to be more adaptable for various applications such as wearable medical devices.

The biggest advantage of using this type of display would be the fact that it has extremely low power consumption. The only time it uses power is to update the display. Other notable advantages of using this display are that it does not have any glare from the sunlight, does not emit blue light, and has a wide range of viewing angles. The cons of the device heavily outweigh its benefits, due to the fact that its touchscreen capabilities are very limited. A major hardware

component in our project design is a touchscreen display that creates an interactive user interface on the device. With the ePaper display, it can support capacitive touch but has a very slow refresh rate making it unsuitable for interactive applications. Other disadvantages of this device include its limits in color range, fragility, and cost when compared to LCD displays.

3.6.4 Multi-Segment LED Displays

Multi-Segment LED Displays are seen very often in everyday life in digital clocks, scales, calculators and more. The display consists of light emitting diodes (LED) arranged in segments that light up in different combinations to create numbers, letters and symbols. Each segment has its own circuit that can be turned on and off using a driver IC or microcontroller.

There are four different types of Multi-segment LED displays: 7-Segment, 14-Segment, 16-Segment and Dot Matrix. The 7-Segment display consists of 7 individual segments arranged in the shape of an 8 with an optional decimal point. This type can display numbers 0-9 and some letters. Some benefits of this type of display are that it's easy to interface with a microcontroller, low cost, has low power consumption, and long lifespan. This type is limited in what can be displayed and has low resolution. The difference between the 14-Segment and the 16-Segment is the 14 can support both letters and numbers, but the 16 allows them to be more legible. As you go up in the number of segments it means higher power consumption as well as more complex circuitry. The Dot Matrix has a different setup where the LEDs are not put into segments but instead grids to be able to display any character, number or symbol. This type can support scrolling text, animations, and different font styles. It would be most suitable for the display we need since it is the most flexible with its functionality and can be expanded by linking multiple displays together. This type of display does require more complex control and consumes more power than the other three types.

The two biggest advantages of using the multi-segment display would be that it is extremely power efficient and has a long lifespan in comparison to the OLED and LCD. However, it is very limited in display capabilities, has no multi-touch support at all, and can't display interactive elements.

Table 6: Display Screen Comparisons

Display Type	Power Consumption	Viewing Angles	Refresh Rate	Color	Durability	Cost	Touchscreen Support
OLED	High	Excellent	Very Fast	Full Color	Moderate (Burn-in-risk)	\$50-\$200	Excellent
LCD	Medium	Good	Moderate	Full Color	High	\$30-\$100	Excellent
EPaper	Ultra Low	Excellent	Very Slow	Limited	Fragile	\$15-\$50	Limited
Multi-Segment LED	Very Low	Limited	Limited	Monochrome or some color	Very High	\$5-\$40	None

3.6.5 LCD Part Selection

After doing research on possible display screens we could use for our project, we decided on using an LCD touchscreen display. Out of the options we mentioned only LCD and OLED were able to support touchscreen capabilities, which is a huge factor in our project. The reason why we decided on the LCD specifically is that it is cheaper, does not have a high burn-in risk like the OLED, and we have prior experience using LCDs. For our device since the touchscreen is what will control the device inputs and outputs. We also wanted our screen to be about 4 to 5 inches since we will have a lot of options on the screen, but it needed to be small enough to fit within our expected dimensions. The options we have decided to proceed with are the ILI9488 LCD Display, DT050BTFT – 5" Color TFT LCD Display, and NHD-4.3-480272FT-CSXP-T (4.3" TFT LCD Display).

Table 7: LCD Display Screen Comparison

	ILI9488 TFT Display	DT050BTFT – 5" Color TFT LCD Display	NHD-4.3-480272FT-CSXP-T (4.3" TFT LCD Display)
Screen Size	3.5 inches	5.0 inches	5.0 inches
Resolution	320 x 480	800 x 480	480x272
Display Type	TFT LCD	IPS TFT LCD	IPS TFT LCD
Viewing Angle	Wide-angle	ALL(IPS)	ALL(IPS)
LCD Driver IC	ILI9488	ILI6137	ILI6137
Interface	SPI	SPI/RGB	SPI
Supply Voltage	3.3V - 5V	3.0-3.6V	3.0-3.6V
Operating Temperature	~ -20-60 °C	~20-70 °C	~20-70 °C
Touchscreen options	Resistive	Resistive or Capacitive	Resistive
Cost	\$16.99	\$55-\$92	\$60-\$75

We chose the ILI9488 TFT Display for its cost-effectiveness while still offering a comparable resolution and high visibility. Among the options considered, it is the most affordable, making it a practical choice without compromising essential features. Additionally, its 3.3V - 5V supply voltage aligns with the other models, eliminating the need for extra voltage regulators. The SPI interface ensures seamless integration with the microcontrollers we plan to use, further simplifying the design.

3.7 LEDs

In our project, we will implement visual feedback using LEDs to provide important status indications. LEDs will serve as a user-friendly way to communicate the device's state at a glance. Specifically, we will include an LED to indicate that the device is powered on and another to signal when the battery is low. These LEDs will enhance usability by giving clear, real-time feedback without requiring a display or serial output.

3.7.1 Standard Through-Hole LEDs

Standard through-hole LEDs are simple, low-cost, and easy to integrate into our circuit. They typically come in a 3mm or 5mm package and require a current-limiting resistor, typically between 220Ω and $1k\Omega$, to prevent damage from excessive current. The most common colors for status indication include red, which is commonly used for low battery or error signals, green, which is often used for power-on indicators, and blue, which is sometimes used for general status indication. Through-hole LEDs can be directly connected to a microcontroller GPIO pin, allowing us to turn them on or off programmatically based on battery status or power conditions. Their main advantages are their simplicity, durability, and ease of prototyping, making them ideal for initial breadboard testing and later integration into a PCB.

3.7.2 Surface-Mount Device LEDs

Surface-mount device LEDs are compact, lightweight, and well-suited for PCB integration. They come in different sizes, such as 0805 or 1206, making them ideal for small-scale or space-constrained designs. These LEDs can be placed directly on the PCB near the battery management circuit, reducing wiring complexity and improving reliability. However, SMD LEDs require precise soldering techniques, making them slightly more challenging to work with compared to through-hole LEDs. They also need careful PCB layout planning to ensure proper current-limiting resistor placement and correct routing to the microcontroller. Despite this, they are ideal for final production designs due to their compact nature and high efficiency.

3.7.3 Bi-color or Tri-color LEDs

Bi-color and tri-color LEDs are multi-functional LEDs that can display different colors based on the voltage applied. These LEDs typically have a common anode or common cathode configuration, meaning they share a common pin while allowing independent control of different color segments. A bi-color LED can display red to indicate low battery and green to indicate sufficient power. Similarly, a tri-color LED can offer additional status signals, such as red for low battery, green for normal operation, and blue for charging. These LEDs are efficient for space-saving designs because they replace multiple separate LEDs with a single component. However, they require careful microcontroller control, as the correct voltages must be applied to achieve the desired colors.

3.7.4 WS2812B

WS2812B LEDs are programmable, addressable RGB LEDs that allow for dynamic status indication. Unlike standard LEDs, which require a separate GPIO pin for each LED, WS2812B LEDs can be controlled using a single microcontroller pin via a serial data protocol. These LEDs are beneficial for applications where more advanced feedback is needed. Instead of a single-color low battery warning, a WS2812B LED could gradually change from green to yellow as the battery discharges, flash red when the battery is critically low, and pulse blue while charging before turning solid green when fully charged. While these LEDs provide highly customizable feedback, they are more complex to integrate and may be considered unnecessary for basic indicators. However, they are ideal if we want to provide a more visually appealing and user-friendly battery level display.

For our project, the final choice of LED will depend on factors such as simplicity, power efficiency, and ease of integration. If we want a straightforward and reliable approach, standard

through-hole or SMD LEDs will suffice. If we prefer a more compact and feature-rich solution, bi-color LEDs may be the best option. If we want to add dynamic visual feedback, WS2812B LEDs would be the most suitable.

Table 8: LEDs Comparison

LED Type	Size/Form Factor	Power Consumption	Control Method	Advantages	Disadvantages
Standard Through-Hole LEDs	3mm, 5mm (Through-Hole)	Low (Requires resistor)	Direct GPIO control	Simple, low-cost, easy to prototype	Requires external resistors, larger footprint
Surface-Mount Device (SMD) LEDs	0805, 1206 (SMD)	Low (Requires resistor)	Direct GPIO control	Compact, ideal for PCB integration, efficient	Requires precise soldering, harder to replace
Bi-color LEDs	Common Anode/Cathode	Low (Depends on active color)	Direct GPIO control	Displays two colors with one LED, space-saving	Requires multiple control signals, limited color options
Tri-color LEDs	Common Anode/Cathode	Low (Depends on active color)	Direct GPIO control	Displays three colors, flexible status indication	Requires more complex control logic
WS2812B (Addressable RGB LEDs)	Varies (SMD/Through-Hole)	Moderate to High	Serial data (1-wire)	Fully programmable, dynamic color transitions, requires only one microcontroller pin	More complex integration, higher power consumption

3.7.5 LED Part Selection

For our LED selection, we will use standard through-hole LEDs during the breadboard prototyping phase due to their ease of use and direct compatibility with microcontroller GPIO pins when paired with a current-limiting resistor. They allow for quick testing and modifications, making them ideal for initial development.

For our final PCB design, we will transition to surface-mount device (SMD) LEDs. These LEDs are compact, efficient, and space-saving, helping us meet our portability requirements. Their smaller size compared to through-hole LEDs allows for a more streamlined PCB layout. Additionally, SMD LEDs are more resistant to vibrations and movement, enhancing durability, which is crucial for a portable device. Another advantage is that they are compatible with automated assembly processes, ensuring efficient and reliable soldering for a professional-quality PCB.

3.8 Power Supply

After doing some calculations we estimated the amount of power our device will need is about 11.25 watts per hour/day based on a duration of 8hrs a day. All power supply information will be based on this estimated amount.

3.8.1 Primary (Non-Rechargeable) Batteries

Batteries are one of the most widely used power sources for portable devices, embedded systems and manufacturing applications. They can be split into two categories, primary and secondary. Primary batteries are ready to use and better for devices that need low power over long periods of time. They have a long shelf life, can be easily replaced and are less likely to overheat/explode. The common types of batteries are alkaline, lithium, zinc carbon and silver oxide. For this project we will not be considering lithium batteries since they are highly flammable.

3.8.2 Alkaline Batteries

Alkaline batteries are commonly found in households powering toys, remote controls and other devices. They are known as your AA, AAA and 9V batteries which are easily found in stores for pretty cheap. They have a typical voltage output of 1.5V per cell for all batteries except the 9V rectangular ones. They have a high energy density ranging from 100-150 Wh/kg and long shelf life but low lifespan.

3.8.3 Zinc-Carbon Batteries

Zinc carbon batteries, however, are less common but still used for low-power devices such as clocks and have a lower energy density. They have an output voltage of 1.5V per cell and an energy density of 35-55 Wh/kg. They are known as your AA, AAA, C and D batteries.

3.8.4 Silver Oxide Batteries

Lastly are silver oxide batteries which are small button cell batteries commonly found in watches and hearing aids. They output a voltage of about 1.55V per cell. They are known for their high energy-density ranging from 100-180 Wh/kg, stable voltage output and long shelf life. Some of the advantages of using primary batteries are, as mentioned before, that they have a long shelf life due to their slow discharge rate. They only use about 2-3% of their charge per year and can last up to 5 years or more. Another advantage is that they have a stable output meaning they are able to provide steady voltage throughout their lifespan. This also allows them to have longer runtimes since they have more energy stored per unit weight. Other reasons why we think these could work as our supply include the fact that they can operate in extreme temperatures, instantly ready to use and the low cost.

Although the advantages would be beneficial to our device, the drawbacks are just as significant too. Primary batteries are known for their longer lifespan but once they are depleted, they must be disposed of and replaced with new ones. This can be considered inconvenient in the sense that it is for mothers who are breastfeeding and may not have batteries at their disposal. This is also not economically friendly since it is contributing to electronic waste. Some of the metals the batteries are composed of can be toxic and harmful to dispose of. Another disadvantage is that even though they are cheaper in cost upfront, overtime is generally less cost-effective due to the fact that they must be replaced. Another major disadvantage is that they are not sustainable for

high drain devices such as power tools. This is important because we do not know how power intensive our device will be, and we want to make sure our power supply supports the device's needs.

3.8.5 Secondary (Rechargeable) Batteries

Secondary batteries are mainly used for high-drain devices since they can be recharged and reused. They use an electric current to reverse the chemical reaction inside the batteries. Commonly used secondary batteries include Lithium-Ion, Lead-acid and Nickel metal hydride.

3.8.6 Lithium-Ion Batteries

Lithium-ion (Li-ion) batteries are widely used in portable electronics, electric vehicles and other high-performance applications due to their high energy density (150-250 Wh/kg) and lightweight design. They typically output 3.6-3.7V per cell and have a relatively low self-discharge rate compared to other secondary batteries. One of the biggest advantages of this type of battery is their efficiency and long lifespan before noticeable degradation. They also have a high charge retention rate meaning they lose less energy when not in use and do not suffer from memory effect which can cause the battery to hold less charge. Although it does have some strong advantages, Li-ion batteries come with safety concerns. They are very sensitive to temperature and physical damage. This can lead to overheating, swelling or thermal runaway that may cause fires or explosions.

3.8.7 Lead-Acid Batteries

Lead-acid batteries are usually found in things like cars and backup power systems. Their output voltage is typically around 2V per cell and comes in 6V or 12V packs usually and have an energy density of 20-50 Wh/kg. They are inexpensive, reliable, and have a high-power output. They are also heavy and bulky, which is not what we want for our lightweight portable device.

3.8.8 Nickel-Metal Hydride Batteries

Nickel metal hydride batteries are much safer than lithium since they have no expulsion risk, but they have a higher discharge rate (~1% per day). They output about 1.2V per cell with an energy density of 60-120 Wh/kg. They are often found in hybrid cars and as rechargeable AA/AAA batteries.

One major advantage of using a secondary battery is the fact that it is rechargeable which reduces waste and long-term cost. They also hold more power than primary batteries allowing for longer usage between charge cycles. The batteries are safer than primary batteries as well due to the fact that they will not leak harmful chemicals once degraded.

The rechargeable factor in secondary batteries is useful but leads to their biggest disadvantage of a higher energy loss each time it is charging and even when not in use. The lifespan is also very dependent on how many charging cycles it can handle before it needs to be replaced. There is also a safety concern with using them due to some being highly flammable or having the capability to explode if mishandled.

3.8.9 Solar Power

Solar Power is the process of converting sunlight into electricity using photovoltaic cells and other solar energy technologies. It is a renewable, clean energy source that can be used for portable devices, homes, businesses, and even large-scale power grids. Solar power requires 3 components which include solar panels, charge controller and an inverter. There are three types of solar panels that can be used. Monocrystalline Solar Panels have the highest efficiency (~22%), long lifespan and work in conditions with dimmer light but are more expensive. Polycrystalline Solar Panels are more affordable with good efficiency (~16-18%) but require a larger size to produce the same power output as Monocrystalline. The last type is Thin-Film Solar Panels (~10-12%) which are lightweight and flexible, ideal for the pumped device and works well in high temperatures. It has a shorter lifespan and lower power efficiency than the other two types.

One major advantage to using solar panels is the fact that it is eco-friendly. There are no bounds to what it can power and requires minimal upkeep. However, it can be very costly just for the solar panel itself and is dependent on sunlight. If it is too cloudy or raining the device may not function making it less reliable for mothers who intend to use it.

3.8.10 Supercapacitors

Supercapacitors are energy storage devices that store electrical energy through electrostatic fields rather than chemical reactions. They can discharge and charge quickly despite their size, making them ideal for things like flashlights or power backups. The one major advantage is that unlike batteries their life cycle is extremely long, and it can last many cycles without significant degradation. It is also able to provide quick and powerful surges of energy for short periods and they have better thermal stability compared to batteries. This is important because its performance stability in various temperatures and increases the reliability of the device. Some disadvantages to using a supercapacitor would be the gradual voltage decrease because it can make it harder to regulate a constant voltage. Another disadvantage is that it may be larger in size to store larger amounts of energy as a battery. Their energy density is typically only 5-15 Wh/kg which is significantly lower than batteries. This is something for us to consider due to the fact that we are trying to make more of a compact portable device.

Table 9: Battery Comparison

Power supply	Energy Density (Wh/kg)	Lifespan	Charge Time	Cost Range (\$)	Size and Portability	Environmental impact	Best For	Drawbacks
Alkaline Batteries (primary)	100-150	5+ years (shelf life)	Non-rechargeable	\$0.50-\$5 per battery	Compact and lightweight	E-waste (single use)	Low-power devices	Short lifespan, costly long-term
Zinc Carbon batteries (primary)	35-55	3-5 years (shelf life)	Non-rechargeable	\$0.30-\$3 per battery	Compact and lightweight	E-waste (single use)	Very low-power devices	Low energy output

Silver Oxide batteries (primary)	100-180	3-10 years	Non-rechargeable	\$1-\$10 per battery	Very small and light weight	More sustainable than alkaline	Watches and small electronics	Expensive per unit
Lithium-Ion Batteries (secondary)	150-250	500-2000 cycles	1-3hrs	\$30-\$150 per battery	Compact, varies by form factor	Recyclable	Electric vehicles and portable electronics	High cost, may be limited by charge/discharge overtime
Nickle Metal Hydride batteries (secondary)	60-120	500-1000 cycles	1-4 hrs	\$5-\$20 per battery	Medium weight, AA/AA A size	Rechargeable but self-charges quickly	Rechargeable household appliances	Lower power capacity than Li-ion
Lead-Acid batteries (secondary)	30-50	300-500 cycles	5-10 hrs	\$20-\$200	Large and heavy	Toxic chemicals, recyclable	High power applications	Bulk, not portable
Solar Power	Varies	20+ years	Dependent on sunlight	\$50-\$500	Panels can be compact	Sustainable and renewable	Outdoor/portable use	Sunlight dependent and high upfront cost
Supercapacitors	5-15	1,000,00+ cycles	Seconds	\$10-\$200	Small to medium	Recyclable, long-life span	Quick energy burst	Lower energy storage than batteries

3.8.11 Battery Part Selection

For our device we decided to go with the lithium-ion battery due to it providing an optimal balance of power, size and weight. The battery is rechargeable as well, which is the most feasible option for our portable device. Due to us having multiple components that may require different amounts of power, we wanted a battery that could support high energy density and be flexible when it comes to different power needs.

Table 10: Lithium-Ion Battery Type Comparison

	18650 Li-ion Battery	14500 Li-ion Battery	16340 Li-ion Battery
Normal Capacity	2500 mAh	750 mAh	700 mAh
Nominal Voltage	3.7 V	3.7 V	3.7 V
Charge Voltage	4.2	4.2V	4.2V
Weight	~50 g	~20 g	~20 g
Operating Temperature Charge	0-45 °C	-20-45 °C	-0-45 °C

Operating Temperature Discharge	-10-60 °C	-20-60 °C	-10-60 °C
Cost	\$3-\$5 (cheaper in bulk)	\$5-\$16	\$4-\$10

We have decided to go with the 18650 Li-ion battery for our project. One reason is due to the higher capacity than the other two batteries listed which allows us more flexibility since we do not yet know how much power will be required for our device. It also allows for a more extended runtime as well as reducing the need for persistent charging. This type of battery is also very commonly used in small electronics like laptops and flashlights.

3.8.12 Part Selection Battery Charging IC/Module

Table 11: Battery Charging IC/ Module Comparison

	TP4056	BQ24296	TP5000	MCP73837
Best For	Single cell charging	Fast charging and USB-C PD	1S or 2S battery packs	Portable devices
Input Voltage	5V (Micro-USB or Type-C)	5V –17V(USB-C)	4V-9V	4.5V to 6V
Output Voltage	4.2 V	4.2 V	4.2 V (1S)/8.4 V(2S)	4.2V
Max Charge Current	1A	3A	1A	1A
Protection	Overcharge, Over-discharge and short circuit using DW01A and FS8205A MOSFETs	Thermal, Overvoltage and battery monitoring	Thermal and Overvoltage	Overvoltage, overcurrent, and thermal protection. Safety timer
Efficiency	~80-90%	~90-95%	~85-92%	~85-90%
Size	Compact	Small	Small	Compact
Cost	\$1-\$6	\$4-\$20	\$2-\$5	\$1.50-\$3.50

We have selected the MCP73837 as our battery charging module due to its versatility and suitability for our portable design. This module supports charging from both USB and an AC adapter, providing flexibility in power sources. Its compact form factor makes it ideal for our portable application, while its integrated protection features, including overvoltage, overcurrent, and thermal regulation, enhance reliability and safety. Additionally, the programmable charge current allows for customization based on battery capacity, and the built-in status indicators provide real-time charging feedback.

