PUMPED

Final Presentation Presented by Group 14

Sponsored by Victoria Rodriguez & Matthew Aberman

OUR TEAM

Simran Aulakh
Electrical Engineer

Johan Gales
Electrical Engineer

Alexandra Atuncar
Computer Engineer

Mya Camacho-Hill
Computer Engineer

MOTIVATION & BACKGROUND

- Our sponsors, Victoria and Matthew, are full-time working parents who are committed to providing the best care for their baby. They want to ensure their child continues to receive the nutritional benefits of breastmilk, even when mom is at work or when feeding is handled by dad or another caregiver.
- Breastmilk is often pumped for future use, requiring accurate labeling of the volume, date, and time.
- Sleep-deprived mothers may struggle to consistently track and record this information during or after pumping.
- Current storage bags rely on handwritten labels, which can be inconvenient—especially when caring for a newborn.
- Our goal is to simplify this process by automating the tracking and labeling of breastmilk, reducing mental load and improving accuracy.

GOALS

Basic Goals

- Help mothers manage breastmilk by designing a device that measures and prints a label for pumped breastmilk
- Create a user-friendly interface for mothers to access information

Advanced Goals

- Optimize design for portability, offer customizable labeling
- Help mothers by monitor milk inventory
- Design a physical guide system to minimize milk spillage while providing a visual feedback on system status

Stretch Goals

- Provide insights to help mothers make informed dietary decisions
- Implement multi-user access with varying levels of permission

OBJECTIVES

Accurate Measurement & Labeling:

- Device will measure and log milk volume using a built-in sensor
- Labels will include date, time, weight, and optional dietary/allergen restrictions, adhering to milk bank/hospital requirements
- Mobile app will allow manual editing of details

Portability:

- Device will be compact, lightweight, and battery-operated for on-the-go use
- LEDs will provide low battery alerts

Smart Inventory Management

- Accompanying software will manage inventory, track expiration dates, and provide various notifications
- User-friendly app will analyze patterns (feeding frequency, baby's age, stored amount) to anticipate future milk needs.

SPECIFICATIONS

Parameter	Description	Value
Weighing Accuracy	**Accuracy of weight sensor's detection**	**±0.2 fl oz**
Maximum Weight Capacity	Maximum amount of weight the scale can hold	1-11 lbs.
Label Dimensions	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 database**	**< 5 seconds**
Printing Time	**Time it takes to print one label after button is pressed**	**< 8 seconds**
Weight	Weight of the product	1-5 lbs.
Storage Capacity	Total logs of milk bags that can be stored in data inventory	1200 entries
Battery Life	Device's usage time before it has to be charged	> 9 Hours
Cost	Total cost of final product	$874.28
Power Consumption	Total power consumed during system's usage	30.3375 W

HOUSE OF QUALITY

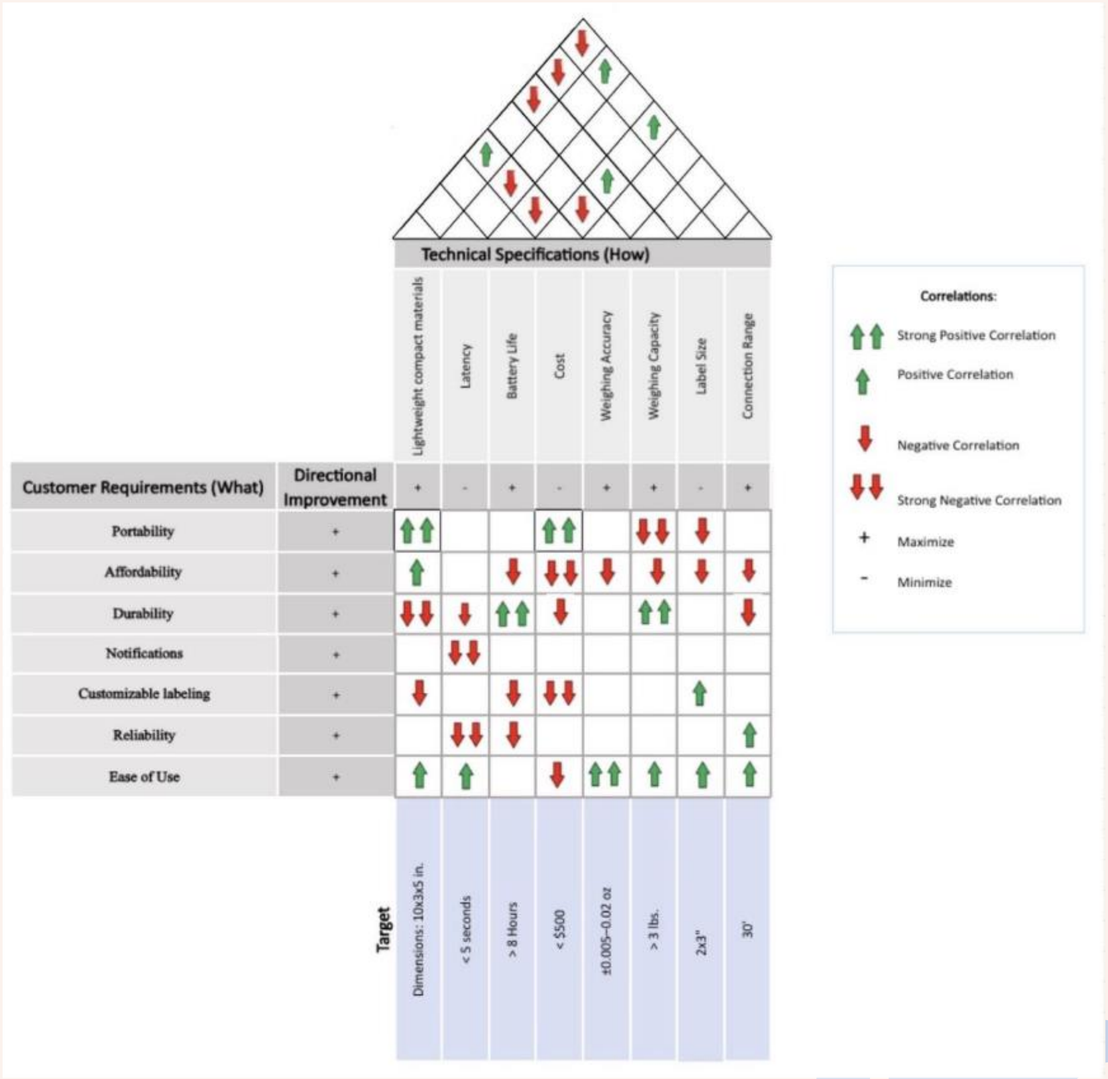
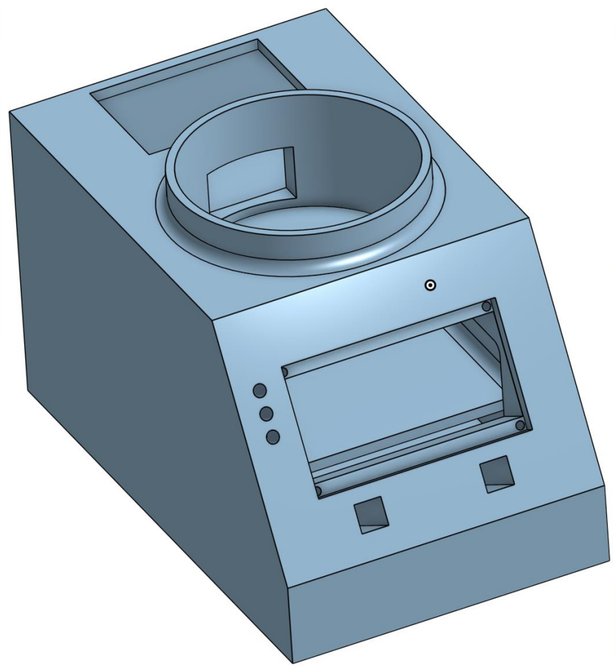
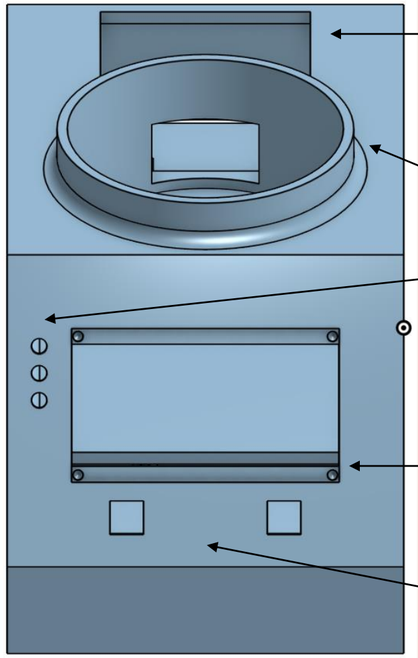