3.9 Voltage Regulators

In our project, we will be utilizing voltage regulators to ensure that each component receives the correct input voltage without overloading or underloading. Since our design includes multiple components with varying voltage requirements—such as the Arduino, touchscreen display, load cell amplifier, and label printer—proper voltage regulation is essential for stable and efficient operation.

Voltage regulators can be classified into two main types:

1. Linear regulators (LDOs) – Provide a stable voltage but are less efficient due to heat dissipation.
2. Switching regulators (buck/boost converters) – Efficiently step voltage up or down using high-frequency switching and external components like inductors and capacitors.

For our project, switching regulators are preferred due to their higher efficiency, making them ideal for battery-powered and portable applications. In this section we will discuss which regulators can be used in our project.

3.9.1 LM2576

The LM2576 is a step-down switching regulator that can take an input voltage of 7V-40V and output a fixed voltage of 3.3V, 5V, or 12V, or an adjustable voltage from 1.23V-37V. It has a maximum output current of 3A and an efficiency of around 77-80%. While it includes built-in thermal shutdown and short-circuit protection, it requires external inductors and capacitors for operation. The LM2576 is best suited for simple, low-current applications where efficiency is important but high power is not required. In our project, it could be used for stepping down a 12V source to 5V to power the Arduino or HX711 load cell amplifier, although its lower efficiency compared to newer regulators makes it less ideal for high-power components like the touchscreen or label printer.

3.9.2 LM2679

The LM2679 is a high-current step-down switching regulator that supports an input voltage range of 8V-40V and offers output voltage options of 3.3V, 5V, 12V, or an adjustable range of 1.2V-37V. It has a maximum output current of 5A and a higher efficiency of around 85%. Operating at 260 kHz, its increased switching frequency allows for smaller external components and improved performance. Given its ability to supply higher current output, this regulator is well-suited for powering the label printer, which will likely require a stable 5A output.

3.9.3 LM2596

The LM2596 is another step-down buck regulator similar to the LM2576 but commonly found in ready-to-use modules, making it popular in DIY projects. It has an input voltage range of 4.5V-40V, an output voltage range of 1.23V-37V (fixed at 3.3V, 5V, or 12V), and a maximum output current of 3A. With an efficiency of 80-85%, it is a practical choice for stepping down voltage for microcontrollers. In our project, it is an excellent option for supplying 3.3V to the ESP32 and touchscreen display, offering a compact and efficient solution for stable power delivery.

3.9.4 MP1584

The MP1584 is a compact, highly efficient step-down regulator with an adjustable output voltage range of 0.8V-20V and a maximum output current of 3A. It boasts 90% efficiency and operates at a higher frequency of 500 kHz, allowing for the use of smaller inductors. This makes it particularly useful for low-power circuits where space is limited. In our project, the MP1584 can be used to efficiently regulate power for the HX711 load cell amplifier or touchscreen display, ensuring stable operation while minimizing energy loss.

3.9.5 XL4015

For higher power requirements, the XL4015 is a step-down regulator with an adjustable output voltage range of 1.25V-36V and a maximum output current of 5A. It has an efficiency of up to 96%, making it one of the most efficient regulators for handling high-current loads. With its ability to drive power-intensive components, the XL4015 is well-suited for powering the touchscreen display, ensuring stable voltage delivery without excessive heat generation.

3.9.6 XL6009

For higher power requirements, the XL4015 is a step-down regulator with an adjustable output voltage range of 1.25V-36V and a maximum output current of 5A. It has an efficiency of up to 96%, making it one of the most efficient regulators for handling high-current loads. With its ability to drive power-intensive components, the XL4015 is well-suited for powering the touchscreen display, ensuring stable voltage delivery without excessive heat generation.

3.9.7 MT3608

The MT3608 is a step-up (boost) switching regulator that can take an input voltage as low as 2V and step it up to a maximum of 28V, making it useful when we need to power components that require a higher voltage than our power source provides. It has an efficiency of around 90% and a maximum output current of 2A, although at higher voltages, the current capacity decreases. Since our ESP32 operates at 3.3V, the MT3608 is not directly useful for powering it, but it can be beneficial in our project if we need to boost a lower voltage source (such as a Li-ion battery) to 5V or 12V for the touchscreen or label printer. However, for high-current loads like the label printer, the MT3608 may not be ideal due to its 2A limit, so we may need a more powerful boost converter like the XL6009 instead.

3.9.8 AMS1117

The AMS1117-3.3V is a linear voltage regulator that provides a fixed 3.3V output from an input voltage range of 4.5V to 15V. It is commonly used to supply power to microcontrollers like the ESP32, but it has a key limitation: being a linear regulator, it dissipates excess energy as heat, making it inefficient when stepping down from higher voltages like 12V. With a maximum output current of 1A, it can support basic ESP32 operations but may struggle with additional peripherals that require more power. In our project, if we are powering the ESP32 from 5V, the AMS1117 could provide a stable 3.3V supply, but using a switching regulator like the MP1584 would be a better choice due to its higher efficiency and lower heat generation.

3.9.9 LM7805

The LM7805 is another linear voltage regulator, providing a fixed 5V output from an input voltage of 7V to 25V. It has a maximum output current of 1.5A, making it useful for powering 5V components. However, like the AMS1117, it is a linear regulator and therefore suffers from low efficiency and heat dissipation, especially when stepping down from higher voltages like 12V or 24V. In our project, the LM7805 could be used if we need a simple 5V source for components like the touchscreen or a 5V sensor, but a switching regulator like the LM2596 or XL4015 would be a better alternative, providing higher efficiency and support for higher current loads.

Table 12: Voltage Regulator Comparison

Voltage Regulator	Type	Output Voltage Range	Max Output Current	Efficiency
LM2576	Buck	3.3V, 5V, 12V (Fixed) or 1.23V - 37V (Adjustable)	3A	~77%-80%
LM2679	Buck	3.3V, 5V, 12V (Fixed) or 1.2V - 37V (Adjustable)	5A	~85%
LM2596	Buck	3.3V, 5V, 12V (Fixed) or 1.23V - 37V (Adjustable)	3A	~80%-85%
MP1584	Buck	0.8V - 20V (Adjustable)	3A	~90%
XL4015	Buck	1.25V - 36V (Adjustable)	5A	~96%
XL6009	Boost	5V - 35V (Adjustable)	4A	~94%
MT3608	Boost	5V - 28V (Adjustable)	2A	~90%
AMS1117	Linear	3.3V (Fixed)	1A	~50%
LM7805	Linear	5V (Fixed)	1.5A	~50%

3.9.10 Voltage Regulator Selection

We have selected the LM2576 as our buck converter and the XL6009 as our boost converter based on their efficiency, reliability, and suitability for our project's power requirements. Our battery IC already has boost and buck functionality. However, if our systems draw more than 3A or require a higher voltage then these components will be necessary.

The LM2576 is a robust step-down regulator capable of providing a stable 5V output to critical components. With its wide input voltage range (7V-40V) and 3A current capacity, it ensures reliable power delivery to low-power components. Its built-in thermal shutdown and short-circuit protection further enhance system stability.

The XL6009 is a highly efficient and versatile boost converter, ideal for stepping up lower voltage sources to meet the demands of higher-voltage components. Since our label printer or touchscreen display may require more voltage than our power source provides, the XL6009 offers a reliable solution with high efficiency (up to 94%) and support for a broad output range (up to 40V).

3.10 Wireless Communication

3.10.1 Bluetooth

Bluetooth enables wireless communication between devices using a Bluetooth-enabled microcontroller or a dedicated Bluetooth module. Its low power consumption makes it ideal for our portable design. In our project, Bluetooth would allow the scale to connect to a mobile app for real-time data display. Additionally, it can sync weight readings from the scale to the app, where values could be stored and accessed later.

3.10.2 Wi-Fi

Wi-Fi enables wireless communication over local networks, allowing the scale to transmit weight data to the internet for remote access. Using a Wi-Fi module like the ESP32 or ESP8266, our scale could connect to a network and send weight measurements in real time. Since the ESP32 includes both sensor processing and Wi-Fi capabilities, it eliminates the need for an additional module. A major advantage of Wi-Fi is that it allows remote access from anywhere with an internet connection, making it a strong option for cloud-based data logging.

3.10.3 Zigbee

Zigbee is a low-power, low-data-rate communication protocol designed for short-range communication and mesh networking. It is commonly used in applications requiring energy efficiency and reliability. To integrate Zigbee, a Zigbee module can be paired with a microcontroller such as a Raspberry Pi, ESP32, or Arduino. By programming the microcontroller to communicate with other Zigbee devices, weight data can be transmitted to a Zigbee hub. This is particularly useful for applications where data needs to be shared over long distances within a network of connected devices.

3.10.4 Ultra-Wideband

Ultra-Wideband (UWB) is a high-bandwidth, low-power communication technology primarily used for precise location tracking over short to medium ranges. It provides centimeter-level accuracy, making it ideal for positioning systems. UWB can be integrated using specialized UWB modules programmed with a microcontroller. However, since our project does not require precise location tracking, UWB is unlikely to be useful for our scale.

3.10.5 Near Field Communication (NFC)

Near Field Communication (NFC) enables wireless communication between devices within just a few centimeters. To integrate NFC, an NFC module can be paired with a microcontroller, with software programmed to trigger actions when an NFC tag is scanned. The best use case for NFC in our project would be for secure, quick interactions, such as pairing the scale with a mobile app. However, due to its short range and limited data transfer capabilities, NFC is not suitable for continuous data transmission, making it unlikely to be a primary communication method in our design.

3.11 Microcontroller Interfaces

3.11.1 PWM

Pulse Width Modulation (PWM) is a technique used to control power delivery by varying the duty cycle of a digital signal. It is generally used for things like controlling brightness and motor speed. It can provide backlight support for our touchscreen as well as possible audio feedback and motor control for the printer.

Some advantages include energy-efficient power control, precise control over brightness and simple implementation. Disadvantages include electromagnetic interference from the constant switching of PWM signals and also power loss from the switching. Other disadvantages include Audible noise produced from the switching frequency possibly dropping to an audible range and incompatibility with certain analog systems such as audio amplifiers.

3.11.2 ADC/DAC

Microcontrollers primarily operate in the digital domain, but many sensors work with analog signals where ADC and DAC come into play. An ADC converts an analog signal into a digital value that the microcontroller can process. This would be used for sensor inputs such as the load cell which would have to convert the outputs received into digital signal. It would also apply for battery monitoring by measuring the battery voltage levels to determine the charge status. DAC does the complete opposite and converts digital signal into analog voltage. Some applications of this include audio feedback where DAC is used to generate sound or voltage control.

3.11.3 Serial Communication Interfaces

Serial communication is essential for microcontrollers to communicate with peripherals and external devices. The most common protocols are UART, SPI, and I2C.

3.11.4 UART (Universal Asynchronous Receiver Transmitter)

UART works by transmitting data one bit at a time in a start-stop manner. Some common applications of UART are for debugging, logging and communication with modules. UART is often used to send debug messages to a computer and another device. It is also used with many external devices such as Bluetooth modules, which is a huge aspect of our project.

Some advantages of using UART would for one be its simplicity. It only requires two wires, one for transmitting (TX) and one for receiving (RX). Another great thing about UART is that it does not need a clock signal and is already commonly used in many embedded systems.

Some disadvantages of using UART would be that it has a lower data transfer speed compared to SPI and I2C. This would be a minor consideration for us since one of the things we want to minimize in our project is latency and this would be a major contributor to that aspect of it. Another disadvantage is that each device needs a dedicated UART channel which can make the software composition of our project more complex.

3.11.5 SPI (Serial Peripheral Interface)

SPI is a high-speed, full-duplex communication protocol commonly used for sensors displays and memory chips. Some applications include a display interface which uses SPI for data

transfer and flash memory/SD cards which use it for reading and writing external storage. For SPI configuration it uses four main signals MOSI, MISO, SCLK and SS.

Some advantages of using SPI would be that it is very fast and supports multiple devices using CS (chip select) lines. The disadvantages are that it requires more wires in comparison to UART and I2C and doesn't have built-in error checking. This increases the likelihood of mistakes when configuring our devices and will cause troubleshooting time to be extensive.

3.11.6 I2C (Inter-Integrated Circuit)

I2C is a two-wire communication protocol commonly used to connect multiple low-speed devices to a microcontroller. Some applications of it include sensor communication commonly used by environmental and biometric sensors, as well as EEPROM storage to read or write to small memory chips and used by touchscreen controllers. For I2C configuration it uses two main lines: SDA for data transmission and SCL for the synchronization clock.

Some advantages of using I2C would be that it only requires two wires, devices have their own unique address allowing for easier communication and it has built-in error detection. The disadvantages are that it is slower than SPI and can be complex when it comes to setting up the addresses.

3.11.7 Two Wire Automotive Interface

Two-wire automotive refers to CAN (Controller Area Network) or LIN (Local Interconnect Network). CAN is a two-wire communication protocol used in vehicles and industrial systems. It allows multiple microcontrollers to communicate without needing a host computer. Common applications are vehicle communication commonly found in engine control units and airbag systems. It is also used for industrial automation in factories and medical devices.

Advantages include the fact that it supports multiple devices using a single bus, has robust error detection and correction. Some disadvantages are that it requires additional hardware for implementation such as a transceiver and is more complex than SPI or I2C.

LIN is a simpler, low-speed alternative for CAN. Common applications include automotive body control where it is used to control things like window motors and seat adjustments. Some advantages of using this type of interface would be that it has a lower overall cost than CAN and requires fewer components. However, it is slower and less robust than CAN.

Table 13: Microcontroller Interface Comparison

Interface	Speed	Wires Needed	Used For	Compatible Components
PWM	Variable	1 per channel	Backlight, motor control feedback audio	LCD backlight, servos, buzzers and DC motors
ADC/DAC	Variable	Analog input/output	Sensor reading, audio output	Load cells, temperature sensors, microphone speakers

UART	Up to 1 Mbps	2 (TX, RX)	Debugging and communication with modules	GPS modules, Bluetooth modules, RFID readers
SPI	Up to 50 Mbps	4 (MOSI, MISO, SCLK, SS)	Display communication and high-speed data transfer	TFT LCDs, SD cards, flash memory and high-speed ACDs
I2C	Up to 3.4 Mbps	2 (SDA, SCL)	Sensors, memory and low-power devices	OLED displays, EEPROMs, IMUs and environmental sensors
CAN	Up to 1 Mbps	2	Automotive and industrial automation	Motor controllers, vehicle ECUs, robotics systems
LIN	Up to 20 kbps	1	Automotive body control	Car window regulators, seat controls and lighting systems

3.12 Software Research

3.12.1 Mobile Operating Systems

Mobile operating systems are the backbone of mobile devices, managing hardware resources, applications and user interactions. As of today, there are over 6 billion smartphones users and the demand for them is constantly growing. Mobile OS platforms enable developers to create applications that cater to their user's needs. When developing a mobile application selecting the target OS is very critical since it determines the development tools, programming languages, and frameworks that will be used. The two most dominate operating systems today are Apple iOS and Google Android, which collectively make up about 99% of the smartphone OS worldwide. Other platforms such as KaiOS and HarmonyOS serve specific markets but remain secondary to the industry leaders.

3.12.2 Apple iOS

Since its debut in 2007, iOS has redefined mobile technology by offering a seamless and secure experience. Apple has continuously worked to enhance iOS so that it aligns with the evolving user expectations making it one of the most user-friendly operating systems available. iOS is known for its seamless interface that is designed to be accessible even to those with limited technical knowledge. It puts an emphasis on usability, ensuring most functions require minimal setup or customization.

Apple has always placed a strong emphasis on security and data privacy. They have updated iOS to protect user data through many applications including Face/Touch ID, app sandboxing, and

strict app store review process. With increasing security concerns in the digital world, it has led many smartphone users to favor Apple devices due to their privacy-first approach. With that being said, Apple has developed environments tailored for iOS due to their stricter security protocols.

- Swift: A modern, high performance programming language known for its safety and efficiency.
- Xcode: Apple's Integrated Development Environment (IDE) which provides tools for coding, debugging, and testing. Although Apple's strict guidelines provide both quality and security, it can also cause challenges for developers to meet the criteria.

3.12.3 Google Android

Android has been around since 2008 and has since become the world's most widely used mobile operating system. One of Android's most defining features is its open-source nature, allowing developers and manufacturers to modify the OS to suit their needs. This type of flexibility enables customization at both the software and hardware level, as well as diverse device options. Even though open-source software enhances flexibility, it also increases security risks. Unlike iOS which enforces a strict app review process, Android does not, leaving it more susceptible to malware and security breaches. Besides its openness, Android also caters to a wide range of price points, making it more accessible worldwide.

Just like Apple, Android provides development environments, offering tools and languages designed for scalability and efficiency.

- Java and Kotlin: Primary coding languages used for Android OS. Kotlin has gained more popularity due to its modern and improved safety features.
- Android Studio: An advanced IDE that simplifies app creation for Android iOS and optimization.

Due to global spread, Android provides developers with more opportunity when it comes to app creation for diverse audiences and device categories.

3.12.4 HarmonyOS

Harmony OS was developed by Huawei and introduced as an alternative to Android for Huawei devices. It is designed for seamless integration across multiple devices including smartphones, IoT gadgets, and smart TVs. Some key features of this OS are that it uses Microkernel-based architecture, is multi-device compatible, and provides support for Android applications. While HarmonyOS is still in early development stages compared to Android or iOS it is growing in China and other Asian Markets.

3.12.5 KaiOS

KaiOS is a lightweight mobile OS designed for feature phones. Feature phones are devices that provide essential smartphone capabilities while maintaining a physical keyboard and long battery life. Some key features of this OS include that it supports HTML5-based applications and provides smartphone functions like Google Maps. Although it is a lot less common in smartphones, KaiOS is ideal for some looking for budget-friendly devices in regions with limited internet access.

3.13 Programming Languages

3.13.1 C

C and C++ are fundamental low-level programming languages widely used in embedded systems due to their ability to directly manipulate memory and hardware registers. They follow a procedural programming structure, enabling precise control over system resources and allowing direct access to hardware components. These languages are highly compatible with platforms such as Arduino and ESP32, making them strong candidates for handling real-time sensor data, Bluetooth communication, and printer controls. C/C++ offer efficient execution with minimal abstraction from the hardware, allowing developers to manage memory manually, interact with registers, and integrate with device drivers.

This high level of control and efficiency makes C/C++ a reliable choice for low-power, performance-sensitive applications like our project. However, a notable drawback is the manual memory management and limited built-in support for object-oriented programming, which can increase development complexity and debugging time. While C/C++ require careful handling to avoid issues such as memory leaks or segmentation faults, they remain among the most suitable languages for embedded software development.

3.13.2 Python (MicroPython)

Python is a high-level programming language that supports multiple paradigms, including object-oriented, procedural, and functional programming. It uses dynamic typing, allowing for a beginner-friendly development environment where variable types do not need to be explicitly declared. This flexibility, combined with a vast collection of libraries and frameworks, makes Python highly efficient for a wide range of applications. Its line-by-line execution and availability of pre-built functions enhance code readability and simplify debugging.

MicroPython, a lightweight version of Python, allows developers to run Python code on microcontrollers while offering a faster development cycle compared to C/C++. This is particularly advantageous during the prototyping phase of embedded applications. However, MicroPython generally falls short in terms of performance and stability compared to traditional C/C++ firmware. This is a critical consideration for our project, since python seems best suited for rapid prototyping and this project requires efficient and reliable Bluetooth/Wi-Fi communication and precise control of the printer.

3.14 Mobile Application Programming Languages

3.14.1 Java/Kotlin

Java and Kotlin are widely used languages for Android development. Java is a high-level, object-oriented programming language that allows applications to run seamlessly across different operating systems via the Java Virtual Machine (JVM). It promotes modularity, reusability, and maintainability, making it a preferred choice for large-scale projects. Kotlin, a modern statically typed language, requires less code to achieve the same functionality as Java and is often favored for its improved developer productivity and cleaner syntax.

Both languages integrate smoothly with Android's native hardware features, including Bluetooth, local data storage, and system notifications, making them strong candidates for developing an app that communicates with embedded hardware. One major advantage is the ability to fully control performance and ensure compatibility with the microcontroller. However, a key limitation is platform dependency, applications built with Java or Kotlin are Android-only. This means a separate codebase would be required for iOS, leading to increased development time and ongoing maintenance. Despite this, they remain highly reliable choices for Android-focused mobile application development.

3.14.2 Swift

Swift is Apple's modern, general purpose programming language designed for native iOS development. It is recognized for its focus on safety, performance and ease of use. Swift is engineered to be both developer-friendly and highly efficient. It aims to produce software that runs lightning fast, while preventing common programming errors for an optimized performance across apple products. Its concise and readable syntax makes it accessible for beginners, while offering the flexibility to handle complex application development as well. Swift's many safety features include automatic memory management and protection against null pointer dereferencing, which reduces runtime errors. Additionally, it integrates seamlessly with Apple's development tools, making it easy to build responsive and polished interfaces. Swift supports features like Bluetooth, local notifications, and user interface framework to provide an interactive learning environment ideal for learning, experimenting with code and rapid prototyping.

However, similar to Java/Kotlin, Swift is platform-dependent, requiring a separate codebase for cross-platform compatibility. This increases both development effort and long-term maintenance. Nevertheless, Swift remains a future-focused and efficient choice for applications targeting the Apple ecosystem.

3.14.3 React Native (JavaScript/Typescript)

React Native is a cross-platform mobile development framework developed by Meta that enables apps to be built using JavaScript or TypeScript. It allows developers to maintain a single codebase for both Android and iOS, significantly reducing development time and complexity compared to building separate native apps.

The framework benefits from a large ecosystem, enabling rapid UI development and easy integration with third-party libraries. React Native also supports Bluetooth libraries, which could be used to build the inventory tracking interface and manage user input for this project. However, one key drawback is the performance trade-off due to its abstraction layer. Applications requiring constant Bluetooth communication or intensive data updates may experience reduced efficiency, as the bridge between JavaScript and native code can become a bottleneck. Additionally, debugging may be more challenging compared to fully native solutions. Despite these limitations, React Native remains a popular and effective solution for building cross-platform apps that target both Android and iOS.

3.14.4 Flutter (Dart)

Flutter is a cross-platform mobile development framework developed by Google that uses Dart programming language. Dart is an object-oriented, class-based language with a syntax style

similar to C. Flutter enables developers to build high performance renderings applications for both Android and iOS using a single codebase, significantly reducing development time. Flutter's rendering engine allows for near-native performance and offers features like hot reload, where developers can make rapid iterations and test the updated elements together, allowing for an accelerated development process. Additionally, Flutter supports Bluetooth communication, making it well suited for applications that need to interface with embedded devices. Although Flutter is a cross-platform framework, it delivers performance comparable to native applications. Dart's architecture promotes code scalability, reusability and maintainability. All these qualities make Flutter an optimal choice for a project requiring broader device compatibility without sacrificing performance or user experience.

Table 14: Programming Languages Comparison

Language / Framework	Platform Support	Programming Language	Key Features	Performance
Java/ Kotlin	Android only	Java or Kotlin	Direct access for Android native features, strong support for Bluetooth and hardware communication	High
Swift	iOS only	Swift	Optimized for Apple hardware, modern syntax with high performance and memory safety	High
React Native	Cross-Platform (Android and iOS)	JavaScript/ TypeScript	Large plugin ecosystem and hot reload for fast UI iteration on both platforms	Medium
Flutter (Dart)	Cross-Platform (Android and iOS)	Dart	Custom rendering engine for high performance UI on both platforms, near native speed	High

3.15 Code Repositories

3.15.1 GitHub

GitHub is a cloud-based platform that allows developers to create, manage, store, and share code. Built on Git, it provides powerful version control features that support collaboration and track changes throughout the development process. GitHub promotes team-based development by offering both public and private repositories with customizable access permissions. Developers can create branches to experiment or make updates without altering the original codebase, preventing accidental overwrites and enabling organized teamwork. These features make GitHub

a strong choice for managing both small projects and large-scale applications while reducing conflicts during development.

3.15.2 Bitbucket

Bitbucket is a Git-based platform designed for collaborative software development. It supports both public and private repositories and includes features such as pull requests, branch permissions, and code reviews to maintain a smooth workflow and high-quality code output. Bitbucket also integrates continuous integration and continuous deployment (CI/CD) pipelines, allowing for automated testing and improved delivery efficiency. Compared to GitHub, Bitbucket has a smaller open-source community, which can make third-party integration more challenging. However, it integrates seamlessly with Atlassian tools such as Jira, making it an excellent choice for teams using modern development and project management practices.

3.16 Databases

3.16.1 SQLite

SQLite is a lightweight, in-process database engine designed for embedded applications and mobile devices. It does not require a separate server process, making it highly portable and easy to integrate into software. SQLite uses minimal setup, reading and writing data directly to a disk file containing tables, indexes, triggers, and views. However, it is not ideal for high-concurrency applications, as it locks the database during write operations. Additionally, it lacks user-defined functions, limiting its relational capabilities. Despite these drawbacks, SQLite is a suitable choice for lightweight applications where simplicity and low resource consumption are key.

3.16.2 MySQL Database

MySQL is an open-source relational database that is commonly used due to its reliability and performance with large datasets. It supports high-concurrency environments and is able to handle large amounts of data by organizing it into one or more data tables. Developers are able to create, access, store, and modify data with ease making it a popular choice for data management. Nevertheless, with the advantage of scalability this causes more memory to be consumed in comparison to other database structures. Even with this disadvantage, MySQL remains a preferred choice for web development and cloud applications, because of its ease of use, and straightforwardness when it comes to managing and storing data.

3.16.3 Firebase Realtime Database

The Firebase Realtime Database is a cloud hosted NoSQL database, where data is stored in JSON format and synchronized in real-time to connected clients. It ensures data persistence by storing information locally and continues to update even when offline. This gives the user a responsive experience across multiple devices without the need to be connected online. The NoSQL database offers a flexible data structure making it well suited for applications that require real-time updates. However, the firebase has limited capabilities for larger-scale applications causing high data throughput and price. Despite the limitations, the firebase database could be used to sync measurements and timestamps in real-time, making it well suited for our project.

3.16.4 MongoDB

MongoDB is a NoSQL database that can store JSON formatted files making it an ideal database for applications that require dynamic schemas and scalability. It uses horizontal scaling allowing for distributed storage across multiple servers. MongoDB also ensures real-time content management for its clients that require indexing and handling of larger datasets. However, when handling larger datasets this can lead to higher memory consumption and inconsistencies. MongoDB lacks ACID (Atomicity, Consistency, Isolation, Durability) compliance which ultimately causes instability within the database. Although, these drawbacks are slightly outweighed by the ability to handle dynamic data changes and unstructured data for this project.

3.16.5 PostgreSQL

PostgreSQL is an open-source relational database management system (RDBMS) that supports both structured and semi-structured data. It is fully ACID compliant, offering high reliability, scalability, and extensibility, ideal for complex and data-critical applications. PostgreSQL supports advanced features such as custom data types, procedural languages, and sophisticated indexing strategies, giving developers flexibility in data modeling. The main drawback is its learning curve; optimal performance often requires a more experienced user. Nonetheless, PostgreSQL is a top-tier choice for enterprise-level applications, financial systems, and any use case where data integrity and consistency are paramount.

Table 15: Database Comparisons

Database	Type	Data Structure	RealTime Sync	Offline Access	Advantages	Disadvantages
MySQL Lite DB	Relational (Embedded)	Structured (tables)	No	Yes	Lightweight, fast local storage, low overhead, fully offline	No Realtime sync, not suitable for multi-user
MySQL DB	Relational (Server-based)	Structured (tables)	No	No	Powerful SQL querying, scales well for large datasets	No Realtime sync or offline support, custom backend server needed to support Flutter
Firestore Realtime DB	NoSQL (Cloud-hosted)	JSON	Yes	Yes	Real time updates, offline caching, syncs across all devices	Limited sorting ability for complex filtering

MongoDB	NoSQL	JSON/BSON	With backend	No	Supports unstructured, flexible data, scales well with large data sets	No Realtime sync, backend needed to support Flutter, not fully ACID compliant
PostgreSQL	Relational (Server-based)	Semi-structured	No	No	ACID compliant, supports complex queries, scales well with large data sets	No Realtime syncing without additional setup, requires backend server/ API to support Flutter

3.16.6 Database Selection

After researching multiple databases, we have selected the Firebase Realtime Database for our project. While several of the databases discussed above offer robust relational structures, our application requires a solution that implements real-time syncing data across devices. The Firebase Realtime Database provides seamless integration with Flutter, along with built-in offline support that automatically synchronizes data upon reconnection. Its ability to deliver live updates ensuring that caregivers and mothers receive timely information, making it the most suitable choice for our database needs.

3.17 3D Modeling Software

For the shell of our device, we are considering utilizing a 3D printer. It would be more cost efficient since we have access to them on campus and we can cater it to fit the exact dimensions needed for our device.

3.17.1 Tinkercad

Tinkercad is a free, web-based 3D modeling tool that has an easy-to-use interface. It is great for beginners, educational use and simple designs. Some key features of it include:

- Drag and drop modeling interface
- Large library of pre-made shapes
- Direct export to 3D printing formats (STL, OBJ)

Some advantages of using this software would be the fact that it is beginner friendly and easy to learn how to use. A major disadvantage of using this software would be that it has limited features for complex designs.

3.17.2 Fusion 360 (Autodesk)

Fusion 360 is a powerful, cloud-based 3D CAD software. It offers advanced design, engineering and simulation features. Key features include:

- Parametric modeling which allows for precise dimensions and assembly
- Cloud collaboration

- Computer-Aided Manufacturing for 3D-printing

Some advantages to using this software would be for one that we have prior experience using it in junior design. Another advantage is that it is free for students and has tutorials for how to do certain designs. Without any prior experience using this software it can become very complex and hard to learn how to do.

3.17.3 SolidWorks

SolidWorks is an industry-standard CAD tool used by engineers for creating complex mechanical parts, assemblies and simulations. It offers advanced features for product design and testing including the following:

- Parametric and direct modeling
- Simulation tools for stress testing and real-world performance
- Assembly design with part management

The advantages of using this software would be that it's good for creating precise, functional parts, has a large library of CAM tools and we have prior experience utilizing it.

3.17.4 FreeCad

FreeCAD is an open-source parametric 3D CAD modeler. It is primarily used in mechanical engineering and product design but is versatile when it comes to creating technical designs. Some key features of this software include:

- Parametric modeling
- Has Modularity with a variety of plugins available
- 3D printing integration with built-in export options

Some advantages of using this software would be that it is free, open-sourced and good for technical designs and projects. However, it can be less user friendly than other CAD software, has limited graphic output and animation tools.

3.17.5 OnShape

Onshape is a cloud-based CAD software that allows for real-time collaboration. It's great for team remote work and allows design files to be shared easily. Some features include:

- It is cloud based so no installation required
- Allows real-time collaboration for teams
- Has parametric and direct modeling

The advantages of using this software would be that it is free for non-commercial use and is great for team-collaboration, but it does have limited offline capabilities meaning we could lose our work.

Chapter 4 Standards and Design Constraints

4.1 Standards

4.1.1 Electrical and Power Standards

In this section, we aim to address the standards and design constraints that are critical for developing a successful and functional design. These factors include regulatory requirements,

material limitations, and technological constraints that will influence our design choices. Following these standards ensures reliability, safety, and efficiency when addressing potential challenges including manufacturability and user experience. All these issues have an impact on the shaping of our final design, which we must keep in mind throughout the development process.

4.1.1.1 PCB Standards

Our PCB layout must adhere to several critical requirements derived from multiple sources, with one key standard being IPC-2221. IPC-2221 is a widely recognized standard developed by the Institute of Printed Circuits (IPC), a global association that establishes guidelines for electronics manufacturing. This standard provides detailed instructions for layout, material selection, signal integrity, thermal management, and manufacturability.

Among the specific design rules outlined by IPC-2221 are those related to trace width, clearance, pad sizes, via dimensions, and layer stackup configuration. Trace width should be calculated based on the current requirements, using formulas specified by IPC-2221A or IPC-2222. For example, for a current of 2A and 1 oz copper, the trace width can be calculated to be approximately 0.336 mm. Clearance between traces depends on the voltage levels, with higher voltages requiring wider spacing to prevent electrical interference. For low-voltage signals, a minimum clearance of 0.2 mm is typically required, whereas for medium-voltage or high-voltage signals, the clearance should increase to 0.3 mm or 0.5 mm, respectively. For power and ground planes, IPC-2221 recommends using copper pours to ensure efficient power distribution and to minimize electromagnetic noise, which is particularly important for ensuring the accuracy of our load cell readings.

To ensure signal integrity, signal traces should avoid sharp bends, instead using rounded corners to reduce reflections. The length of traces should be kept as short as possible to minimize signal degradation and latency. Additionally, trace width should be matched to the impedance of the signal line (e.g., SPI or I2C) to maintain proper signal transmission. For example, SPI or I2C signal traces should be matched to a standard impedance of 50-70 ohms. Given the potential high current draw from components like the label printer, thermal management becomes essential. IPC-2221 recommends the use of thermal vias and heat sinks to dissipate heat generated by high-power components, such as the printer. Thermal vias should be placed in pads or beneath high-power components, with diameters of 0.5 mm or larger, to efficiently transfer heat to the PCB's back layer for better heat dissipation.

In addition to thermal and signal considerations, the standard specifies the importance of pad sizes and hole sizes to ensure that they conform to standardized dimensions for easier component placement by manufacturers. For example, for components with 0.8 mm lead diameters, the pad diameter should be approximately 1.2 mm, and for through-hole vias, the hole size should be at least 1.0 mm. It is also necessary to ensure that creepage and clearance distances are sufficient to prevent electrical shorts or arcing, particularly in high-voltage areas. For low-voltage circuits, the minimum creepage distance is 0.4 mm, while it should be increased to 1.0 mm or more for higher voltages.

By following these guidelines, our design will ensure signal trace widths are optimized for impedance matching with our HX711 load cell amplifier, and the use of power planes will reduce voltage drops, ensuring stable operation of the system. The inclusion of thermal vias and

copper pours will address heat dissipation, and proper pad and hole sizes will ensure manufacturability. Overall, IPC-2221 provides comprehensive guidelines that are crucial for achieving a reliable, manufacturable, and high-performance PCB layout for our project.

As part of our Senior Design course, we must adhere to specific engineering and design criteria throughout the development process. One of the primary requirements is that our final PCB must include at least one microcontroller unit (MCU), a dedicated power supply unit, and multiple peripherals integrated with the MCU. These peripherals must be carefully selected to align with our project's objectives while ensuring full compliance with course expectations.

Additionally, our final prototype must not contain any solder-less components, meaning breadboards and jumper wires used during early prototyping must be eliminated in favor of permanent, soldered connections.

Another key requirement is the replacement of development boards with a fully customized PCB. While we may begin prototyping with off-the-shelf modules (e.g., an ESP32 development board), these must be replaced with a custom-designed PCB featuring a well-structured layout. This process involves incorporating the essential circuitry directly onto our board, including voltage regulation, communication interfaces, and peripheral connections, ensuring a professional and manufacturable product.

Furthermore, we are required to integrate existing reference circuit designs into our project. This means leveraging proven industry-standard schematics for key components—such as power management, signal amplification, and communication protocols—and then modifying these designs to suit our specific application. This approach ensures that our final board design is both functional and optimized, reducing potential design flaws and improving overall system performance.

4.1.2 Battery and Power Standards

IEC 62133 is an internationally recognized safety standard issued by the International Electrotechnical Commission (IEC) for rechargeable batteries, specifically those using lithium-ion (Li-ion) and nickel-based chemistries. This standard ensures the safe use, design, and manufacturing of secondary (rechargeable) batteries, which are commonly found in portable electronic devices. For our project, which uses a lithium-ion battery, compliance with IEC 62133 is critical to ensuring that the battery is safe for use by mothers and caregivers. One key aspect of IEC 62133 is the overcharge test, which simulates a scenario where a battery is charged beyond its recommended voltage limit. For lithium-ion batteries, this means exceeding the standard 4.2V per cell. The test ensures that the battery does not overheat, leak, or explode under these conditions. Our device will need to include a Battery Management System (BMS) to prevent overcharging and to limit the voltage to 4.2V per cell, thus ensuring safe battery operation.

Another critical test described in IEC 62133 is the short-circuit test, which evaluates how a battery handles an accidental short circuit. The test ensures that the battery's internal protection mechanisms prevent catastrophic failures such as fires or explosions. To comply with this standard, our design must incorporate protection circuits that detect and mitigate short circuits, ensuring the battery remains safe under malfunctioning conditions. The standard also outlines the importance of battery durability through an impact and drop test, which simulates physical accidents. For our project, this is crucial as the portable scale may be handled frequently and subject to drops or impacts. The test ensures that the battery casing can withstand physical stress

without compromising the internal battery structure. To meet these requirements, we must use robust battery casings that can endure typical wear and tear.

Additionally, the standard specifies a thermal test to ensure that the battery operates safely at high temperatures. The battery must be able to handle up to 130°C without leakage or failure. Since our scale may involve high-power components such as the label printer, it's essential to ensure that the battery can handle the heat generated by these components. Incorporating thermal management solutions, such as heat sinks or thermal vias in the PCB layout, will be crucial in ensuring the lithium-ion battery remains within safe operating temperatures. The standard also stresses the importance of limiting charging current and voltage, recommending a 1C (1mA per hour) charging rate for lithium-ion batteries. This ensures that the battery charges at a safe rate that prevents excessive heat buildup during charging. Our charging circuitry will need to be designed to handle less than 1C charging rates, preventing overheating and ensuring the battery remains efficient over time.

Overall, adhering to IEC 62133 ensures that the lithium-ion battery in our portable breastmilk scale is safe to use. By following these rigorous testing protocols and design recommendations, we can prevent issues such as thermal runaway, short-circuiting, and physical damage to the battery, ensuring that the scale remains safe and reliable for the user. This includes integrating a Battery Management System (BMS), ensuring robust casing for impact resistance, and incorporating proper thermal management features. These measures are essential not only for user safety but also for regulatory compliance and product reliability.

4.1.3 LED Standards

As a part of our advanced goals, we will be implementing a visual indicator using LEDs. Due to this there are a variety of standards we need to conform to ensure the safety and efficiency of our device.

4.1.3.1 IEC 62031

This standard focuses on the safety requirements for LED modules used in general lightening applications. It provides detailed criteria to ensure that LEDs are safe to use in various environments and applications. The main aspects covered by this standard are the different types of standards such as electrical safety, electromagnetic compatibility, requirements for marketing, durability/lifespan and more.

As it relates to independent LED modules built into the hardware of a product, the standards lay out a commitment to transparency of manufacturing by requiring products to disclose the mark of origin (trademark, manufacturer's name or name of the responsible vendor/supplier) as well as the model number or type reference of the manufacturer. For us, this would mean listing the specific models of LEDs we use, which is listed here in Chapter 10 section 10.2. The component transparency extends to the range of component values; for example, the safe operating voltage of the LED needs to be stated to the user to ensure proper usage of the device. For our selected LEDs, the operating voltage is between 3.2-3.5V. By following these standards, we can ensure that the LEDs used in our project function properly and are not a safety hazard to other users.

4.1.4 Wireless Communication Standards

The device will use Bluetooth Low Energy (BLE) for connectivity to the app so it must comply with the IEEE 802.15.1 standard for Bluetooth and the Bluetooth SIG specifications for BLE.

Compliance with these standards will ensure secure and efficient data transfer as well as allow compatibility with multiple devices.

IEEE 802.15.1 are described under the Wireless Personal area network. Bluetooth SIG established the specifications for BLE (Bluetooth Low Energy) which is when IEEE added a standard for it. BLE must operate in the 2.4 GHz ISM band using Gaussian Frequency Shift Keying (GFSK) modulation which ensures minimal power consumption and reliable data transfer. BLE follows a layered protocol stack which defines how the data is transmitted and received. At the physical layer BLE supports multiple modes including 1Mbps, 2Mbps, and LE Coded PHY which allows for optimal data rates and a range of up to 100 m depending on the environment. The link layer manages device connections, security and packet transmission. The Generic Attribute Profile (GATT) and Attribute Protocol (ATT) define how structured data is exchanged between connected devices. Additional layers such as SMP and GAP provide encryption, authentication and device role management.