IMAGE & VISUAL OF FINAL PRODUCT





Printer

Scale

LED’s

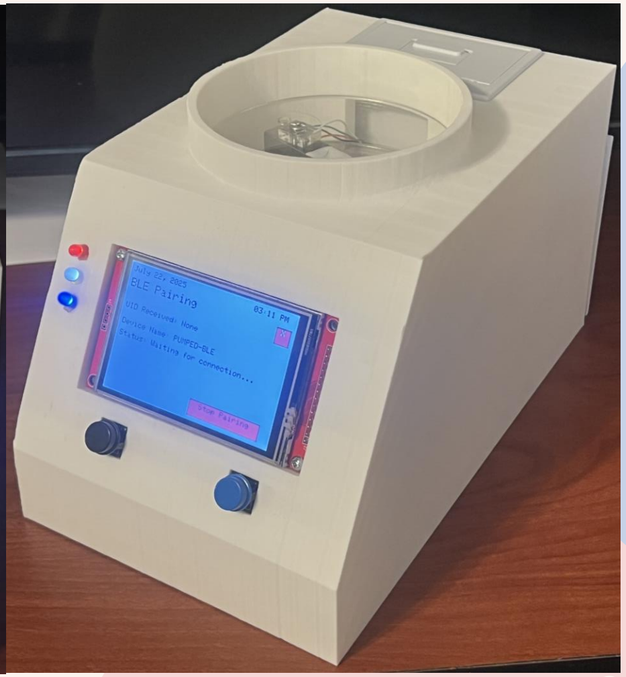
Red: Power

White: WiFi

Blue: Bluetooth

Touchscreen

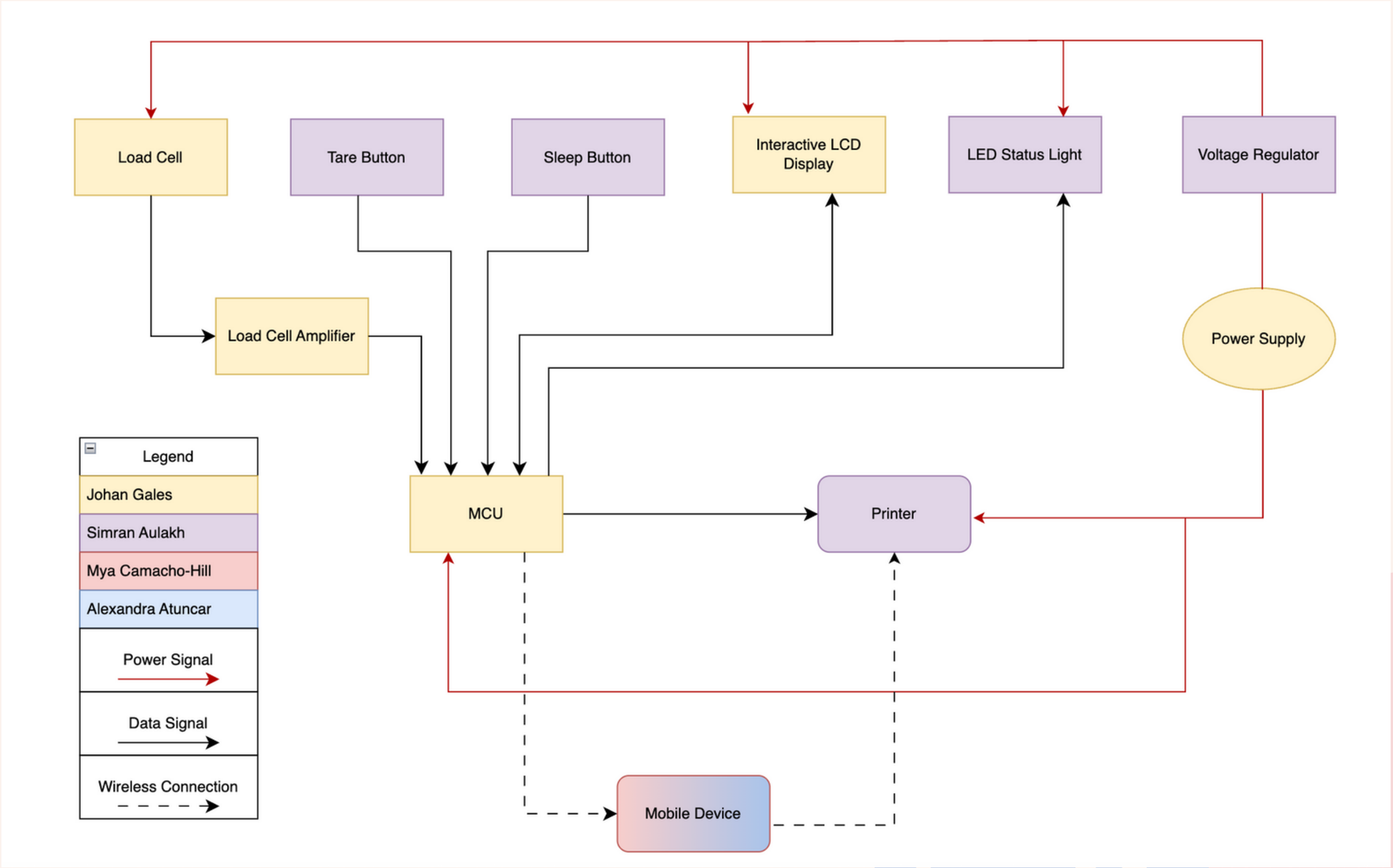
Buttons
(Left: Tare
Right: Sleep)



OnShape CAD

Final Product

HARDWARE BLOCK DIAGRAM



HARDWARE SELECTION: MCU

Microcontroller	Architecture	Operating Voltage	RAM	I/O Interfaces	Connectivity	Power Consumption
MSP430	16-bit	1.8-3.6V	2-8 KB	UART, SPI, I2C, ADC	None	Ultra-low power
Arduino Uno R3	8-bit	5V	2 KB	UART, SPI, I2C, ADC	None	Moderate
ESP32 - WROOM32	**32-bit**	**2.2-3.6V**	**520 KB**	**UART, SPI, I2C, ADC, DAC, PWM**	**Wi-Fi, Bluetooth**	**Higher Power Consumption**
Raspberry Pi Pico	32-bit	1.8-3.3V	264 KB	UART, SPI, I2C, ADC, PWM	None	Low Power
STM32F411	32-bit	1.7-3.6V	128 KB	UART, SPI, I2C, ADC, DAC, PWM	None	Low Power

- Built-in Bluetooth & Wi-Fi features are crucial to our project
- Operates within a voltage range that works with our other selections

HARDWARE SELECTION: LOAD CELL

- The SparkFun SEN-13329 is compact & lightweight
- Has high accuracy which is needed for breastmilk weight measurement
- Good for small scale usage
- Easily available and good for prototyping

Model	Load Cell Type	Max Capacity	Material	Output Sensitivity	Excitation Voltage
SparkFun SEN-13329	**Single-Point**	**10 KG**	**Aluminum**	**0.9-1.1mV/V**	**5V**
CZL635	Single-Point	10 KG	Aluminum	0.85 - 1.15 mV/V	5V-10V
Bosche H30A	Bar-Type	60 KG	Stainless Steel	2.0 mV/V	5V-12V
Vishay Tedea – Huntleigh 1042	Bar-Type	50 KG – 200 KG	Stainless Steel	2.0mV/V	5V-12V
Vishay Tedea – Huntleigh 614	S-Type	100 KG – 500 KG	Nickel-Plated Alloy Steel	3.0mV/V	5V-15V