Our breastmilk scale functions as a peripheral that advertises its presence and waits for a central device (ie. smartphone) to establish a connection. This setup ensures energy-efficient communication since the scale remains in a low-power state when it is not actively transmitting data. We will also be implementing the previously mentioned BLE mode to reduce the power drain when the device is idle.

4.1.5 Breastmilk Storage Standards

There are several important guidelines for the safe storage and handling of breast milk, as recommended by the CDC. Breast milk should be stored in breastmilk-specific storage bags or clean, food-grade glass or plastic containers with tight-fitting lids. Each container must be clearly labeled with the date the milk was produced, and if the milk is intended for childcare or donation, the child's name should also be included. Proper storage location is essential. Milk should be kept at the back of the refrigerator or freezer instead of the door to avoid temperature changes. To reduce waste, it is best to freeze milk in small portions of two to four ounces. There should be at least one inch of space left at the top of each container to allow for expansion during freezing. When traveling, expressed milk can be stored safely for up to twenty-four hours in an insulated cooler bag with ice packs. Regarding storage time, freshly pumped milk can sit at room temperature for up to four hours, refrigerated milk is good for up to four days, and frozen milk is best used within six months but can be kept for up to twelve months. Thawed milk should be used within twenty-four hours and should never be refrozen. The oldest milk should always be used first following the first in, first out principle.

These guidelines have helped shape the design and features of our breastmilk management system. Our project combines a digital scale, touchscreen display, and label printer to help mothers store and track expressed milk more easily. By including these recommendations in our system's software, users will receive printed labels with the date, volume, and optional dietary restrictions. The device will also send reminders when milk is approaching expiration to help prevent waste and ensure safety. The mobile app will log all milk entries and guide the user to use the oldest milk first. By turning public health recommendations into helpful tools and features, our system supports mothers in managing breastmilk safely and effectively while reducing daily stress and decision making.

4.1.6 Security and Software Protocols

Our project includes a mobile application that will use Bluetooth Low Energy (BLE) to transmit data between the device and the app. Given the sensitivity of the data being exchanged, it is essential to implement robust security protocols to protect user information and ensure the integrity of our system.

4.1.6.1 ISO/IEC 27001 (Information Security Management)

ISO/IEC 27001 is an international standard for Information Security Management Systems (ISMS). It outlines the requirements for establishing, implementing, maintaining, and continually improving an organization's information security management system. For our project, adhering to ISO/IEC 27001 is crucial because the device and mobile application will be collecting and transmitting sensitive personal health data, such as milk storage records and other health-related information. By following this standard, we can ensure that the data is securely handled and protected from unauthorized access, which is paramount to safeguarding user privacy and ensuring compliance with relevant regulations like HIPAA (Health Insurance Portability and Accountability Act).

One of the key aspects of ISO/IEC 27001 is its focus on risk management, which involves identifying potential security threats and vulnerabilities and mitigating them through effective controls. Specifically, the standard advocates for end-to-end encryption to secure data at all stages of its transmission. For example, as our device will be using Bluetooth Low Energy to transmit data between the scale and the mobile application, the data must be encrypted to protect it from unauthorized interception. BLE itself supports pairing and bonding mechanisms, which can be used to establish secure communication between the device and the app, ensuring that only authorized devices can connect and exchange data.

Moreover, ISO/IEC 27001 specifies the implementation of audit trails and logging mechanisms, which allow the tracking of access to sensitive data and any changes made. This is critical in detecting potential security breaches or unauthorized activities. By incorporating this feature into our system, we can provide a transparent record of all interactions with the device and the data, which could be crucial for both security audits and regulatory compliance.

Furthermore, ISO/IEC 27001 mandates the regular review and improvement of security practices to adapt to emerging threats and vulnerabilities. As our device will continuously evolve, ensuring that security measures are regularly updated and tested will be essential to maintaining a high level of data protection.

By adhering to the guidelines set forth in ISO/IEC 27001, we can ensure that both the device and mobile application handle personal health data securely, protecting user privacy and ensuring that sensitive information is transmitted and stored in compliance with the highest standards of data security.

4.1.6.2 ISTQB Software Testing Standard

The International Software Testing Qualifications Board (ISTQB) is an internationally recognized organization that provides a structured approach to software testing, ensuring the reliability, performance, and quality of software systems. ISTQB's testing methodologies and frameworks are designed to cover all aspects of software development, from functional testing to

security and performance testing. For our project, which includes firmware development, touchscreen UI design, mobile application functionality, and wireless communication integration with the microcontroller, adhering to ISTQB guidelines will be crucial to guarantee that each component operates as expected and that potential issues are identified early in the development process.

One of the core principles of ISTQB is the V-Model of software testing, which emphasizes the importance of verifying and validating software at each stage of development. This model involves planning tests early in the software development lifecycle and performing them in parallel with the development process. For instance, when developing the firmware for our ESP32 microcontroller, we will apply unit testing to ensure individual functions and modules work as expected. This can include tests for sensor readings, data processing, and communication with peripherals such as the HX711 load cell amplifier.

In addition to unit testing, integration testing will be crucial for validating the interactions between the mobile app, firmware, touchscreen UI, and wireless communication components. Given that our device will transmit weight data via Bluetooth Low Energy, we must ensure that data transmission between the device and the mobile app is secure, reliable, and error-free. System testing will verify that the entire system works cohesively, from the touchscreen UI capturing user inputs to the printer receiving and printing the weight data accurately.

Regression testing is another important aspect of ISTQB's methodology that will apply throughout our project. As we integrate new features or fix bugs, regression testing ensures that existing functionality is not broken by these changes. This will be particularly important when adding features such as secure BLE communication or when enhancing the mobile app with additional functionalities for managing breast milk storage and health data.

Performance testing will also be necessary to ensure the device responds quickly and operates efficiently, particularly when handling real-time sensor data and managing print jobs. Given that our device involves data handling, authentication, and communication over BLE, it's crucial that the system can operate within the expected time frame under typical usage conditions. Load testing may also be required if our mobile app is expected to handle multiple users simultaneously or large amounts of data.

ISTQB also places a strong emphasis on security testing to ensure that sensitive data is protected throughout the software's lifecycle. In our case, security testing will involve verifying that personal health data, such as breast milk storage and usage records, are securely transmitted over Bluetooth Low Energy, properly encrypted, and stored with secure authentication mechanisms. By following ISTQB security testing guidelines, we will mitigate risks associated with unauthorized access or data breaches, which is particularly important given the sensitive nature of the information.

Overall, adopting ISTQB guidelines throughout the software testing process will ensure that the firmware, mobile app, touchscreen UI, and wireless communication components are thoroughly tested, free from critical bugs, and meet the performance and security standards required for a reliable and trustworthy device. By doing so, we will improve the quality of our product, identify issues early in development, and ensure the final system meets the needs of users, particularly in terms of usability, security, and reliability.

4.2 Constraints

4.2.1 Time Constraints

One of the most significant constraints for our project is time. We have a total of 16 weeks in Senior Design I and only 9 weeks in Senior Design II due to the condensed summer semester. This compressed schedule requires careful planning and efficient execution to ensure we meet all project requirements within the allotted time.

Our design process will commence in March 2025, and we must have a fully functional prototype completed by early August 2025. As such, we must approach the timeline with careful planning and disciplined execution. We must prioritize key tasks such as component selection, procurement, software development, hardware integration, and testing to ensure timely completion.

Each of these tasks are heavily interdependent. Delays in any phase could impact subsequent stages, making time management critical. We must allocate sufficient time for component selection, ordering and delivery, especially since hardware delays are common occurrences that may affect integration. This makes early sourcing of components a priority, especially since a common issue seen in senior design is the delay in PCB design. Especially in our case, our microcontrollers, load cells, and printers can vary due to supply chain issues or limited stock. We plan to identify and order long-lead time components as early as possible to avoid bottlenecks later in the development process.

In parallel, software development must progress steadily to allow time for debugging, testing, and refinement. Mobile app development, Bluetooth integration, and Real-time data synchronization cannot be completed in isolation. They heavily are dependent on the functional communication between hardware and software. Unexpected issues with Bluetooth/Wi-Fi communication, sensor calibration or printer reliability could arise, making these integration challenges become unpredictable and time consuming.

To mitigate these risks, we will establish clear milestones, track progress regularly, and address potential bottlenecks proactively. Regular meetings between the team and project sponsors will ensure accountability and provide opportunities to adjust accordingly if setbacks are encountered. Additionally, assigning periods of accommodation for hardware debugging and late-stage testing will ensure the ability to make changes instead of cutting corners to meet the end goal.

By adhering to a well-structured timeline and maintaining flexibility for unforeseen challenges, we aim to deliver a robust and fully operational prototype within the given timeframe. Our ability to manage time effectively will be essential to meet both project requirements and ensure our final project is reliable and usable.

4.2.2 Ethical and Social Constraints

Since our project is sponsored, the intellectual property (IP) rights for the final design and its associated software legally belong to our sponsors. As such, we are both ethically and legally obligated to ensure our design complies with relevant IP regulations and does not infringe on any existing patents. This requires comprehensive research into existing technologies and patent

filings to avoid potential legal conflicts. Given that scales with integrated label printing already exist, it is our responsibility to ensure that our final design introduces unique features or innovations that differentiate it. This may involve modifying functionalities, improving efficiency, or integrating novel elements that do not overlap with patented technologies.

In addition to IP concerns, we also carry ethical responsibilities related to user data privacy and security. Our project involves tracking breastmilk production, storage, and usage for individual mothers, which are all sensitive information that must be handled with care. When this data is shared with caregivers or integrated with milk banks, strong data management practices are essential. This includes secure storage, encryption, and user consent protocols to prevent unauthorized access. As the system evolves, we must continuously evaluate and implement best practices for ethical data handling. For instance, if integrated with hospital systems, the device must comply with HIPAA (Health Insurance Portability and Accountability Act) standards.

From a social perspective, our project is intended to support mothers, infants, and caregivers, which places a heightened responsibility on us to ensure the system is safe, reliable, and accessible. A simple and intuitive user interface is crucial to accommodate users who may be experiencing fatigue, stress, or limited technical experience. With potential commercialization in mind, designing for accessibility and ease of use is a priority to ensure equitable access for all users.

Finally, we have a broader ethical obligation to approach our design with empathy. Our sponsors, who bring both personal and professional insights as parents and engineers, play an active role in this process. Their involvement allows us to remain grounded in real-world needs and better understand the experiences of mothers who pump and store breastmilk.

4.2.3 Economic Constraints

Our sponsors have generously allocated a working budget of \$500 to support the development of our prototype. This amount is expected to cover most of our expenses, including essential components and materials. Fortunately, our design is relatively simple in terms of mechanical complexity, with only a few moving parts. However, a significant portion of our budget is dedicated to the thermal printer, which appears to be the most expensive component based on our research and cost estimates. The printer is critical to the core functionality of our system, as it is responsible for producing the labels that track breastmilk storage data.

Given the printer's importance to our system, we may need to make trade-offs between cost and performance—especially if more affordable printers come with technical limitations or challenges. Other necessary expenses, such as the microcontroller, load cell sensors, power supply, PCB printing, and prototyping materials, will further tighten the budget, leaving little room for error or reordering.

As much as we'd like to stay within our sponsors' allotted budget, we may need to contribute our own personal funds to bring this design to life. Unforeseen circumstances may require higher-specification components or replacements that the current budget does not account for. Our goal is to deliver a reliable and fully functional prototype that meets all our project requirements. To achieve this, we must make strategic financial decisions and manage component usage carefully, keeping in mind that some out-of-pocket investment may be necessary.

4.2.4 Manufacturability Constraints

For the hardware aspect of our project, all the components need to be housed in an enclosure that is compact, portable, and easy to use. Although no one in our group is very familiar with any 3D printing design programs, it may be our best option in terms of price and speed to create the enclosure, as opposed to laser-cutting or outsourcing the manufacturing process to a third-party. Additionally, our purchased printer will already have its own enclosure that will need to be considered in our design since we would have to be able to access the label paper in order to refill it. The printer's charging method would also need to be considered in our overall design's power supply & PCB design.

Something worth noting is that from a course perspective, our enclosure and design could be basic enough to the point where considerations such as the aforementioned printer paper tray and charging port could simply be accessed through holes or cutouts in the enclosure. However, this project was conceptualized by our sponsors who want a design that can be manufactured commercially in the future. With that in mind, the design we came up with for this project needs to reflect a higher standard beyond simple functionality. Whether or not our sponsors reuse our design in its entirety, our design needs to be sleek and marketable enough to hopefully alleviate or dispel the need to be redesigned in the future.

4.2.5 Realistic Design Constraints

Another important consideration for our project is the set of practical design constraints that affect functionality, performance, security, and scalability. One of the most significant technical challenges lies in hardware limitations that directly impact software development. For instance, microcontrollers typically offer limited processing power and memory, which restricts the complexity of embedded software. As a result, the software must be carefully optimized for efficient memory usage to prevent crashes and ensure reliable operation. Features such as Bluetooth communication, real-time data processing, and UI responsiveness must be balanced against these limitations to achieve the best possible performance. Additionally, many microcontrollers have minimal onboard storage, requiring data to be offloaded to an external database. This introduces further constraints related to memory management and the reliability of wireless communication. Altogether, these factors demand careful RAM optimization and robust data handling to maintain system stability.

Portability is another key constraint. To keep the system lightweight and battery-operated, the hardware must be energy efficient. However, features like continuous Bluetooth or Wi-Fi communication can quickly drain the battery. To mitigate this, the software must minimize transmission times and implement low-power protocols whenever possible. While these strategies may impose certain limitations on the user experience, they are essential for maximizing the system's operational longevity and achieving our portability goals.

Chapter 5 Comparison of ChatGPT with other Similar Platforms

The concept of artificial intelligence is a broad one that has many applications in the modern day. In 1950, Alan Turing proposed the idea that by the turn of the century, computer-generated communication would be indistinguishable from human communication, at least 70% of the time (Codecademy). In the last 70 years, the research into AI has been developed through various specific points of interests, whether it be medicine (MYCIN, 1972) or music (WABOT, 1980's);

but most importantly in 1965 with the chatbot ELIZA, one of the first AI programs to imitate human conversation. It is this branch of AI use that has led to the development of chatbots such as ChatGPT, Google Gemini, and Microsoft Copilot; all of which are largely used as search engines for various information, and which most directly relates to our usage of AI in modern education.

Research has always been a difficult process without any aid, seeing as most people have used some form of online search engine for every inquiry ranging from personal to professional. AI chatbots & search engines have recently seen a surge in usage, likely due to its ability to condense research from various sources and respond to inquiries with more insight than regular search engines, all while mimicking the conversational tone of other human beings. However, this useful and innovative tool is not without its drawbacks and repercussions, which we will be discussing in this section as well as its impact on education and this project, specifically.

5.1 Benefits

5.1.1 Specificity

While most search engines will suffice with finding generally available information or answering simple factual questions, many times during research a question will arise that cannot be answered as easily with a single prompt. AI platforms have the advantage here; whereas most search engines are simply compilations of available sources and articles that are then sorted through for relevance to your inquiry, AI can take several relevant sources and use them to develop a solution to a more complex question. In addition, AI can actively improve its response by asking more about the prompted inquiry, rectifying gaps in its own knowledge and understanding in an attempt to better suit the needs of the user.

5.1.2 Inspiration

In addition to the specificity, AI can be a useful tool in *guiding* research. Specifically with instances like Senior Design where the end goal is known but the work leading up to it is up for interpretation, AI and its expertise can offer a place to start. For those who may be lost on what exactly to ask/search, AI can accept a general set of instructions/directions as opposed to a specific question and suggest next steps to follow - something general search engines like Google cannot do. This assistance is true for any stage in the design process, in whatever step a person may be stuck on. Simply explaining the problem at hand to AI can lead to a breakthrough and facilitate progress throughout the entire project.

5.1.3 Multimedia

Generative AI is not only useful as a search engine. Many AI platforms including ChatGPT are capable of providing more than just text-based responses – they can also create a document for various applications. AI can generate schematics, mockups, diagrams, data tables, scripts, codes, and many more resources depending on the needs of the user.

5.2 Disadvantages

5.2.1 Environmental Concerns

Most AI platforms are run by data centers that, according to the UN Environment Programme (UNEP), use a large number of resources to keep running. UNEP cites an estimate of water usage that outnumbers that of the entire country of Denmark 6 times over, as well as hazardous electronic waste (such as mercury and lead) as a byproduct. The construction of the computers dedicated to running the AI platforms also require 800kg of raw materials for a 2kg unit. Finally, UNEP states that a single ChatGPT prompt consumes 10 times the electricity of a Google Search.

5.2.2 Inaccuracy

The tradeoff for instantaneous responses can sometimes be factual errors. Depending on the platform in use, the information being relayed may contain outdated or otherwise inaccurate information, if the platform's database is not updated regularly. In addition, ChatGPT does not always cite its sources, inhibiting users from fact checking the response for themselves. This would lead to users having to do their own research through other search engines and come to their own conclusions; invalidating the use of the AI platform in the first place.

5.2.3 Academic Misuse

As previously stated, Generative AI is capable of producing many different types of responses. Academically, students may feel inclined to take advantage of this and use AI to generate papers, essays, or codes for assignments that are meant to assess the students own skills and knowledge. Although the source is different, this misuse can be classified as plagiarism and can lead to violations of university codes and ethics if discovered, or a lack of the aforementioned skills and knowledge if not.

5.3 Examining AI Usage for Our Project

For our project, AI provided a helpful tool for gathering information on various components, as well as hardware and software guidance. Below, we will examine some of the prompts we asked our AI platform of choice (ChatGPT) and discuss how the responses we received impacted our project overall.

5.3.1 Case Study 1

In our research for the hardware of the project, we needed to consider how to supply power to our device. After researching different methods of power supplies, we decided that our best choice was a lithium-ion battery (section 3.3.12). From there, we then had to decide what specific lithium-ion battery component we wanted to use; which is where we decided to consult ChatGPT for assistance. The prompt we asked was “what would be the best lithium-ion batteries to use for the project”. The response is documented in Appendix E.1.

ChatGPT provided four options, each with six bullet points highlighting what the component was best used for, battery capacity, voltage, size, rechargeability, and a quick support for why that component should be used in our design. While the structure of the response was concise and easily understandable, two out of the four options were not what we had asked for. One of these

two options was for a battery *pack* that differed from the specific battery *cells* that we had in mind. The second extraneous option that ChatGPT provided us with was a lithium-*polymer* battery, which ran into the same problem as the previous option where it is not a type of lithium-*ion* battery and therefore does not contribute to our project.

Both cases highlight how AI can sometimes misinterpret what is being asked and can instead suggest alternatives seemingly relevant to the topic, but which do not actually apply. As both of these options were presented as fact with the real lithium-ion options, it seems that ChatGPT does not understand that these are not all the same, which demonstrates how AI can still have a gap in knowledge/understanding that can hinder the reliability and accuracy of its responses.

On the other hand, we did end up using the other two options provided in our design considerations, and one of them was ultimately the specific component we ended up choosing for our project. While ChatGPT had its issues, the response was still able to provide useful information that helped us make important decisions.

5.3.2 Case Study 2

For Chapter 4 of this report, we were required to examine different standards and constraints that our project would have to adhere to or abide by. On top of all the research required for building and designing our project, this chapter required us to consider the real-world expectations of our project as a product in a greater commercial and socioeconomic context. To expedite the research of this field, we utilized ChatGPT to assist us in finding out what specific standards and constraints would factor into a product like ours. To do this, we first fed ChatGPT a prompt explaining the objectives of our project as stated earlier in the report in Chapter 2 so that it would be able to conceptualize the product we intended to create. After that, we asked it the prompt: “What are some standards and constraints that should be considered for this product?”. The response is documented in Appendix E.2.

Similarly to the last case, the response from ChatGPT had varying degrees of utility and relevance to our project. From a constraint perspective, there weren't many insightful suggestions that we hadn't considered before, seeing as the constraints overall are mostly ones relating to our team specifically: what we are or aren't able to do with the resources and circumstances we have. Despite this, the response from ChatGPT introduced us to many electrical and software standards that we hadn't known about prior such as ISO/IEC standards on quality and the UL/IEC standards on safety.

The extraneous content in this case study highlights a general oversight that AI tends to have. For this case, ChatGPT recommended health and safety standards “for food-contact materials” by the FDA and the DNV (ISO). Upon further research, these standards only apply to food products being marketed and sold and therefore are not relevant to our project. The oversight in question lies in AI assuming food to have a larger role in our project, presumably stemming from the mention of breastmilk. As stated before, with the batteries, AI can occasionally misinterpret the intent of the prompt/user and resort to providing information that is only tangentially related to specific key words or phrases in the prompt.

5.3.3 Case Study 3

For our hardware design, we are required to provide schematics for the different components. Since the AI was able to take the summary of our project from Chapter 2 and gain an understanding of all the different parts required, we prompted ChatGPT **“Help me find reference schematics”** and it was able to provide a compiled list of websites and sources for each component/system in our design: Load Cell with HX711 Amplifier, Microcontroller (ESP32) Integration, Thermal Print Integration, and Power Management. Each section provided between 1-3 sources that ranged from YouTube videos, data sheets, and other projects and blog posts discussing the schematics and connections between our components.

As mentioned in section 5.1.1, AI can be very specific in understanding a prompt in a way that other search engines would not, pulling from the sustained knowledge it is able to acquire from conversation between the user and the AI. It is then able to expedite the process of research by applying a single prompt to several fields of relevance simultaneously and providing more curated information than a simpler search engine could. Beyond the schematics asked for in the prompt, ChatGPT was also able to provide tutorials of project connections and sample codes for software.

The only critique with this response is that the Power Management section only provided general documentation on the ESP32 instead of sources relating to the battery module or voltage regulators, as the header would imply. Overall, this assistance was able to impact the ease of research for the project by extrapolating on its knowledge of our needs.

Chapter 6 Hardware Design

Our hardware design consists of multiple core components, each playing a critical role in ensuring the system functions as intended. This section outlines the individual schematics that will be integrated into a single PCB, detailing their functionality and interconnections.

6.1 Breadboard Demonstration

To validate our design and ensure that all components function as intended, we will first implement our circuit on a breadboard. A video demonstration of our working prototype will be presented at the end of Senior Design I.

For this prototype, we will use an ESP32 microcontroller, a load cell, a touchscreen display, and an HX711 load cell amplifier module. The load cell will be mounted between two plates to form the weighing platform. It will be correctly wired to the HX711 module, which will then be connected to the ESP32. After assembling the hardware, we will install the HX711 library and implement code to display the measured weight of objects placed on the load cell.

6.2 Subsystem Block Diagram

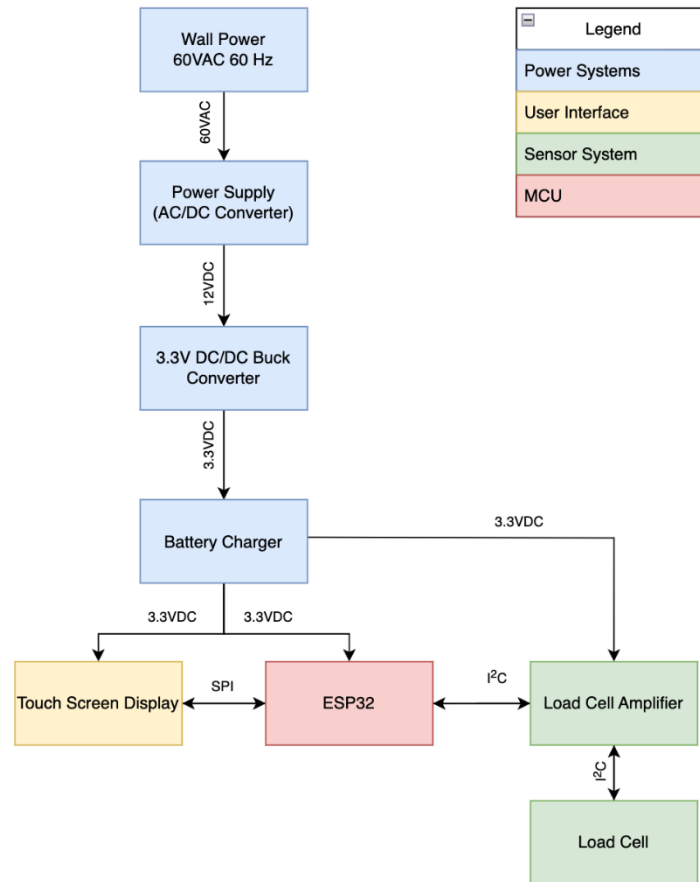


Figure 5: Subsystem Block Diagram

This block diagram outlines the key subsystems of our design. The microcontroller serves as the central unit, coordinating communication between the load cell amplifier and the touchscreen display. The device will be powered primarily through wall power, with a battery backup for portable use when not connected to an outlet. The AC power input will be converted to DC via an AC/DC converter, which will provide a 12V output. This 12V DC output will then be supplied to a 3.3V DC/DC buck converter, which steps down the voltage to power the microcontroller and other components. The buck converter's output will also be routed to a battery charger, enabling the battery to store energy and provide power to the system when not connected to wall power.

6.3 Converter Schematics

Our design incorporates various components and subsystems, each with distinct power requirements. To ensure proper voltage and current levels, we will implement both DC/DC and AC/DC converters. These converters will be integrated into our main PCB to provide stable and efficient power distribution.

6.3.1 AC/DC Converter

***note: this is a placeholder, we will be building this in Fusion360 once we receive approval**

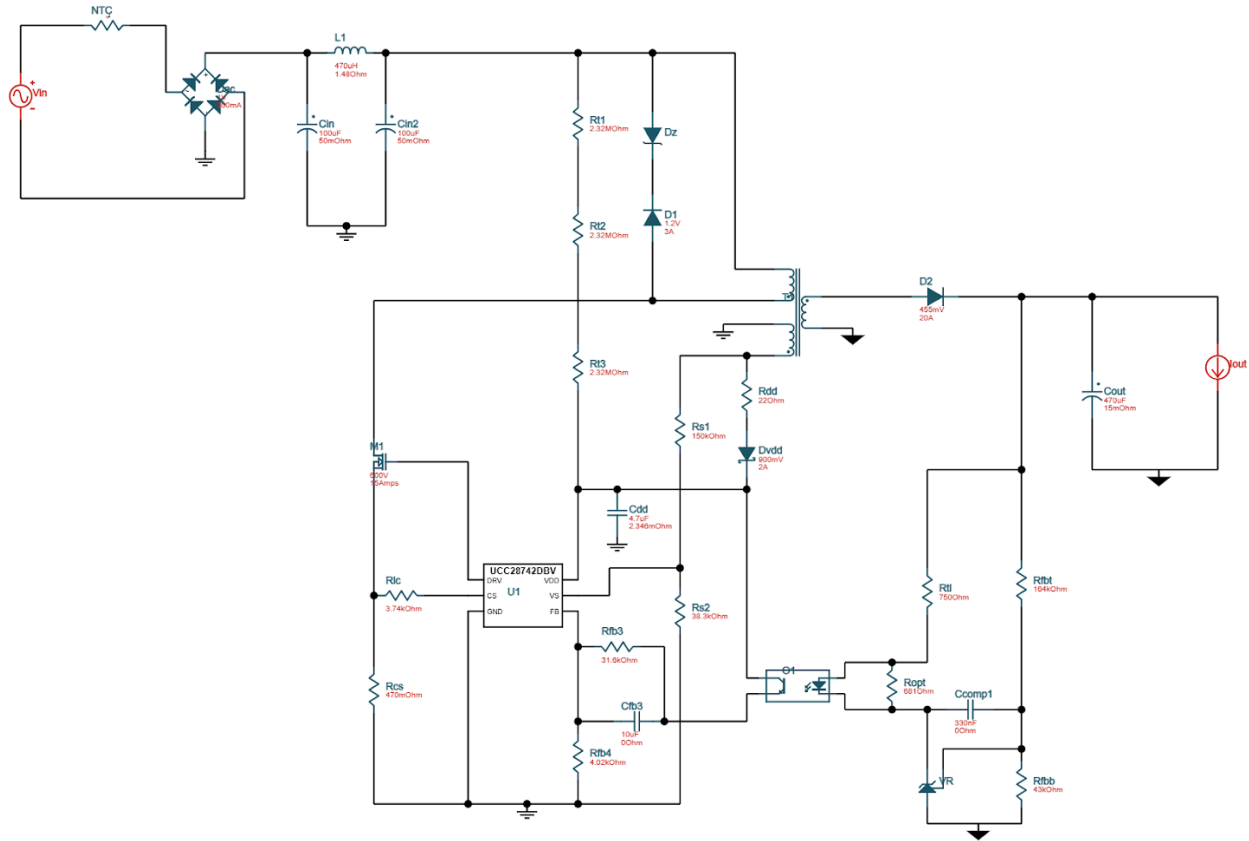


Figure 6: AC/DC Converter Schematic

Our design operates on AC wall power, which requires conversion to a suitable DC voltage for our components. To achieve this, we have implemented an AC/DC converter, designed using Texas Instruments' Webench Power Designer. The converter accepts an AC input range of 80V-160V at 60Hz and outputs 12V DC with a maximum current of 2A.

This design was selected for a balance between efficiency, size, and cost. It achieves 85.5% efficiency, has a compact 2935mm² footprint, and a bill of materials (BOM) cost of \$14.48. At the input stage, a bridge rectifier converts AC voltage into pulsating DC. The MOSFET (M1) acts as the switching element, while the UCC28742 flyback converter IC regulates the output voltage by adjusting the switching duty cycle. A flyback transformer steps down the voltage, and the output stage smooths and regulates the DC voltage. The 12V DC output from this converter will be used to power the 3.3V DC/DC buck converter, which supplies the necessary voltage to other components in our system.

6.3.2 3.3V DC/DC Buck Converter Schematic

Our battery charging module, specifically the MCP73837, provides both charging and protection functions for our system. The MCP73837 is a highly integrated, single-cell lithium-ion (Li-ion) or lithium-polymer (LiPo) battery charge management IC. It is fully compatible with our battery selection, the 18650 Li-ion battery, ensuring safe and efficient charging while protecting the battery from overcharge, over-discharge, and overcurrent conditions.

When the system is connected to wall power, the AC input is converted to DC by an AC/DC converter, which outputs 12V. This 12V DC is fed into a 3.3V DC/DC buck converter, which steps down the voltage to 3.3V. The 3.3V output from the buck converter is then provided as input to the MCP73837 battery charger. The battery charger uses this 3.3V to charge the 18650 Li-ion battery, maintaining it at an optimal charge level.

When wall power is disconnected, the MCP73837 seamlessly switches to battery power, ensuring uninterrupted operation of the system. This transition is automatic, allowing the microcontroller and other components to continue functioning without interruption. The battery's voltage is stepped down to 3.3V via the buck converter to power the microcontroller and other low-voltage subsystems.

The MCP73837 is equipped with various safety features, such as integrated thermal regulation, charge termination, and status monitoring, ensuring that the battery is charged and discharged safely while optimizing its performance and lifespan. Its integration simplifies the design, reducing the need for additional external components.

6.4 MCU Schematic (USB to UART Bridge)

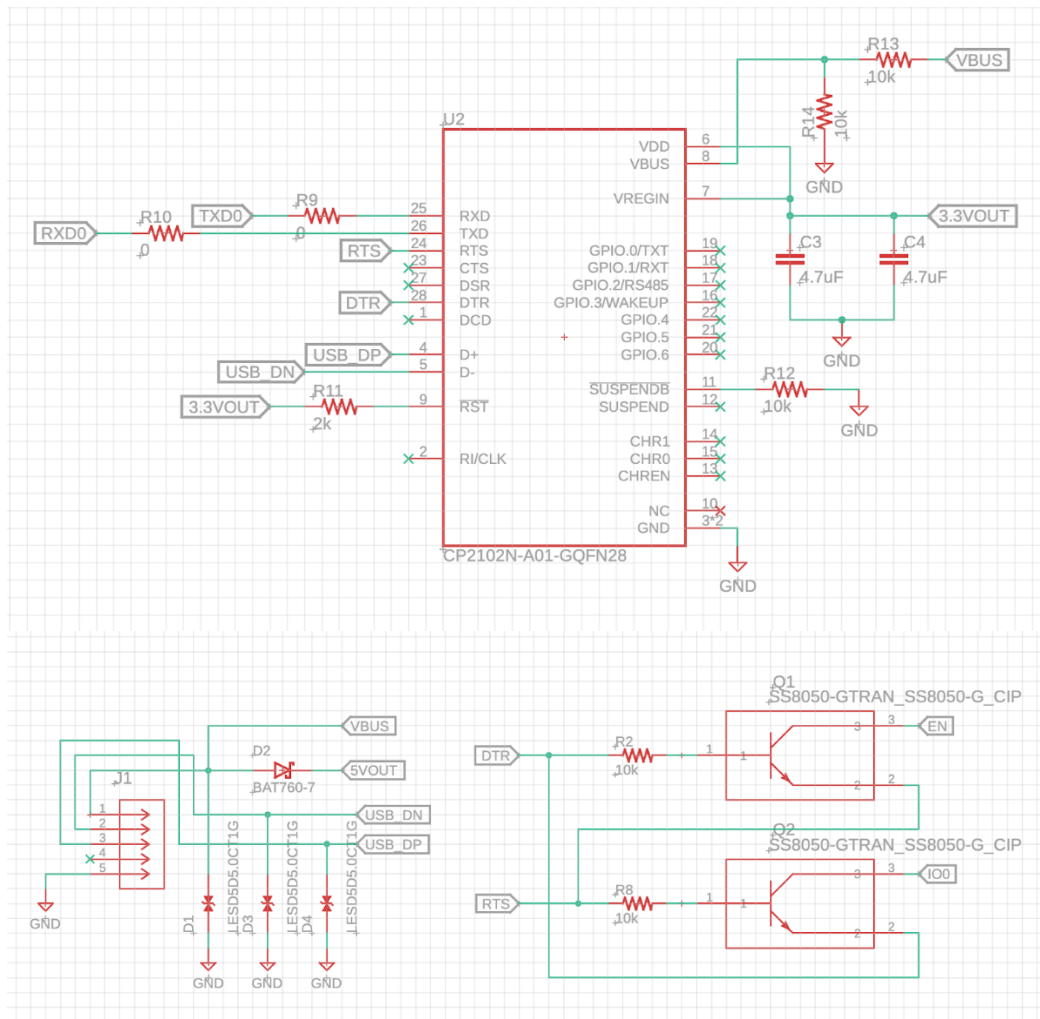


Figure 9: USB to UART MCU schematic

A USB to UART bridge is an interface module that allows communication between a USB port on a computer (or another USB host) and a UART (Universal Asynchronous Receiver-Transmitter) interface on a microcontroller, such as the ESP32-WROOM-32 that we will be implementing. The bridge converts USB signals into UART-compatible signals, enabling seamless data transmission between the computer and the microcontroller.

For our ESP32-WROOM-32, the USB to UART bridge will feature the CP2102 chip which converts the USB protocol to the UART protocol. The ESP32-WROOM-32 can communicate with this bridge to send and receive data. This is especially useful during development and debugging, as the bridge allows the ESP32 to be programmed and monitored via a computer's USB port.

In addition to providing data communication, the USB to UART bridge includes features such as power supply lines to provide 3.3V or 5V to the ESP32 and LED indicators for data transmission activity. The USB bridge will be connected to the USB port of a host (such as a computer), while

the UART side of the bridge connects to the ESP32's UART pins, enabling serial communication for uploading firmware or for use in serial debugging.

6.5 Touchscreen Schematic

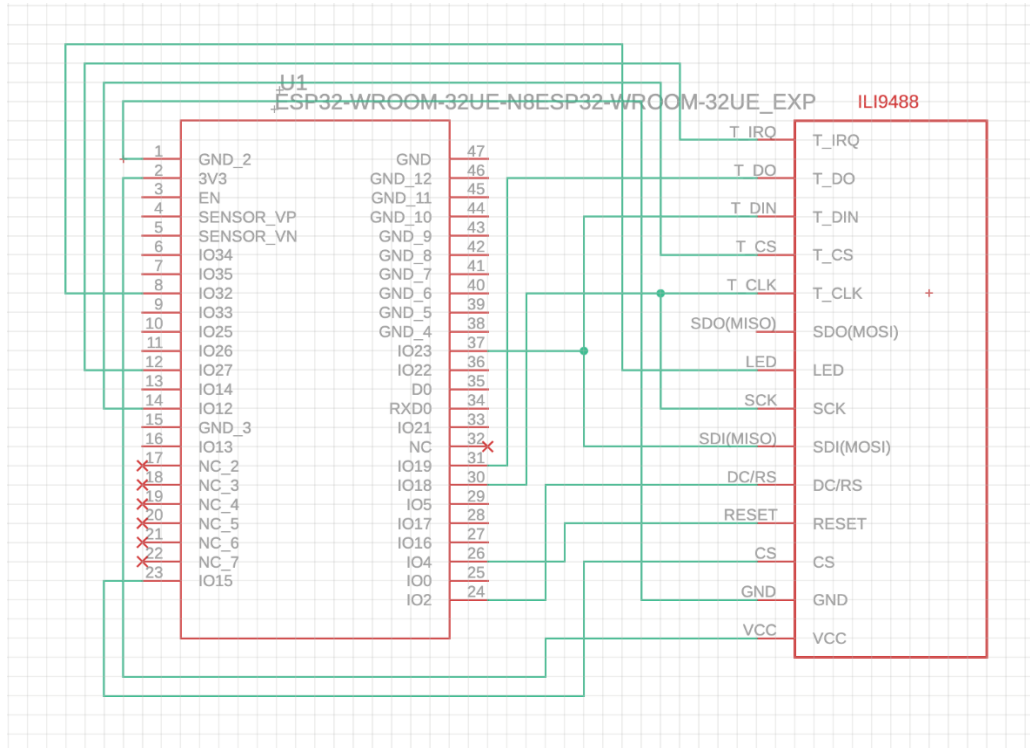


Figure 10: Touchscreen to MCU schematic

For our touchscreen interface, we are implementing the ILI9488 TFT LCD display alongside the ESP32-WROOM-32 microcontroller. In our schematic, the display is positioned on the right side, while the ESP32-WROOM-32 is on the left. The ESP32 communicates with the ILI9488 via SPI (Serial Peripheral Interface), enabling both display functionality and touchscreen input. The ILI9488 is a TFT LCD display with an integrated ILI9488 driver IC, which controls the pixel array and renders images. Communication between the ESP32 and the display occurs through a 4-wire SPI interface, utilizing GPIO 18 (SCK) for clock synchronization, GPIO 23 (MOSI/SDI) to send image data, GPIO 5 (CS) to enable communication, and GPIO 15 (DC/RS) to distinguish between command and pixel data. The RESET pin (GPIO 4) ensures the display starts in a known state, while the LED pin (GPIO 2) controls the backlight. The MISO/SDO pin is intentionally left unconnected, as data transfer from the display back to the ESP32 is not required.

In addition to display functionality, the ILI9488 also integrates a resistive touchscreen, which operates separately within the SPI framework. The touchscreen functionality is enabled by connecting the T_IRQ pin (GPIO 33) for detecting touch events, T_DO (GPIO 25) for transmitting touch data, T_DIN (GPIO 23) for receiving data from the ESP32, T_CLK (GPIO 27) for synchronizing touch data, and T_CS (GPIO 14) for selecting the touchscreen controller. These connections allow the ESP32 to accurately interpret touch inputs while simultaneously rendering graphical content.

By properly wiring these connections, the ESP32 controls the multiplexed rows and columns within the TFT panel, activating specific pixels to render images. The integrated touchscreen controller provides an interactive interface, making it possible for users to interact with the display. With its high-speed SPI communication and efficient control signals, this interface ensures a responsive graphical display with touch capabilities, which will be essential for our portable digital scale user interface.

6.6 Weight Sensor Schematics

Our design incorporates load cells as weight sensors for the scale. These load cells, which detect applied weight, will be interfaced with a load cell amplifier to ensure accurate signal processing. The load cell amplifier will amplify the weak signal output from the load cells, allowing the microcontroller to accurately measure and interpret the weight data. This amplified signal is then sent to the microcontroller, which will process the data for display and further use in the scale's functionality.