HARDWARE SELECTION: LOAD CELL AMPLIFIER

- The specific load cell amplifier we have chosen is the SparkFun SEN-13879
- It is directly compatible with the SparkFun SEN-13329 load cell
- Supports 3.3V, which works with our ESP32
- Has community support which provides example schematics and designs

Amplifier	Type	Power Supply	Interface
ADS1232	Digital ADC	5V	SPI
NAU7802	Digital ADC	5V	SPI
HX711	**Digital ADC**	**3.3V / 5V**	**I2C**
NAU7802	Digital ADC	3.3V / 5V	I2C
INA125	Digital ADC	Analog Instrumentation Amplifier	2.7V - 15V

HARDWARE SELECTION: PRINTER

- TTL Serial Interface
- Supports label printing
- Has a good trade-off between cost and functionality

Printer	Print Speed	Power Type	Connectivity	Label Size Supported
Primuz Micro Thermal Printer	**60 mm per second**	**Wired power, no batter**	**TTL Serial**	**57 x 30 mm**
MUNBYN ITPP941	150 mm per second	Wired power, no battery	USB	Various sizes
Bixolon SPP-R310	80 mm per second	24-hr standby battery life	Bluetooth, WLAN, USB	3 inches
Zebra ZQ620/ZQ520	114.3 mm per second	Battery powered	Bluetooth, USB	3-4 inches
Brother RJ-2035B	101.6 mm per second	Long lasting battery	Bluetooth, NFC, USB	2 inches

HARDWARE SELECTION: DISPLAY SCREEN

- Using an LCD screen over an OLED screen due to lower cost & no burn-in risk
- Has SPI interface and 3.3V-5V range
- Fits within 4'-5' inch size

Screen	Resolution	Display Type	Interface	Supply Voltage	Touchscreen Options
ILI9488 TFT Display	**320 x 480**	**TFT LCD**	**Multiple including SPI**	**3.3V - 5V**	**Resistive**
DT050BTFT – 5" Color TFT LCD Display	800 x 480	IPS TFT LCD	SPI/RGB	3.0V - 3.6V	Resistive or Capacitive
NHD-4.3-480272FT-CSXP-T (4.3" TFT LCD Display)	480 x 272	IPS TFT LCD	SPI	3.0V - 3.6V	Resistive

POWER DISTRIBUTION

Component	Power Requirement	Power Usage
ESP32	3.0V - 3.6V, 80 mA	240 mW – 288 mW
Printer	12V – 24V, 2.5 A	30W - 60W
Load Cell Amplifier	2.7V - 5.5V, 1.5 mA	4.05 mW – 8.25 mW
Touchscreen	3.3V - 5V, 20 mA	66 mW – 100 mW

HARDWARE SELECTION: POWER SUPPLY

- The TalentCell's 12V output satisfies our printer's power requirement while simultaneaously
- Has built-in safety features, such as overcharge, overdischarge and short-circuit protection
- Uses wall adapter plugged into standard AC outlet

Brand Model	Input	Output	Capacity	Weight
TalentCell (YB1206000)	**12.6V**	**12V/5V**	**6000mAh**	**380g**
Rapthor	12.6V	12V/5V	6500mAh/13000 mAh	380g
Melasta	12.6V	12V/9V/5V	5200mAh	400g

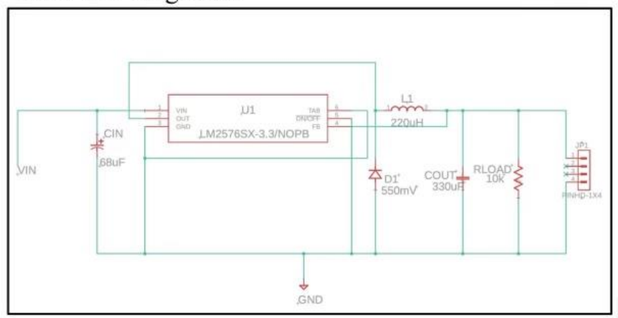
HARDWARE SELECTION: VOLTAGE REGULATOR

- The LM2576 has been highly recommended due to reliability
- 3.3V supports our ESP32 & touchscreen components
- 5V supports our load cell amplifier
- Can handle a wide range of voltage inputs as well
- Cost effective

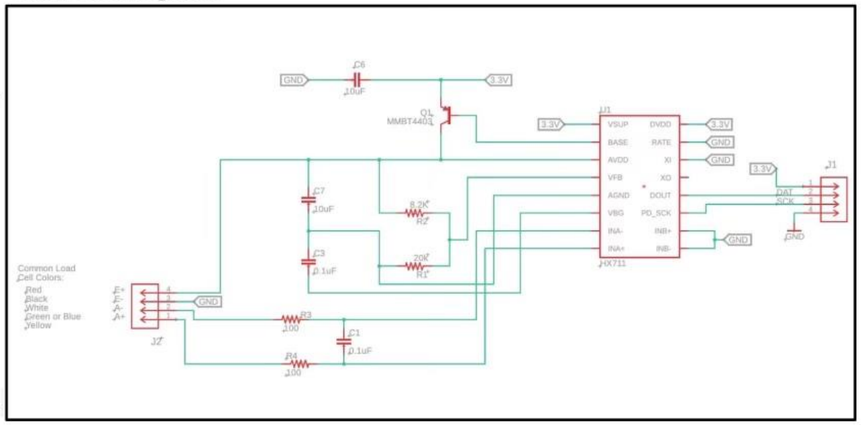
Voltage Regulator	Type	Output Voltage Range	Max Output Current	Efficiency
LM2576	**Buck**	**3.3V, 5V, 12V (Fixed) or 1.23V - 37V (Adjustable)**	**3A**	**~77-80%**
LM2679	Buck	3.3V, 5V, 12V (Fixed) or 1.23V - 37V (Adjustable)	5A	~85%
LM2596	Buck	3.3V, 5V, 12V (Fixed) or 1.23V - 37V (Adjustable)	3A	~80-85%
MP1584	Buck	0.8V - 20V (Adjustable)	3A	~90%
XL4015	Buck	1.25V - 36V (Adjustable)	5A	~96%