6.6.1 Load Cell Amplifier Schematic

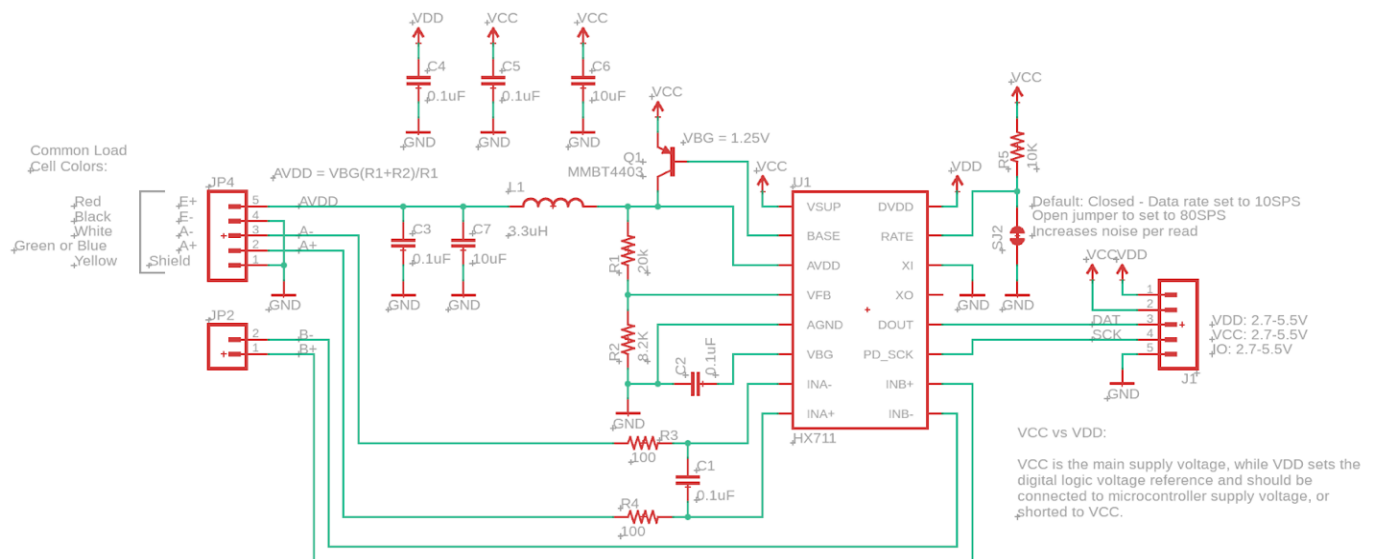


Figure 11: Load Cell Amplifier Schematic

This schematic represents the load cell amplifier circuit, which serves as the interface between the load cell and the microcontroller. The load cell is connected on the left side of the schematic, with its pins clearly marked. Each wire from the load cell is color-coded and corresponds to a specific pin on the amplifier: Excitation (+), Excitation (-), Signal (+), and Signal (-). These connections feed into the HX711 load cell amplifier, which processes the small differential signals produced by the load cell.

The HX711 utilizes INA+, INA-, INB+, and INB- as its differential input channels for reading the signal from the load cell. AVDD serves as the analog power supply, while VBG is the internal reference voltage. The amplifier takes the extremely small output signals from the load

cell and amplifies them for accurate measurement. Communication with the microcontroller is handled via PD_SCK (clock input) and DOUT (data output), which function as part of an SPI-like communication protocol.

On the right-hand side of the schematic, a 5-pin header connects the amplifier to the microcontroller. This header consists of VCC (3.3V), VDD (5V), serial data (LD_CELL), serial clock (SCL), and ground (GND). While most of these pins are common, the serial data pins (LD_CELL and SCL) specifically handle communication between the amplifier and the microcontroller. This setup ensures that the load cell's signals are accurately processed and transmitted, enabling precise weight measurements in our system.

6.6.2 Load Cell Schematic

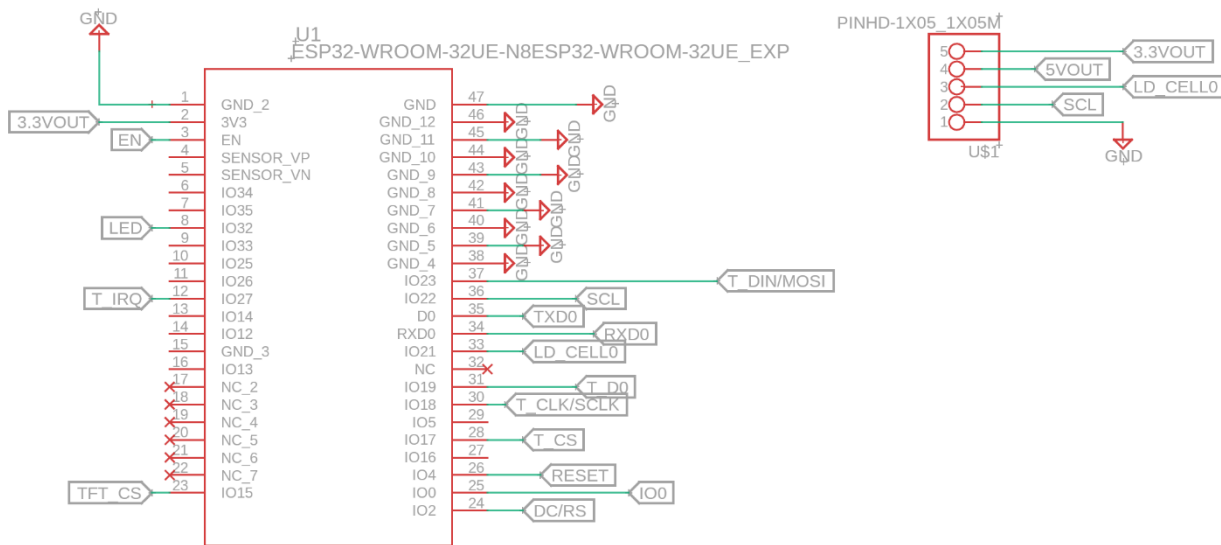


Figure 12: Load Cell Schematic

Our load cell will be configured in a Wheatstone bridge arrangement, which ensures accurate measurement of small changes in resistance due to applied weight. The amplified output from the load cell will be fed into the HX711 ADC (Analog-to-Digital Converter) module, which will convert the analog signals into digital data. This digital data will then be transmitted to the microcontroller via the HX711 output header. The microcontroller will read the digital signals from the HX711, allowing it to process and display the weight measurements.

6.7 LED Schematic

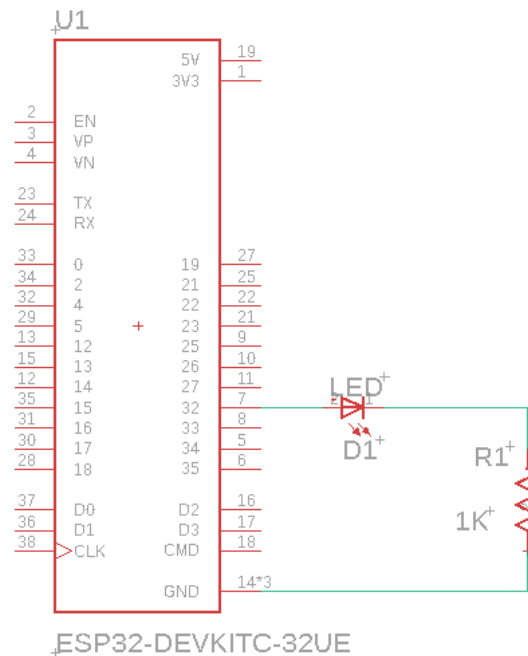


Figure 13: LED to MCU schematic

We will be integrating LEDs into the design to provide real-time system status feedback. Two key indicators will be implemented: the Battery Low LED and the System Online LED. The Battery Low LED will light up when the system's battery voltage falls below a predefined threshold, alerting the user that the battery needs to be recharged or replaced. The System Online LED will indicate that the system is powered on and functioning correctly, remaining illuminated when the system is active and performing as expected. These LEDs will be connected to the microcontroller and controlled based on specific system conditions, offering clear visual indicators of the system's status at any given time. The schematic provided illustrates how we plan to integrate the LEDs into our design once it is finalized.

6.8 Printer

The printer will be integrated into our design at a later stage of the project, once we have finalized the core functionality of the digital scale system. Our plan is to connect the printer directly to the ESP32 microcontroller, which will serve as the central hub for communication between all components. By directly linking the printer to the microcontroller, we aim to facilitate seamless communication for both sending and receiving weight data, allowing the printer to generate labels based on the real-time weight measurements captured by the load cell. This integration is critical to the overall functionality of the system, as it will allow for the automatic printing of weight labels once the weight has been measured and confirmed through the touchscreen interface. The data transfer between the ESP32 and the printer will be managed through efficient communication protocols, with the ESP32 acting as the intermediary between the load cell amplifier, touchscreen, and printer. By utilizing standard communication protocols

like I2C, the integration process can be streamlined, ensuring that all components work in tandem without significant delays or interference.

Chapter 7 Software Design

From the software perspective, platform capability is a critical factor in ensuring accessibility. The choice of programming language and development framework determines whether the mobile app will be Android-only, iOS-only, or cross-platform. Developing a native application specifically for Android (using Java or Kotlin) or iOS (using Swift) allows for optimal performance and full access to device-specific features. However, this approach increases development time and ongoing maintenance due to the need for separate codebases.

Alternatively, using a cross-platform solution like Flutter or React Native streamlines development by enabling a single codebase for both platforms. While this simplifies deployment and reduces development effort, it comes with trade-offs such as limited Bluetooth connectivity support and reduced performance compared to native applications.

Implementing real-time data syncing with offline functionality adds another layer of complexity, particularly when receiving live weight data from the scale and managing dynamic inventory updates. The choice of platform and framework directly affects how the app will be maintained, scaled, and updated over time.

As previously mentioned, the application could be kept minimal by focusing on core features. However, considering the potential for future commercialization, it is also essential to design with manufacturability, scalability, and long-term software maintainability in mind.

7.1 Mobile Application Development

The pumped project aims to create a portable scale and label maker for breastmilk with a mobile application that can directly receive information from the device. To begin developing this application we need to establish user requirements for the targeted market. The primary function of this application is to provide caregivers with a seamless way to measure, log and label breastmilk storage.

The device will be able to transmit data after the label is printed regarding the breast milk to the mobile application via Bluetooth. This allows for real-time weight-management and automated label generation. Important features of the application will include users being able to edit their milk logs, notified of upcoming expirations as well as being able to predict when more milk will be needed. These essential functionalities are designed to meet the needs of users by diminishing manual input errors, improper labeling, and inaccurate storage inventory. By establishing the user's needs, it will make the integration of our software application easy to develop and an enhanced experience for all users.

Since the target audience is breastfeeding mothers, the application needs to be easy to navigate and have adaptive features to create a customized environment for each mother. The user interface will be designed with simplicity and accessibility in mind to ensure ease of use for all individuals. Adding features like voice commands and high-contrast displays will further improve accessibility. As mentioned before, to recognize the diverse needs of caregivers, the

application will offer customization options for labeling preferences, storage duration reminders, and feeding logs. Users will be able to create multiple profiles for each infant allowing more personalized tracking of milk intake and storage for each child. By providing this feature, it ensures caregivers can efficiently manage milk storage and feeding schedules.

The application will also have a history log that enables users to keep track of past measurements and storage. Utilizing this data, users will receive notifications about upcoming milk expirations, low inventory alerts when more milk may be needed, and easy access to storage history and expiration dates. With these notifications it will help users adhere to best practices for milk preservation. Overall, by focusing on user's needs and expectations, we aim to enhance the efficiency of milk tracking and labeling. Through thoughtful design and functionality, the application will seamlessly integrate with the breastmilk scale and labeler.

7.1.1 Mobile Application

Our mobile application follows a structured workflow to ensure seamless logging, inventory management, and label printing functionalities. The process begins when a user opens the application and encounters the login screen. The user enters their credentials and, if correct, is directed to the milk log homepage. If the credentials are incorrect, they are given the option to reset their password. Upon selecting this option, the user is taken to a verification page where they must enter their email address. An email is then sent with a verification link that the user must click to confirm their identity. After verification, the user can return to the app, change their password, and will then be prompted to log in again.

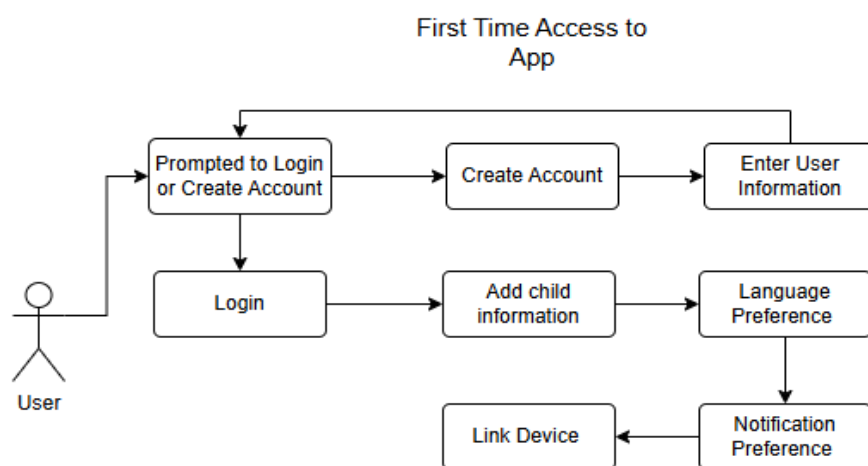


Figure 14: First Time Access to App Flowchart

Once logged in, the main page displays the milk inventory log. Below the logs, users are presented with a variety of options to perform tasks such as adding, editing, and deleting logs, printing labels, and modifying application settings. Each of these functions is essential for maintaining accurate inventory and tracking milk usage. Before using the edit, delete, or print functions, the user must first select the specific log they want to act upon. After each process is completed, the user is returned to the homepage. If the user wishes to return to the main page without completing a process, they can use the back option to navigate away. A description of the main functions showcased in the flowchart, which users will interact with, is provided below.

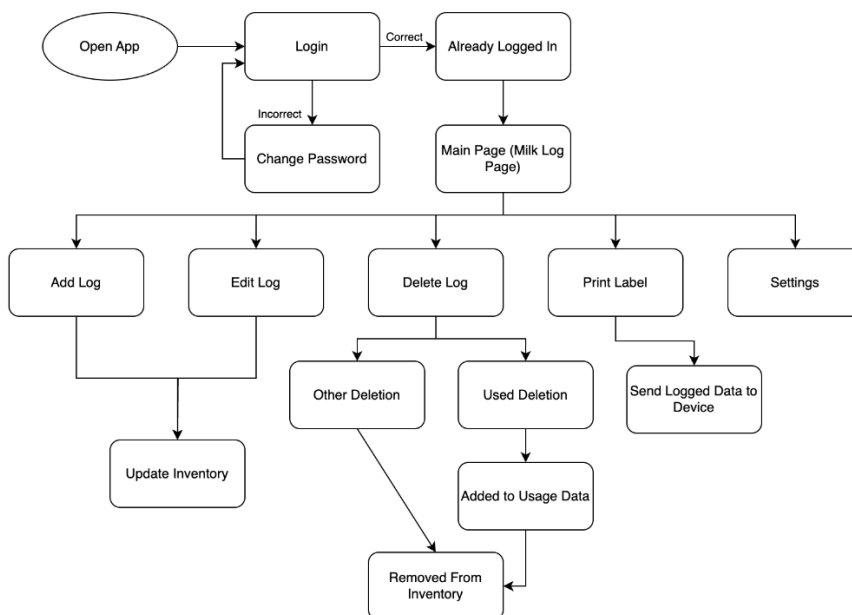


Figure 15: App Functionality Flowchart

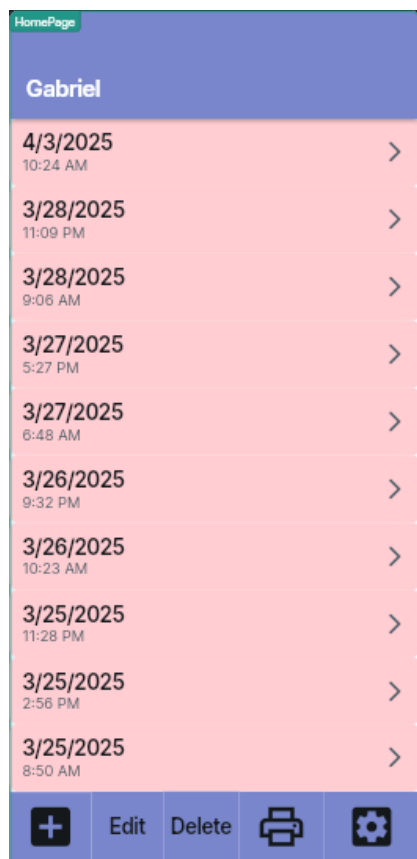


Figure 16: App Prototype

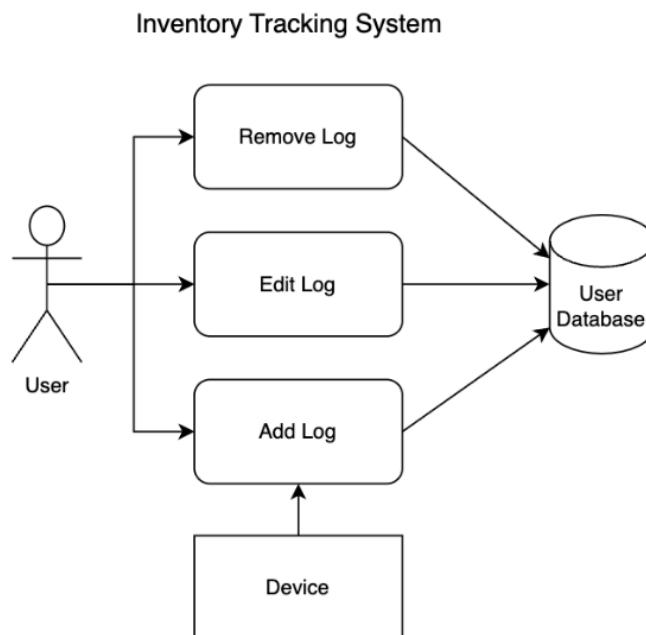


Figure 17: Inventory Tracking System Flowchart

Add_log Function:

After selecting the Add option in the mobile application users will be redirected to a new screen that will prompt them basic information about the milk log including volume, date and time. In order to move on to the next part in the process users must enter values for the volume and date or they will not be able to proceed. Once that information is entered users are then asked to enter dietary details on a new page along with any other notes they would like to leave about the milk. Once they have finished entering all the necessary information, they will confirm it is correct and then the new log will be added to the milk inventory page and user database.

Edit_log Function:

This function will act similarly to the add option. After clicking on the Edit option users will be directed to a new screen that displays all the basic information about the milk such as volume, date and time which they will be able to edit. Then it will take them to the page that showcases the dietary details and notes about the specific log which they can edit at this point. After confirming the details of the milk log it will be updated in the user database and on the main page.

Delete_log Function:

This function is very simplistic in comparison to the previous two described because it only requires two steps. After selecting the log and then selecting the delete option the user will be directed to a screen showcasing the details of the milk log. There will be a back button feature throughout this whole process so if at any point the user needs to return to the previous page, they can utilize it. After reviewing the details, the user will be prompted with the type of deletion it is, used or other. The reason for us having two types is because used deletion will report this data to our usage database. The other is for when users make a mistake and don't want that data added to the usage database. The importance of this is because the usage database will be used to predict future milk needs, so the data stored in it needs to be accurate. Once the type of deletion is confirmed the deletion process will complete and delete that log from the milk inventory and user database.

Print_log Function:

This function will behave similarly to the delete option with its simplistic approach. Once the user selects the print option they will be taken to a page where they can review the details of the log. After doing so they will just confirm they want to print, and the data will be sent to the device.

Settings Function:

Out of all our functions this one will be the most in depth because it will have multiple subsystems. To begin, after the user selects the Settings option a new page with sub options will appear. These options include Account Information, Preference Settings, General Settings Link Device and Logout. Below is a description of each subsection.