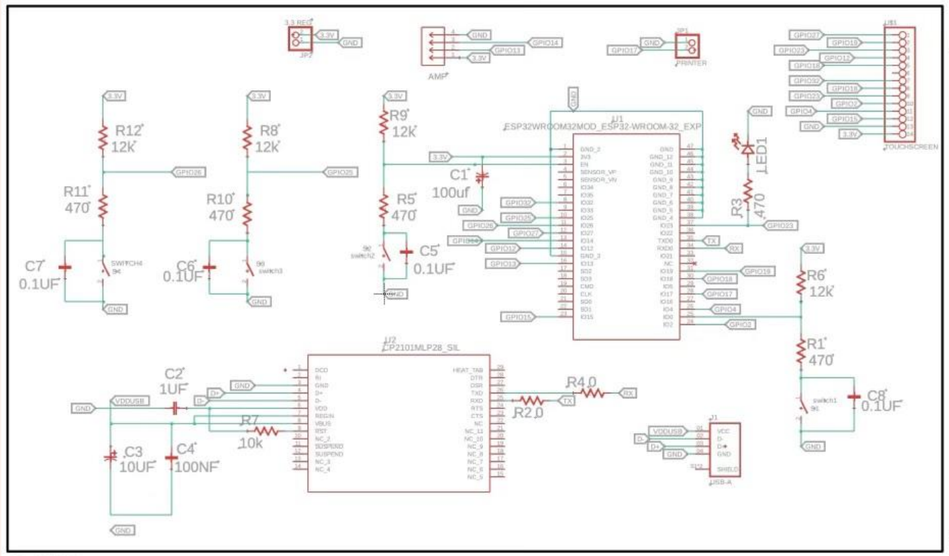
OVERALL SCHEMATIC

3.3V / 5V Regulator

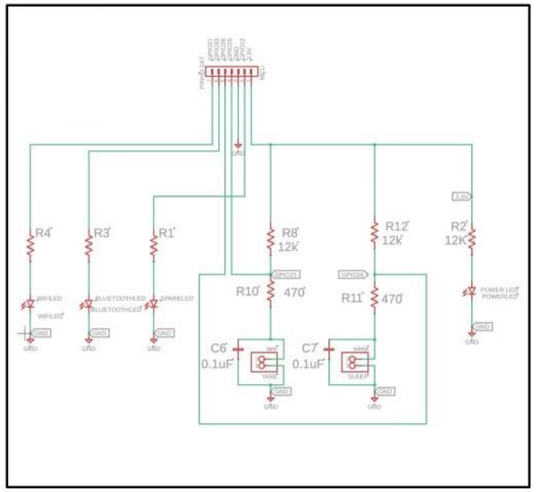


Load Cell Amplifier

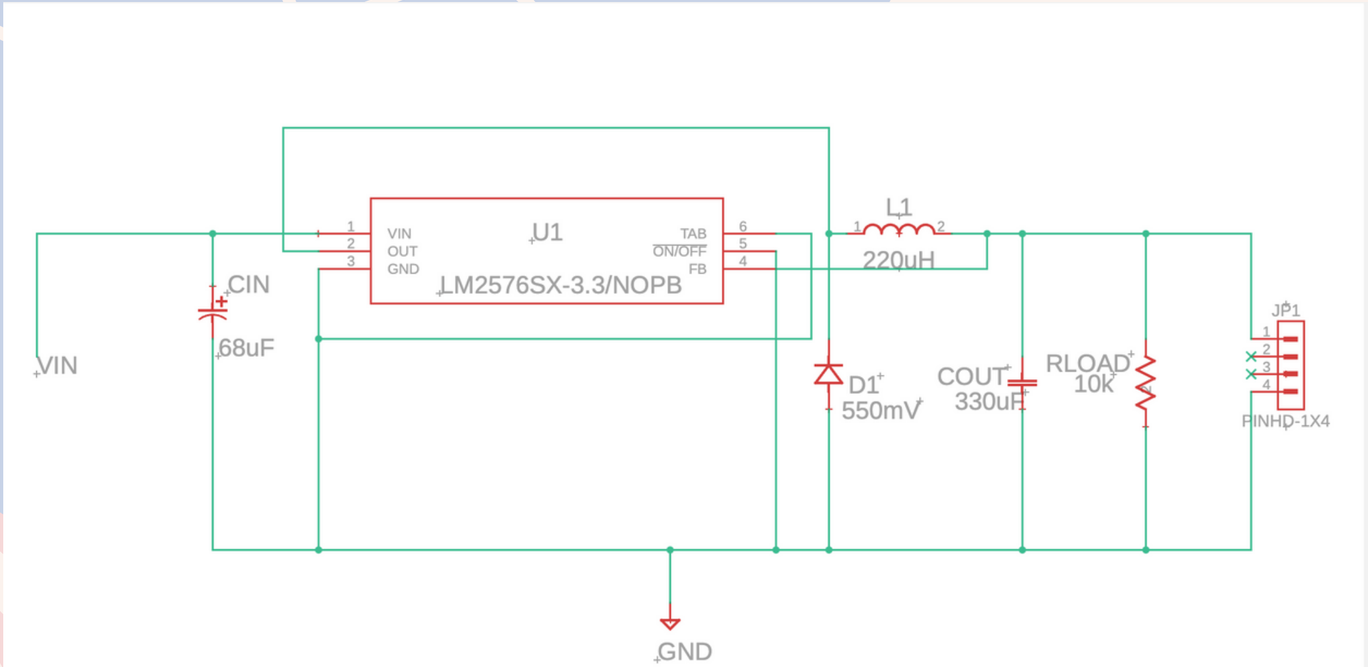


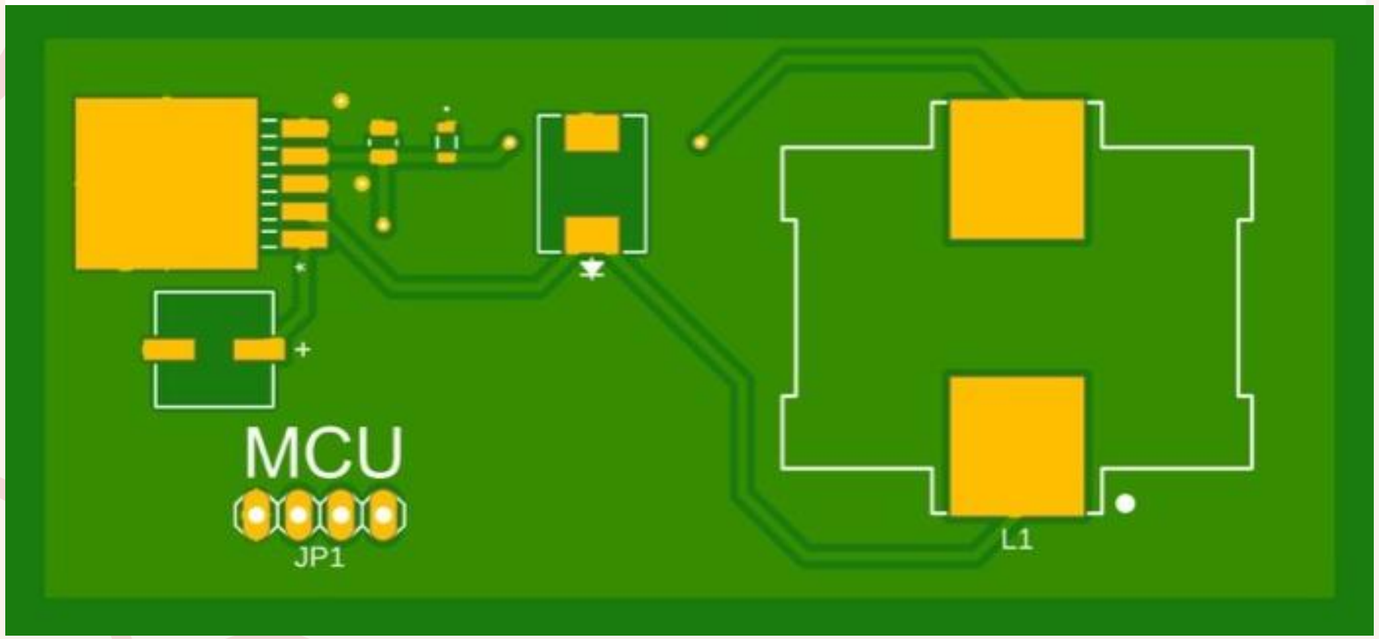


MCU



LED / Button Daughterboard



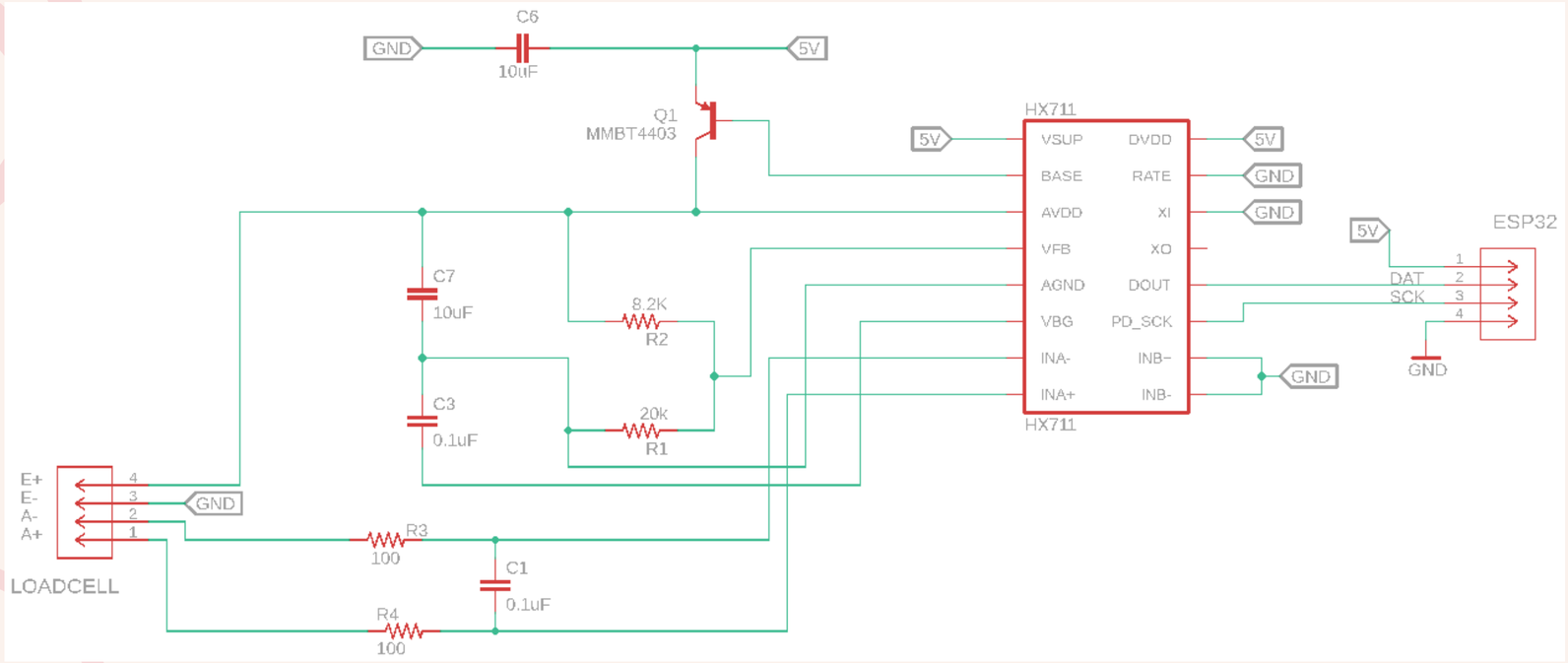


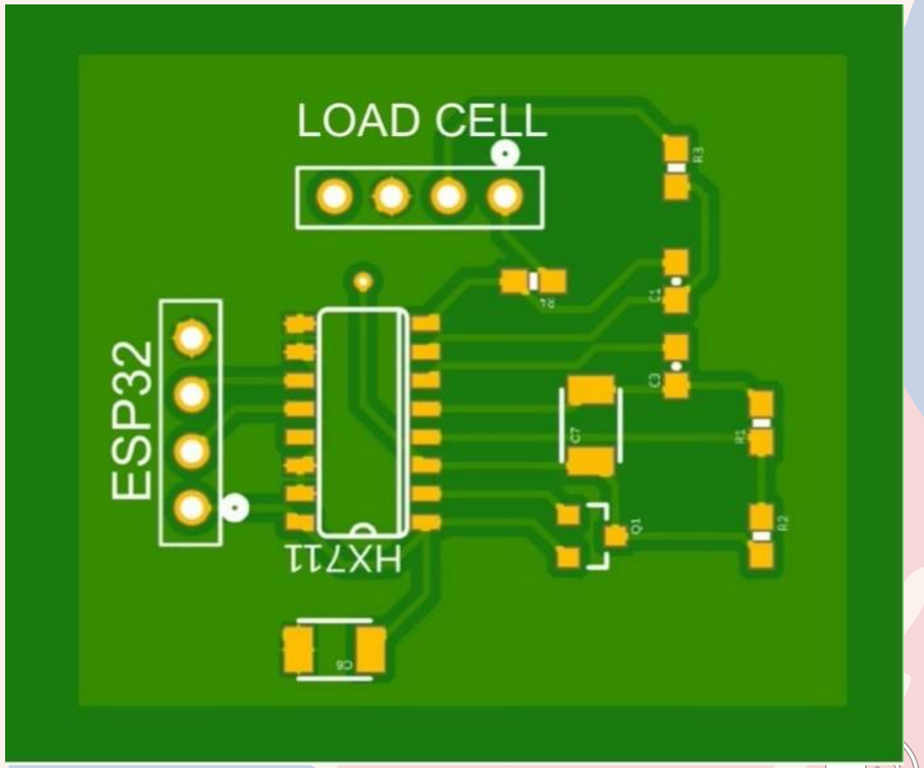