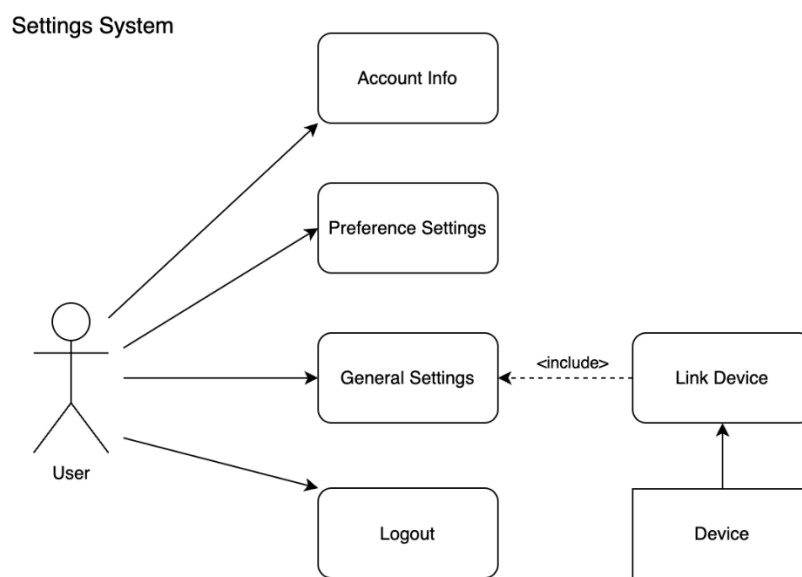


Figure 18: Settings System Flowchart

Account Information:

When selecting account information users will be taken to a page where they can view personal information such as username, email registered with the account and an option to change password. The user is unable to edit the account information from this page. If change password is selected, then they will go through the same process as when they change their password from the login page. There will also be a section within personal info where you can put information about your child including age, gender, weight, and how much milk they typically consume to help track feeding, scheduling and habits.

Preference Settings:

The flowchart below distinguishes the sub options that will be included under Preference Settings. These settings represent the more personalized settings for users such as dietary restrictions feeding schedule and notifications. When dietary restrictions are selected users can add or take away additional options for the dietary notes showcased in the logs. Through the feeding schedule option, the user can view the usage database and can decide to delete data from it as well. The last option is notifications where the user can select the type of notifications they want to receive such as expiration and future milk needs. Through this they can also set when they want the notifications to occur. For example, if they want expiration notifications, they can say that a month away from the pumped date is when the milk is expired and that is when they want the notification.

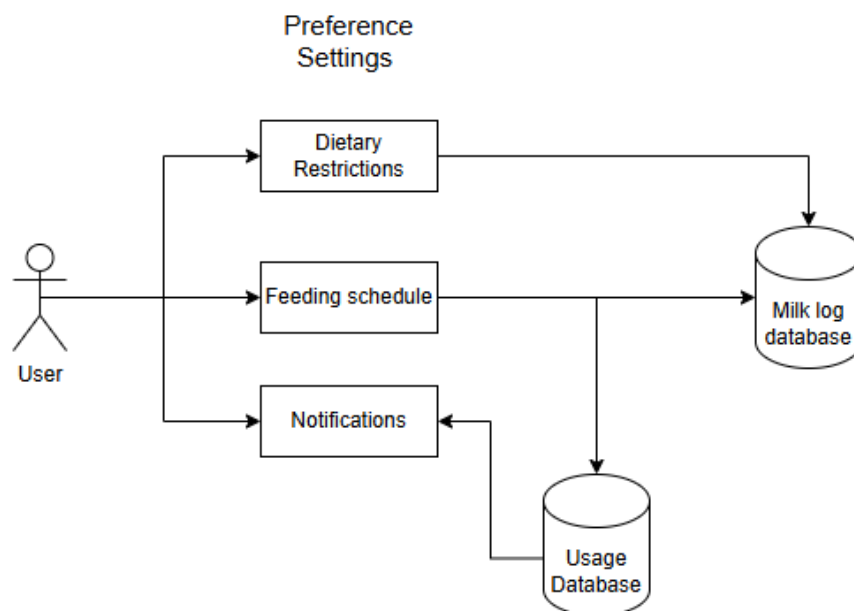


Figure 19: Preference Settings Flowchart

General Settings:

The general settings would display options to modify language preference, access linked devices to the application and manage app alert system. Through the Linked devices option the user would be able to see device that is paired and pair additional devices via Bluetooth. The management for app alert systems will allow users to modify how often they want alerts and to mute alerts from the app.

Logout:

After selecting the logout option under the settings users will be redirected to the login screen.

Table 16: Function and Variable Chart

Label	Function or Variable	Description
Add_log	Function	Users enter essential details such as volume, date and dietary information. Once data is confirmed the log is added to inventory and user database.
Edit_log	Function	Allows users to modify existing milk log entries including volume, date and dietary information. Updated logs are saved in the user database.
Delete_log	Function	Users review log details before selecting either Used Deletion or Other deletion.
Print_log	Function	Users verify log details before sending the data to the device.
Settings	Function	Provides access to account information, preferences, general settings and logout option.
Log	Variable	Represents an individual milk entry containing all relevant data such as user volume, date and dietary information.

7.2 Device Software

The software architecture for our device is structured around an ESP32-WROOM-32UE microcontroller, which controls the input, output, and communication across all peripheral subsystems. The overall firmware will operate in a loop-based execution model, consisting of event flags and state monitoring. These will be utilized to ensure responsive interaction with the user and reliable control of the hardware components. Below our use case diagram for the device can be seen describing how we anticipate the device to be used daily. The user begins by placing the breastmilk bag on the scale. The device will then read the weight utilizing the load cell system. After the weight is read and stored the user will then be prompted to enter dietary details on the LCD touchscreen. For entering the dietary information, we will be using abbreviations for specific allergens so that they can fit onto the label. Once the user has entered the dietary information they can then proceed to confirm if the data is correct. If the data is incorrect, they will be prompted to enter the dietary information again, else, they will proceed to confirmation of print. After they confirm they would like to print the label will be printed and the data will be transmitted to the mobile application log database via Bluetooth. If the user chooses not to print, they will be redirected again to enter dietary information.

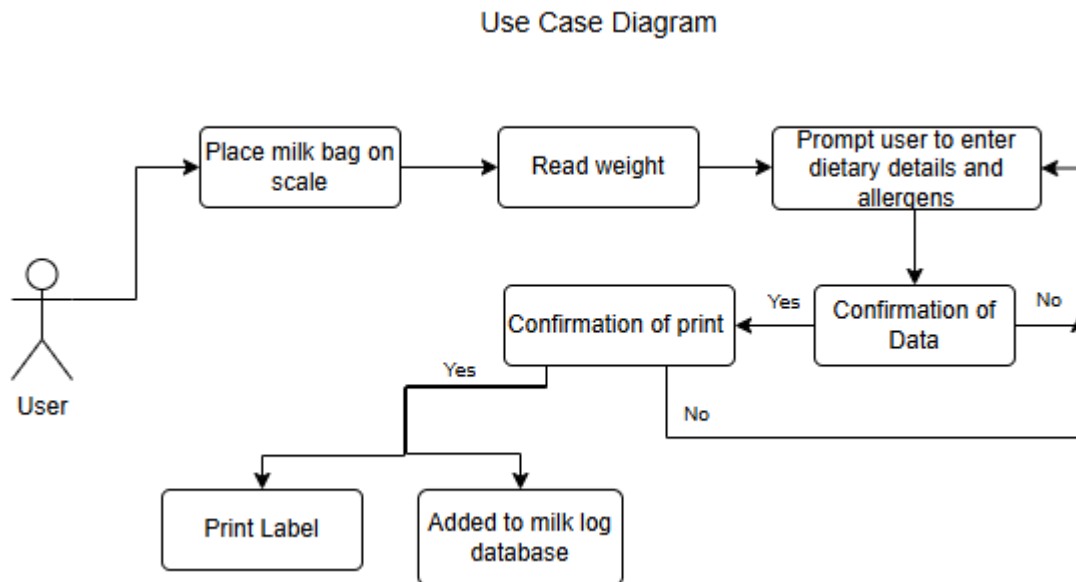


Figure 20: Use Case Diagram

Each subsystem is organized into modules in software to encourage code clarity and expandability. Below is an overview of the individual components and their software logic.

7.2.1 Load Cell Subsystem

For our project, we will use an HX711 load cell amplifier to process weight measurements from the load cell. The HX711 will amplify the small analog signal from the load cell and convert it into a digital signal, which will then be read by the ESP32-WROOM-32UE via I²C communication.

To ensure the system operates safely, we will integrate overload detection to prevent exceeding the scale's maximum weight capacity. This feature will continuously monitor the weight

readings and trigger a warning, such as a notification on the touchscreen, if the measured weight surpasses the predefined safety threshold.

By implementing this setup, we will ensure accurate weight measurements while also protecting the hardware from potential damage due to excessive loads. Depending on our application needs, we may also incorporate additional safety measures, such as disabling further measurements or triggering an alert sound in case of overload detection.

7.2.2 LED Subsystem

To integrate our LEDs, we will use global status variables to determine which LED should be activated. Each LED will be connected to a dedicated GPIO pin on the ESP32, ensuring precise control over system indicators. The LEDs will operate in an active high configuration, meaning a HIGH (1) signal will turn the LED on, while a LOW (0) signal will turn it off. Our system will use flag-based control, where predefined values will correspond to specific LED states.

To optimize performance, the firmware will include a dedicated function to update LED states based on system conditions. This function will check the global status variable and adjust the corresponding GPIO outputs accordingly. Additionally, to prevent flickering and ensure efficient power management, the system will update LEDs only when a status change occurs, rather than continuously polling in the main loop.

7.2.3 Touchscreen Subsystem

The graphical user interface (GUI) is a vital component of our design, serving as the primary bridge between the user and the device. It guides users through the process of labeling and logging breastmilk with minimal effort. The GUI design prioritizes clarity and ease of use while continuously monitoring key system statuses such as weight detection, Bluetooth connectivity, and battery life.

April 4, 2025		10:32 AM
0.00 oz		Mode
		Settings
		Restart
Print	Back	Sleep Mode

Figure 21: Homepage draft

The main feature displayed on the home screen is the print function. This allows users to print a label containing the date, time, milk volume (in fluid ounces), and any selected dietary or allergen notes. In addition to the print option, users can access other features from the home

screen, including Mode, Settings, Sleep Mode, and Reset. The Mode button enables switching between different weight conversion units. The Settings menu includes options for Bluetooth pairing, date and time adjustments, scale calibration, backlight brightness, and printer information. Sleep Mode puts the device into a low-power state to conserve energy, while the Reset button clears the current scale reading displayed on the home screen.

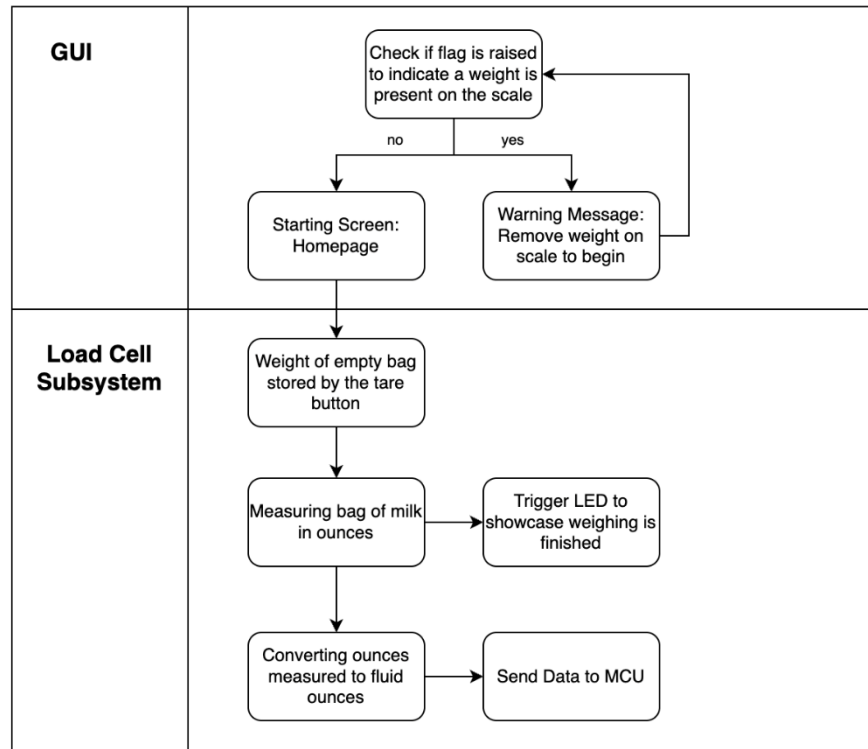


Figure 22: GUI and Load Cell Subsystem integration

The scale reading is prominently displayed on the home screen. It is obtained from the HX711 Load Cell Amplifier and processed through the ESP32's ADC, which converts the raw analog signal into a stable, real-time weight measurement displayed in ounces. Once a non-zero weight is detected, a flag is raised to signal that an object is present on the scale. This flag is monitored by the GUI to determine whether to proceed with the tare and measurement functions or to prompt the user with a warning message. If the flag is raised prematurely, such as when weight is detected before the tare process begins—the GUI displays a warning instructing the user to remove any objects from the scale. Ensuring each session begins with a clear baseline, the system allows the weighing process to proceed only after the tare is complete and a weight is added. The flag guides the system through the correct sequence, triggers an LED to indicate that weighing is complete, and prepares the measured data for conversion and display.

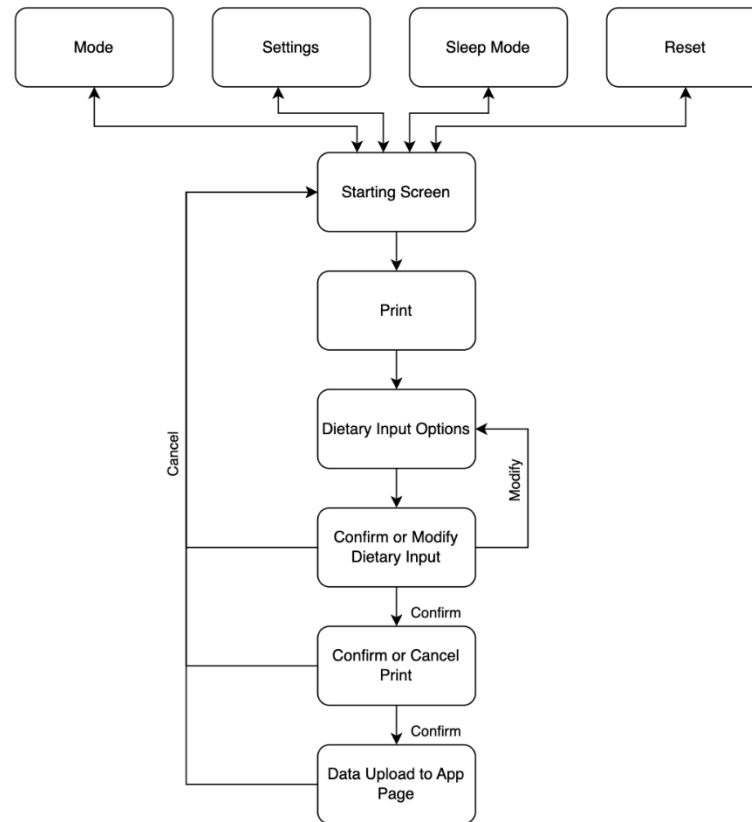


Figure 23: Touchscreen Flowchart

The touchscreen flowchart outlines the sequential logic the user follows when starting and completing a labeling session. After the print function is selected from the home screen, users are guided through a series of screens that allow them to choose and confirm dietary or allergen notes to be printed on the label. Once the dietary input is confirmed or modified, the user is prompted to either confirm or cancel the print. This multi-step confirmation process ensures that all relevant information is accurately captured before printing and reinforces a user-friendly, error-resistant experience. To further reduce the likelihood of incorrect input, the touchscreen interface is programmed to recognize touch only within clearly defined screen regions associated with selectable options. Touches outside these active zones are ignored, helping prevent unintended actions. Upon final confirmation, the label is printed and the data recorded from the session is uploaded to the app, completing the workflow.

7.2.4 Printer Subsystem

Our system will use the ESP32 to facilitate communication between components, specifically leveraging I2C for data transfer. The ESP32 will receive weight data from the load cell amplifier, which is connected via I2C. Once the weight data is successfully captured and processed by the ESP32, it will communicate this information to the printer.

The communication process is designed to be seamless: once the touchscreen user interface confirms the print command, the data will be transferred from the ESP32 to the printer. The printer will then utilize the data to generate a label that reflects the current weight and other relevant information, such as time or product details. The I2C protocol is chosen for this

interaction due to its simplicity, low power consumption, and ability to efficiently handle communication between the ESP32 and the printer, as well as the potential for multiple devices on the same bus.

Furthermore, once the print command is confirmed on the touchscreen, the app will receive the data to monitor the status and ensure that the printing process is successful. This adds a layer of feedback for the user, allowing them to track whether the print job has been completed and if any issues arise, such as errors during communication or printing.

This approach ensures that the system is both user-friendly and highly functional, allowing for seamless integration of weight data transfer, user confirmation, and label printing in a connected ecosystem. The use of the ESP32 with I2C helps keep the communication efficient and reduces complexity, which is especially important for ensuring accurate and timely printing.

Chapter 10 Administrative Content

10.1 Budget

The budget provided to us for this project is \$500. This money is funded by us through our sponsor, and our goal is to be as cost-efficient as possible while meeting all our goals in our prototype. We are still updating our project cost table, but the list below has all the known current supplies needed.

Table 17: Estimated Pumped Project Budget

Product Name	Cost
Microcontroller	\$10-\$35
Load-Cell	\$6-\$10
Touchscreen Display	\$20-\$60
Thermal Label Printer	\$30-\$120
Thermal Label Paper	\$10-\$30
Power Supply	\$20
PCB Designs	\$30-\$50
LED	\$0.10-\$0.70
Load-Cell Amplifier	\$40-\$150
Rechargeable battery	\$9-\$20
Battery Charging IC/module	\$5-\$20
Voltage Regulator	\$1-\$5

10.2 Bill of Materials

The Bill of Materials (BOM) will outline all the components and materials we need for our project. This will help us keep track of resources and manage costs.

Table 18: Bill of Materials

Component	Unit(s)	Manufacturer	Manufacturer Part number	Price
3.5 Inches TFT LCD Touch Screen	1	Hosyond	ILI9488	\$16.99
Microcontroller	3	Espressif Systems	ESP32-DEVKITC-32UE	\$24.27
Load-Cell	2	SparkFun Electronics	SEN-13329	\$19.38
Buck Voltage Regulator	4	Texas Instruments	TPS561208DDC R	\$2.08
PRIMUZ Micro Thermal Printer	1	PRIMUZ	MC206H_12V	\$46.88
Thermal Label Paper	3	Zonon	HU-96587-H	\$6.99
PCB Design				
LED(blue) 0805 SMD	8	Würth Elektronik	150080BS75000	\$1.36
LED(green) 0805 SMD	8	Würth Elektronik	150060VS75000	\$1.36
LED(red) 0805 SMD	8	Würth Elektronik	150080SS7500	\$1.36
Loadcell-Amplifier HX711	2	SparkFun Electronics	SEN-13879	\$21.90
18650 Li-ion Battery	10	18650batterystore		\$35.00
IC BATT CHG LI-ION 1CELL 10MSOP	4	Microchip Technology	MCP73837-NVI/UN	\$6.04

10.3 Distribution of Work

Our goal is to distribute our work in an efficient way, ensuring each team member can focus on their strengths and contribute to the project in a meaningful way. By breaking down our tasks into manageable portions, we can stay organized and make progress towards our end goal.

Table 19: Distribution of Work

Requirements	Responsible Team Member(s)
--------------	----------------------------

Shell CAD Design	Johan, Simran
Mobile Application Design Development	Alexandra, Mya
Device Software Design and Implementation	Mya, Alexandra
Printed Circuit Board Design	Simran, Johan
Weight Sensor Design and Implementation	Johan, Simran
Mobile Application Debugging	Mya, Alexandra
Battery Intergration and Management	Simran
LED Control System	Simran, Mya
Display Intergration	Alexandra, Johan
Wireless Communication	Alexandra, Mya
Thermal and Durability Testing	Simran, Johan
Testing and Debugging	All

10.4 Milestones

In Senior Design I, we have a timeline of 16 weeks to research and finish our documentation for our prototype. Every week we are planning to update our documentation based on developments with our software and hardware development. We will also be meeting with our sponsor regularly to ensure our direction aligns with their requirements.