3.3V / 5V REGULATOR

- Provides 3.3V & 5V to our subsystems
- Adjustable regulator is used to modify voltage for our 5V regulator
- Schematic was generated using WeBench

LOAD CELL AMPLIFIER

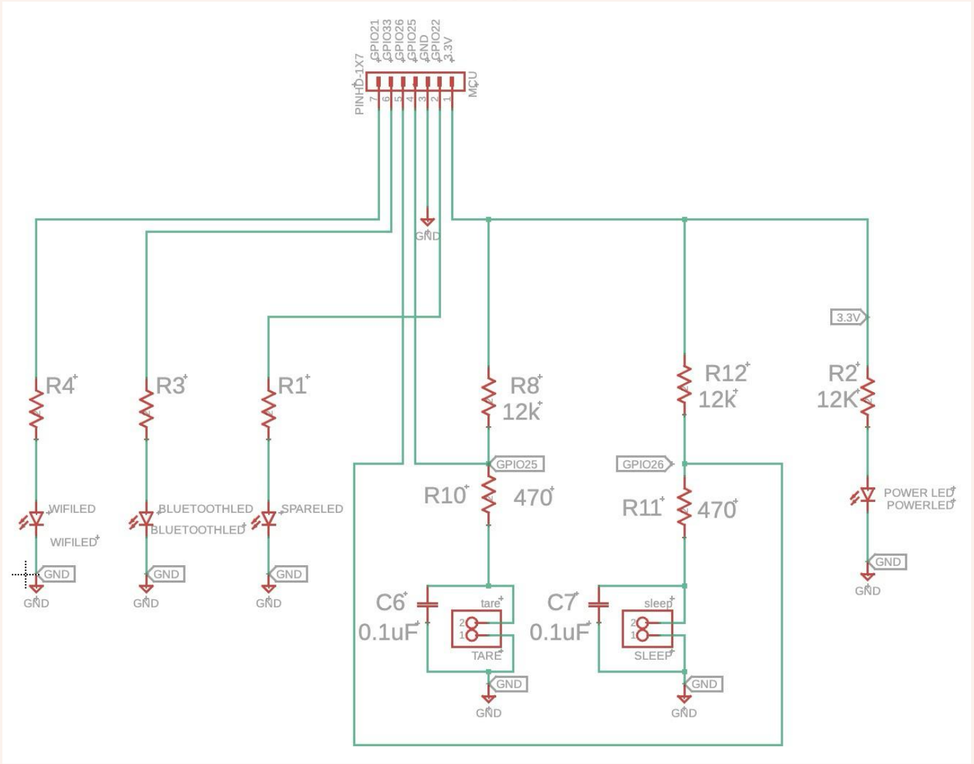
- Analog-to-Digital Converter that amplifies load cell's signal
- Left side is connected to load cell, right side is connected to MCU

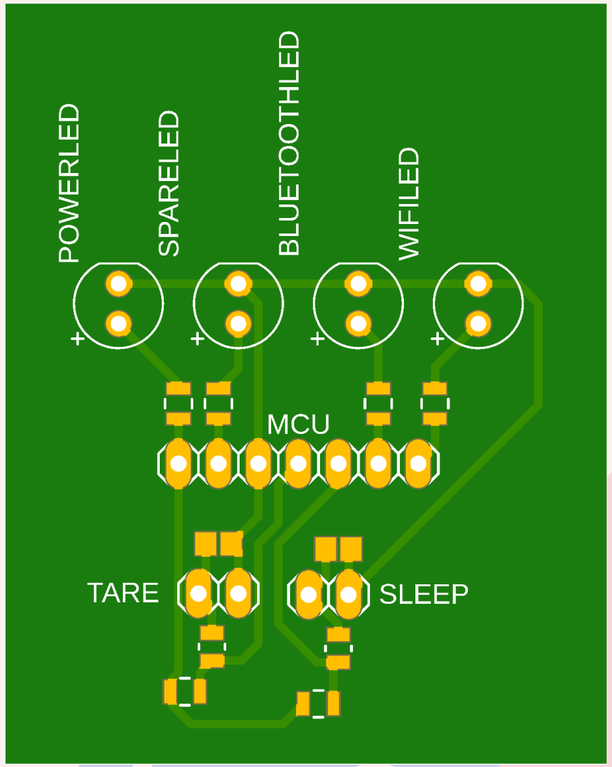




LEDS / BUTTONS

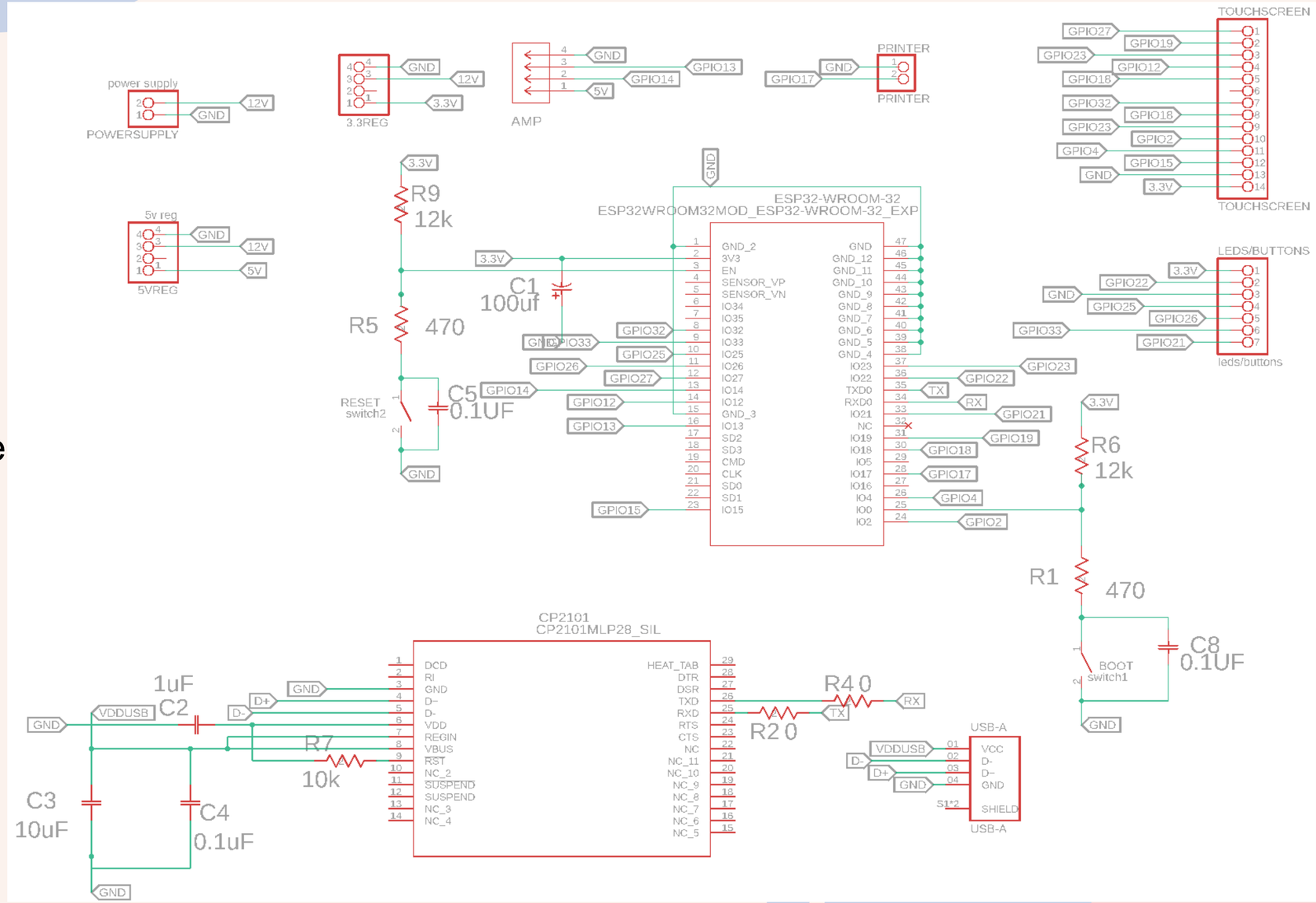
- Daughter board for connecting our external buttons / LEDs to the rest of the system
- Includes space for our sleep button, tare button, bluetooth LED, WiFi LED, and power LED
- All buttons / LEDs will be placed on front of our enclosure