Table 20: Pumped Project Timeline

Weeks	Important Tasks + Due Dates
1 (Jan. 5 th – Jan. 11 th)	Form Group and Discuss Ideas
2 (Jan. 12 th – Jan. 18 th)	Finalize Idea and begin Divide & Conquer Document
3 (Jan. 19 th – Jan. 25 th)	Meet with Sponsor, Divide & Conquer Document Due
4 (Jan. 26 th – Feb. 1 st)	Divide & Conquer Meeting and Revisions Research the software needed to create the app.
6-7 (Feb. 9 th – Feb. 22 nd)	Research scale and label printer designs
8 (Feb. 23 rd – Mar. 1 st)	Start PCB design
9-10 (Mar. 2 nd – Mar. 15 th)	Continue software and PCB development
11 (Mar. 16 th – Mar. 22 nd)	Continue software and PCB development
12 (Mar. 23 rd – Mar. 29 th)	Midterm Report and Meeting
13 (Mar. 30 th – Apr. 5 th)	Midterm Report Revisions and Upload

14-15 (Apr. 6 th – Apr. 19 th)	Continue Report Revisions
16 (Apr. 20 th – Apr. 26 th)	Final Documentation

Appendix A: References

1. **3DPrinting.com**. *3D Printing Software*. [Online]. Available: <https://3dprinting.com/software/>. Accessed: 12 Mar. 2025
2. Apple Inc. Swift. Apple Developer, <https://developer.apple.com/swift/>. Accessed 10 Mar. 2025.
3. **ASP, Inc.** *ASP 53037 Material Safety Data Sheet (MSDS)*. [Online]. Available: <https://files.batteryjunction.com/frontend/files/asp/msds/ASP-53037-MSDS.pdf>. Accessed: 2 Mar. 2025.
4. **Avnet**. *Comparing Common MCU Interface Protocols*. [Online]. Available: <https://www.avnet.com/wps/portal/us/resources/article/comparing-common-mcu-interface-protocols/>. Accessed: 20-Feb-2025
5. Bitbucket: Key Terms to Know. Bitbucket, <https://bitbucket.org/product/guides/getting-started/overview#key-terms-to-know>. Accessed 15 Mar. 2025.
6. Barcodes Inc. Zebra Breastmilk Tracking Solutions. Barcodes Inc., <https://www.barcodesinc.com/zebra/breastmilk-tracking.htm>. Accessed 2 Feb. 2025.
7. **Batteries, Inc.** *Primary vs. Secondary Batteries*. [Online]. Available: <https://batteriesinc.net/primary-vs-secondary-batteries/>. Accessed: 2-Mar-2025
8. **Bluetooth SIG**. *Bluetooth Specifications*. [Online]. Available: <https://www.bluetooth.com/specifications/specs/>. Accessed: 10-Mar-2025
9. Codecademy. "The History of Chatbots." Codecademy, <https://www.codecademy.com/article/history-of-chatbots> Accessed 13 Mar. 2025.
10. **Consumer Product Safety Commission (CPSC)**. *Battery Regulations and Standards*. [Online]. Available: <https://www.cpsc.gov/Regulations-Laws--Standards/Voluntary-Standards/Topics/Batteries>. Accessed: 2-Mar-2025
11. **Constellation**. *Solar Energy: Pros and Cons*. [Online]. Available: <https://www.constellation.com/energy-101/energy-innovation/solar-energy-pros-and-cons.html>. Accessed: 2-Mar-2025
12. Cubtale. Smart Milk Inventory Tracking Stickers. Cubtale, <https://www.cubtale.com/products/smart-milk-inventory-tracking-stickers>. Accessed 4 Feb. 2025.
13. DNV. ISO 22000 - Food Safety Management. DNV, <https://www.dnv.com/services/iso-22000-food-safety-management-5173/#:~:text=The%20ISO%2022000%20%2D%20food%20safety,as%20statutory%20and%20regulatory%20demands>. Accessed 13 Mar. 2025.
14. **Eaton**. *Supercapacitors vs. Batteries*. [Online]. Available: <https://www.eaton.com/us/en-us/products/electronic-components/topics/supercapacitors-vs-batteries.html>. Accessed: 2-Mar-2025.
15. **EEMB**. *ASR00050 18650 2500mAh Datasheet*. [Online]. Available: https://www.mouser.com/datasheet/2/855/ASR00050_18650_2500mAh-3078640.pdf. Accessed: 2-Mar-2025
16. **Electronics For You**. *E-Paper Display: Advantages & Disadvantages*. [Online]. Available: <https://www.electronicsforu.com/technology-trends/learn-electronics/e-paper-display-advantages-disadvantages>. Accessed: 4-Mar-2025.
17. **Electronics For You**. *LCD (Liquid Crystal Display) Basics*. [Online]. Available: <https://www.electronicsforu.com/technology-trends/learn-electronics/lcd-liquid-crystal-display-basics>. Accessed: 4-Mar-2025

18. "Firebase Realtime Database." GeeksforGeeks, <https://www.geeksforgeeks.org/firebase-realtime-database/>. Accessed 7 Mar. 2025.
19. "Getting Started with Cross-Platform Mobile Application Using Flutter." GeeksforGeeks, <https://www.geeksforgeeks.org/getting-started-with-cross-platform-mobile-application-using-flutter/>. Accessed 10 Mar. 2025.
20. GitHub Definition. TechTarget, <https://www.techtarget.com/searchitoperations/definition/GitHub>. Accessed 15 Mar. 2025.
21. Hot Reload. Flutter Documentation, <https://docs.flutter.dev/tools/hot-reload>. Accessed 7 Mar. 2025.
22. **Huawei**. HarmonyOS. [Online]. Available: <https://www.harmonyos.com/en/>. Accessed: 10-Mar-2025
23. IBM. The History of Artificial Intelligence. IBM, <https://www.ibm.com/think/topics/history-of-artificial-intelligence>. Accessed 13 Mar. 2025.
24. **IEEE**. *IEEE 802.15.1 Standard*. [Online]. Available: <https://standards.ieee.org/ieee/802.15.1/1180/>. Accessed: 10-Mar-2025
25. **International Electrotechnical Commission (IEC)**. *IEC Publication 62790*. [Online]. Available: <https://webstore.iec.ch/en/publication/62790>. Accessed: 10-Mar-2025
26. Introduction to MySQL. Hostinger, <https://www.hostinger.com/tutorials/what-is-mysql>. Accessed 8 Mar. 2025.
27. Instructables. "How to Build Arduino Weighing Scales." Instructables, www.instructables.com/How-to-Build-Arduino-Weighing-Scales/. Accessed 19 Jan. 2025.
28. **International Organization for Standardization (ISO)**. *ISO/IEC 27001 – Information Security Management*. [Online]. Available: <https://www.iso.org/standard/27001>. Accessed: 10 Mar. 2025
29. **Klarus**. *KLARUS 16340 700mAh Datasheet*. [Online]. Available: <https://www.shorepowerinc.com/media/wysiwyg/files/klarus/datasheet/KLARUS-16340-700-DataSheet.pdf>. Accessed: 2 Mar. 2025.
30. MicroPython. MicroPython, <https://micropython.org/>. Accessed 11 Mar. 2025.
31. **Newhaven Display**. *NHD-4.3-480272FT-CSXP-T Specification*. [Online]. Available: <https://newhavendisplay.com/content/specs/NHD-4.3-480272FT-CSXP-T.pdf>. Accessed: 10-Mar-2025
32. **PCBONLINE**. *Top 6 Popular Microcontrollers*. [Online]. Available: <https://www.pcbonline.com/blog/top-6-popular-microcontrollers.html>. Accessed: 12-Mar-2025
33. React Native Definition. Netguru, <https://www.netguru.com/glossary/react-native>. Accessed 10 Mar. 2025.
34. Real-time Database Documentation. Firebase, <https://firebase.google.com/docs/database>. Accessed 7 Mar. 2025.
35. **Riverdi**. *OLED vs. LCD: A Comprehensive Comparison*. [Online]. Available: <https://riverdi.com/blog/oled-vs-lcd-a-comprehensive-comparison>. Accessed: 4-Mar-2025.
36. ScienceDirect. "Cross-platform Mobile Application Development: A Review of Features and Challenges." ScienceDirect, <https://www.sciencedirect.com/science/article/abs/pii/S0950584920301439>. Accessed 9 Mar. 2025.

37. **SEACOMP.** *DT043CTFT-IPS Display Specification*. [Online]. Available: <https://www.seacomp.com/sites/default/files/datasheets/DT043CTFT-IPS-Displaytech-Spec.pdf>. Accessed: 10-Mar-2025
38. **SEACOMP.** *5-Inch TFT LCD DT050BTFT Specification*. [Online]. Available: <https://www.seacomp.com/product/5-inch-tft-lcd-dt050btft>. Accessed: 10-Mar-2025
39. SQLite About Page. SQLite, <https://www.sqlite.org/about.html>. Accessed 7 Mar. 2025.
40. SparkFun Electronics. SparkFun Load Cell - 5kg (SEN-13329). SparkFun Electronics, 2025, www.sparkfun.com/products/13329.
41. Swift. Swift.org, <https://www.swift.org/>. Accessed 10 Mar. 2025.
42. Texas Instruments. Weigh Scale Design Using Load Cell and ADS1232 (TIDUAC1A). Texas Instruments, 2015, www.ti.com/lit/ug/tiduac1a/tiduac1a.pdf.
43. **Tiny Circuits.** *ASR00050 18650 2500mAh Datasheet*. [Online]. Available: https://www.mouser.com/datasheet/2/855/ASR00050_18650_2500mAh-3078640.pdf. Accessed: 2-Mar-2025.
44. United Nations Environment Programme. "AI Has an Environmental Problem. Here's What the World Can Do About It." UNEP, <https://www.unep.org/news-and-stories/story/ai-has-environmental-problem-heres-what-world-can-do-about>. Accessed 13 Mar. 2025.
45. United States Food and Drug Administration. Food Packaging & Other Substances that Come into Contact with Food - Information for Consumers. FDA, <https://www.fda.gov/food/food-ingredients-packaging/food-packaging-other-substances-come-contact-food-information-consumers>. Accessed 13 Mar. 2025.
46. **US Micro Products.** *Introduction to OLED Technology & Its Pros/Cons*. [Online]. Available: <https://www.usmicroproducts.com/blog/introduction-oled-technology-its-proscons>. Accessed: 4 Mar. 2025.
47. **Uswitch.** *Mobile Operating Systems Guide*. [Online]. Available: <https://www.uswitch.com/mobiles/guides/mobile-operating-systems/>. Accessed: 10 Mar. 2025
48. W3Schools. "C Language Introduction." W3Schools, https://www.w3schools.com/c/c_intro.php. Accessed 10 Mar. 2025.
49. Zebra Technologies. "Mobile Printer Battery FAQ." Zebra Technologies, www.zebra.com/content/dam/zebra_dam/en/faq/mobile-printer-battery-faq-a4-en-us.pdf. Accessed 19 Jan. 2025.

Declaration: We hereby declare that we have not copied more than 7 pages from the Large Language Model (LLM). We have utilized LLM for drafting, outlining, comparing, summarizing, and proofreading purposes.

Appendix E: ChatGPT Prompts & Outcomes

E.1: “what would be the best lithium-ion batteries to use for the project”

ChatGPT said:

The best **lithium-ion battery** for your project depends on factors like power consumption, size, weight, and rechargeability. Since your project involves **LEDs as a guideline on a device**, I assume you need a **compact, rechargeable, and efficient power source**. Here are some good lithium-ion battery options:

Best Lithium-Ion Battery Options for Your Device:

1. 18650 Li-ion Battery (3.7V, ~2200mAh - 3500mAh)

✓ **Best for:** Mid-to-high power consumption devices that need good battery life.

🔋 **Capacity:** 2200mAh - 3500mAh per cell.

⚡ **Voltage:** 3.7V nominal (4.2V fully charged).

📏 **Size:** 18mm diameter, 65mm length (cylindrical).

🔄 **Rechargeable?** Yes, supports many charge cycles (~500+).

Why? Commonly used in flashlights, power banks, and small electronics. A **single** 18650 cell may be enough, or you can use multiple in **series or parallel** to match your voltage/current needs.

2. LiPo (Lithium Polymer) Battery Pack (3.7V - 7.4V, 500mAh - 5000mAh)

✓ **Best for:** Compact, lightweight, and custom-shaped designs.

🔋 **Capacity:** 500mAh - 5000mAh (varies based on size).

⚡ **Voltage:** Usually 3.7V per cell (or 7.4V for a 2S pack).

📏 **Size:** Slim and flat (varies by capacity).

🔄 **Rechargeable?** Yes, but more delicate than cylindrical Li-ion cells.

Why? If space is a concern, **LiPo batteries** are flexible and can be shaped to fit into a custom enclosure. However, they require **careful charging** to avoid swelling.

3. 14500 Li-ion Battery (3.7V, ~800mAh - 1200mAh)

✓ **Best for:** Compact, AA-sized battery-powered devices.

🔋 **Capacity:** 800mAh - 1200mAh.

⚡ **Voltage:** 3.7V per cell.

📏 **Size:** Same as AA batteries (14mm diameter, 50mm length).

🔄 **Rechargeable?** Yes.

Why? If your device is small and currently designed for **AA batteries**, this is a great drop-in **rechargeable** replacement (but be mindful of the voltage difference).

4. Lithium-ion Battery Packs (7.4V - 12V, 2000mAh - 10,000mAh)

✓ **Best for:** Higher power requirements, longer run time.

🔋 **Capacity:** 2000mAh - 10,000mAh.

⚡ **Voltage:** 7.4V (2-cell), 11.1V (3-cell), 12V (multi-cell packs).

📏 **Size:** Varies, but generally larger.

🔄 **Rechargeable?** Yes, great for continuous use.

Why? If your device needs more **power or longer battery life**, a **multi-cell Li-ion pack** could work. These are commonly used in **robots, LED strips, and portable tools**.

E.2: “What are some standards and constraints that should be considered for this product?”

ChatGPT said:



STANDARDS TO CONSIDER

1. Health and Safety Standards (for food-contact materials)

FDA Food Contact Regulations (21 CFR Part 177)

Any part of your device that comes in direct contact with breastmilk (or storage bags) must be made from **FDA-compliant food-grade materials**.

→ Materials should be **BPA-free, non-toxic, and easy to sanitize**.

ISO 22000 (Food Safety Management)

If you ever plan to expand this device into commercial/clinical use, aligning with **ISO food safety** practices is a big plus. Covers hygiene, contamination prevention, and safe handling.

2. Labeling and Traceability Standards

Human Milk Banking Association of North America (HMBANA) Guidelines

If your device supports milk donation, you should look at HMBANA standards for **labeling breastmilk**:

Date and time of expression.

Donor information.

Milk treatment (if pasteurized).

Storage conditions.

Labels should be **clear, legible, and durable** in freezing/thawing conditions.

GS1 Standards (Optional for future-proofing)

If you envision this system scaling to clinics or milk banks, **GS1 barcodes or QR codes** are used for healthcare traceability. They can hold unique donor or batch data for better **inventory management**.

3. Electrical and Electronic Safety Standards

UL/IEC 62368-1 (Audio/Video, IT and Communication Technology Equipment Safety)

If the device is **battery-operated** and has **electronic components**, you may need to ensure compliance with general electronic safety standards.

→ Focus on **safe voltage levels, overheating, and battery safety**.

FCC Part 15

If your device uses **Bluetooth or Wi-Fi**, it must comply with **FCC standards for electromagnetic interference (EMI)** to ensure it doesn't disrupt other devices.

4. Battery and Charging Standards

IEC 62133-2 (Rechargeable Battery Safety)

For **portable, battery-powered devices**, especially lithium-ion batteries, this standard addresses **thermal runaway, short circuit, overcharge protection**, etc.

5. Software Standards

ISO/IEC 25010 (Software Quality)

Ensures your app/software design focuses on **usability, reliability, and security**.

HIPAA (Health Insurance Portability and Accountability Act)

If you store **personal health information** (baby feeding data, donor details), you should **encrypt** user data and comply with **privacy and security** requirements.

✓ DESIGN CONSTRAINTS TO CONSIDER

1. Portability

The device should be **lightweight, compact, and easy to transport**, since moms may need it at home, work, or while traveling.

Battery-powered operation is key, limiting **power consumption** and requiring **efficient energy use**.

2. Ease of Use

One-handed operation (moms may be holding a baby).

Simple user interface (large buttons or touch areas, intuitive design).

Fast and **hassle-free setup**—no complicated pairing or calibration each use.

3. Accuracy and Precision

Scale must have **high sensitivity** to measure small volumes (typically 1-10 oz), ideally accurate to **±1 gram or better**.

Measurement needs to **account for tare weight** (the bag/container).

4. Environmental Constraints

Must be **resistant to spills, moisture, and condensation** (since milk can leak).

Operate at **common indoor temperatures**, and components should tolerate **freezer or refrigerated** conditions if needed.

Label printer should produce **smudge-proof, water-resistant** labels that adhere even in **cold or damp conditions**.

5. Cost Constraints

Affordable for end-users (mothers, milk banks, healthcare providers).

Use **off-the-shelf components** when possible to minimize **development and manufacturing costs**.

6. Maintenance and Durability

Easy to **clean and sanitize**—no complicated disassembly.

Components (scale, printer) need to be **durable** for repeated use.

Labels and adhesives should **withstand freezing and thawing**.

7. Power/Battery Life

Must offer **long battery life** between charges.

Include **low battery indicators** (LEDs, app notifications).

Quick and **safe recharging** (USB-C or other standard).

8. Connectivity

Bluetooth/Wi-Fi for app communication, but needs to be **secure and stable**.

Optional **offline functionality** if internet isn't available.

✓ How You Can Present It in Your Report

Example Outline for Chapter 4:

Introduction

Purpose of standards and addressing constraints.

Applicable Standards

Health & Safety (FDA, HMBANA, ISO)

Labeling & Traceability (HMBANA, GS1)

Electronic Safety (UL, FCC)

Software (HIPAA, ISO/IEC)

Battery Safety (IEC)

Design Constraints

Portability and Power

Usability (One-handed use, UI simplicity)
 Accuracy and Precision
 Environmental (Temperature, moisture)
 Cost considerations
 Maintenance & Durability
 Connectivity and Data Security

E.3: “Help me find reference schematics

To assist you in developing schematics for your breastmilk tracking and labeling device, here are some reference materials for the key hardware components:

1. Load Cell with HX711 Amplifier

To measure the weight of breastmilk accurately, you'll utilize a load cell in conjunction with the HX711 amplifier. [Random Nerd Tutorials+1Just Do Electronics+1](#)

- **SparkFun's HX711 Load Cell Amplifier Hookup Guide:** This comprehensive tutorial provides detailed instructions on connecting a load cell to the HX711 amplifier and interfacing it with a microcontroller. It includes wiring diagrams and explanations of the color coding for load cell wires. [SparkFun Learn](#)
- **Arduino with Load Cell and HX711 Amplifier Tutorial:** This guide walks through creating a digital scale using an Arduino, load cell, and HX711 amplifier. It covers wiring, calibration, and code examples. [Random Nerd Tutorials](#)

2. Microcontroller (ESP32) Integration

The ESP32 microcontroller will serve as the central processing unit, handling data from the load cell and controlling other peripherals. [Circuit Designer Tutorials+1GitHub+1](#)

- **ESP32 Hardware Design Guidelines:** Espressif's official document offers detailed insights into designing circuits with the ESP32, including recommended schematics for power supply, crystal oscillators, and other essential components. [Espressif Documentation](#)
- **Custom ESP32 Board Design Tutorial:** This step-by-step guide demonstrates how to design custom ESP32-based boards, which can be instrumental in understanding the integration of various components. [Instructables](#)

3. Thermal Printer Integration

For printing labels, integrating a thermal printer with the ESP32 is essential.

- **IoT Thermal Printer Using ESP32:** This video tutorial showcases the integration of a thermal printer with the ESP32, providing practical insights and common troubleshooting tips. [Circuit Designer Tutorials+2YouTube+2Reddit+2](#)
- **ESP32-Controlled Thermal Printer Project:** This project details how to control a thermal printer remotely using an ESP32, including insights into interfacing and communication protocols. [Robotics Hub](#)

4. Power Management

Efficient power management ensures the device operates reliably, especially when battery-powered.

- **ESP32 Hardware Design Guidelines:** Beyond general design, this document provides specific recommendations for power supply designs suitable for the ESP32. [Espressif Documentation](#)

By studying these resources, you can develop a comprehensive understanding of the schematics required for each component of your device. Ensure to adapt the reference designs to fit the specific requirements and constraints of your breastmilk tracking and labeling system.