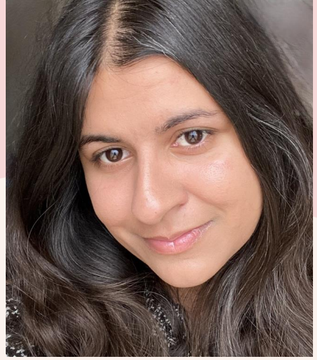


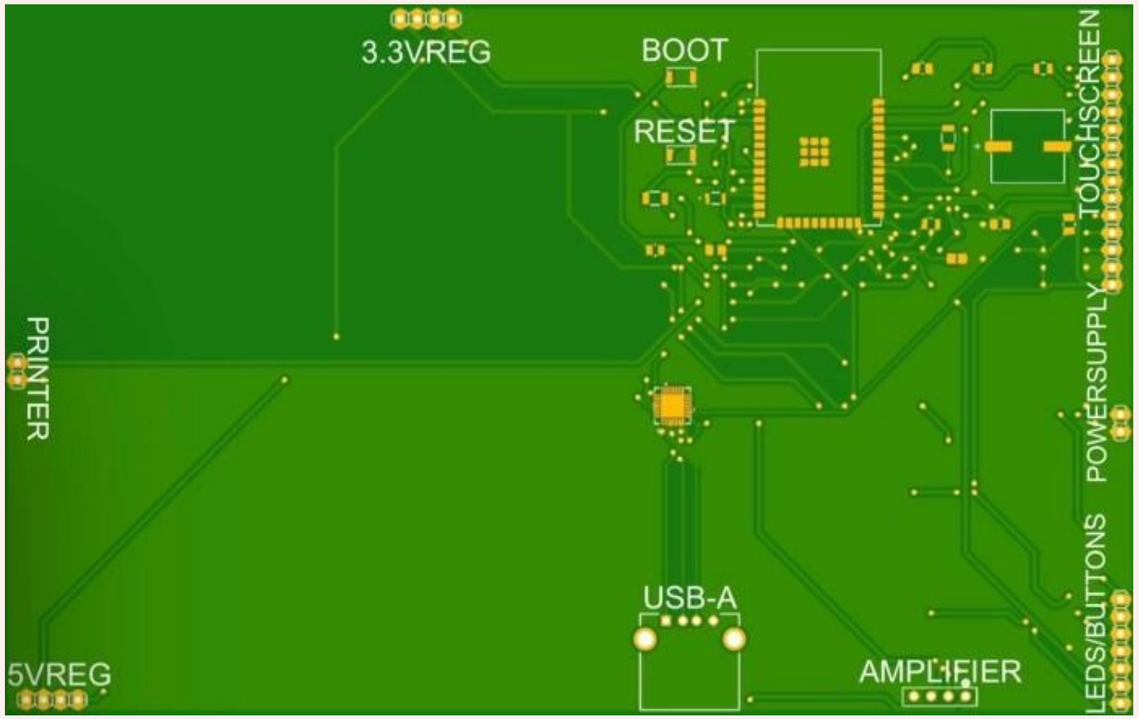
MCU DESIGN

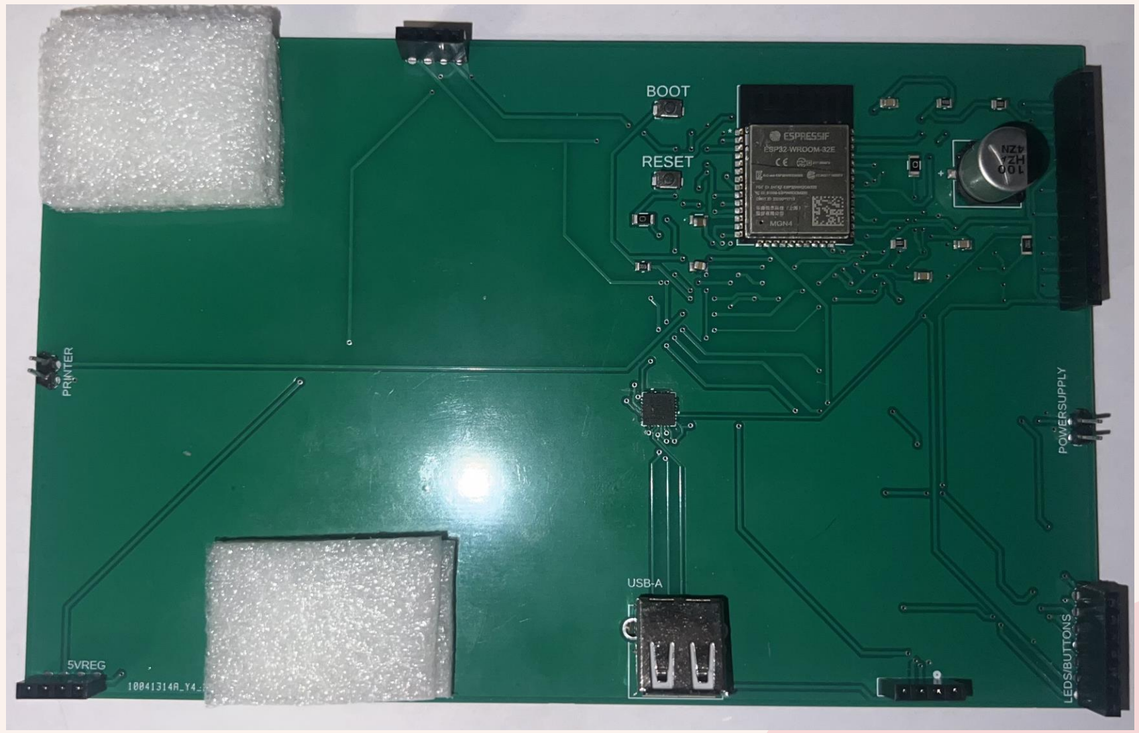
- System is centered around the ESP32
- CP2102N provides a USB-to-UART Bridge
- Several Peripherals are connected to the ESP32



MCU DESIGN




3.3VREG
BOOT
RESET
TOUCHSCREEN
POWERSUPPLY
LEDS/BUTTONS
PRINTER
USB-A
5VREG
AMPLIFIER


BOOT
RESET
PRINTER
POWERSUPPLY
USB-A
5VREG
LEDS/BUTTONS

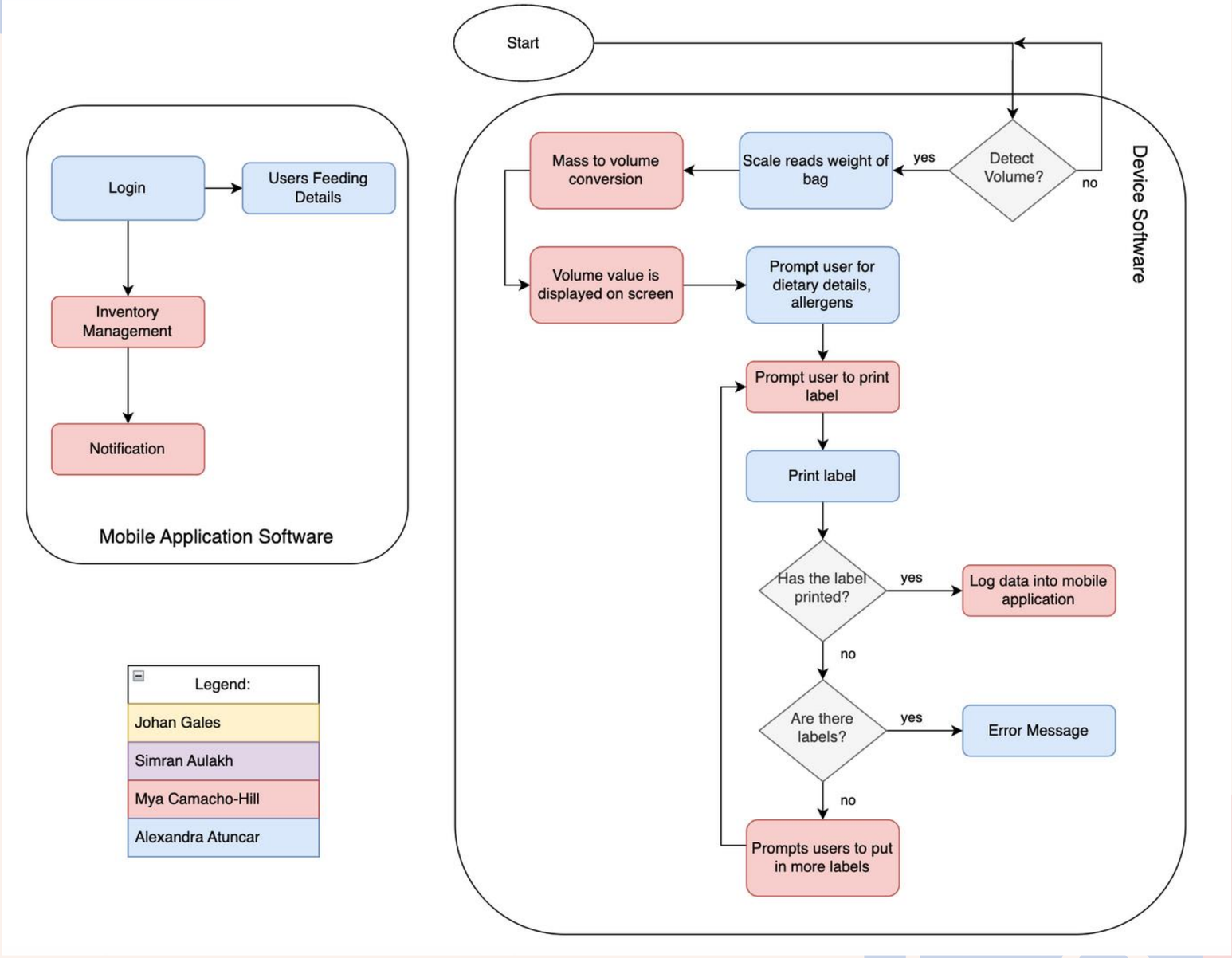
COMMUNICATION PROTOCOL SELECTION:

- In our project, we decided to utilize three microcontroller interfaces to ensure smooth and compatible communication between our components and the microcontroller.

- UART: Printer Communication
- SPI: Touchscreen Communication
- I2C: Load Cell Amplifier Communication

Interface	Speed	Wires Needed	Usage	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 (Universal Asynchronous Receiver Transmitter)	**Up to 1 Mbps**	**2 (TX, RX)**	**Debugging and communication with modules**	**GPS modules, Bluetooth modules, RFID readers**
SPI (Serial Peripheral Interface)	**Up to 50 Mbps**	**4 (MOSI, MISO, SCLK, CS)**	**Display communication and high-speed data transfer**	**TFT LCDs, SD cards, flash memory and high-speed ACDs**
I2C (Inter-Intergrated Circuit)	**Up to 3.4 Mbps**	**2 (SDA/SCL)**	**Sensors, memory and low-power devices**	**OLED displays, EEPROMs, IMUs and environmental sensors**

SOFTWARE FLOWCHART



SOFTWARE SELECTION: DEVELOPMENT ENVIRONMENT

Environment	Ease of Use	Language Support	Board Support	Debugging	Compiler	Firebase Integration
Arduino-IDE	Beginner Friendly (Simple UI)	C/C++ with Arduino API	Arduino, ESP32, Uno, Mega, etc.	Serial Monitor only	Moderate	Easy via Arduino Firebase Client
PlatformIO	**Intermediate (Advanced features)**	**C/C++, embedded Python (Including Micropython)**	**Extensive support: ESP32, STM32, AVR, RP2040, etc.**	**Integrated debugging (JTAG, GDB)**	**Moderate**	**Easy via Arduino Firebase Client**
ESP-IDF	Advanced (Experts use with steep learning curve)	Mainly C with C++, Rust, and Python support	ESP framework (ESP32, ESP32-S, ESP32-C series)	Full debugger support	Optimized	Manual implementation with REST/HTTP

We chose PlatformIO as our development environment because:

- Integrated debugging superior to the Arduino IDE
- Better library management and faster development cycle
- Easy to use with advanced features

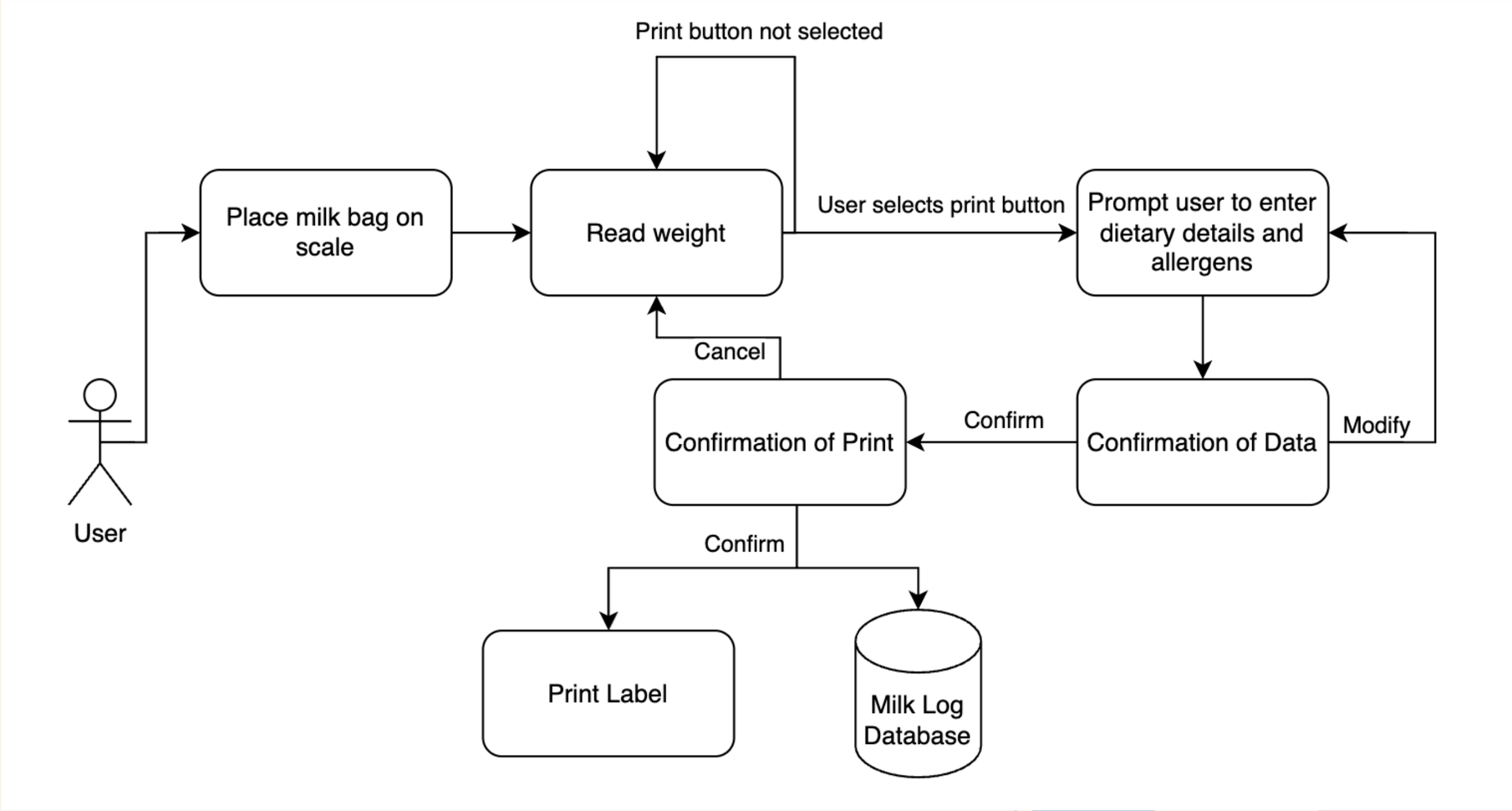
SOFTWARE SELECTION: DATABASE

We chose Firebase Realtime Database because:

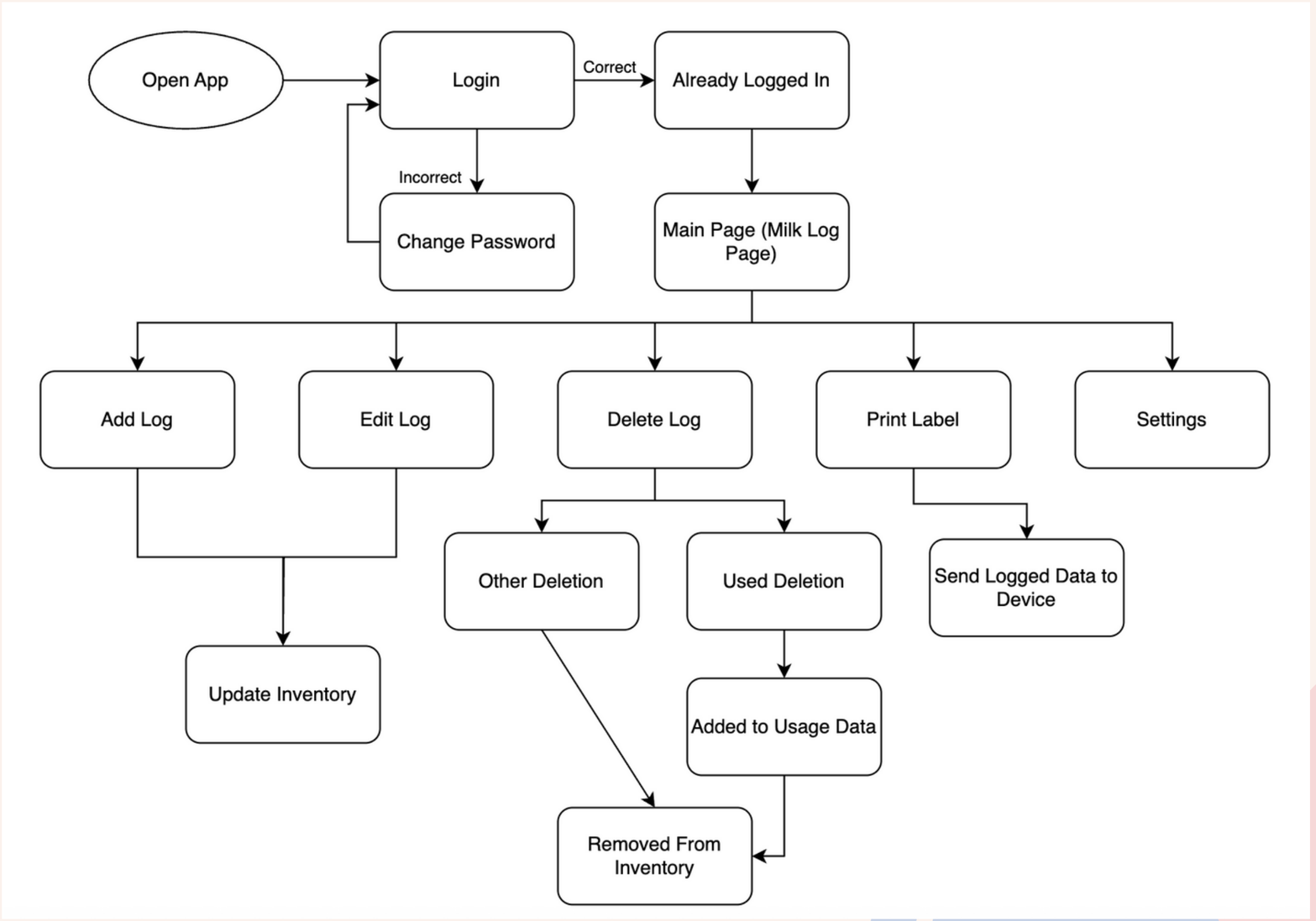
- User-friendly setup and cloud-hosting enable real-time data synchronization
- Native SDKs for Flutter
- It has built-in offline caching

Database	Ease of Setup	Type	Hosting	ESP Compatibility	Realtime Sync	Mobile Integration	Offline Support
SQLite	Easy	Embedded relational DB	Local (on-device/mobile)	Not ideal	No native support	Good (Android Room)	Excellent
MySQL Database	Moderate	Relational (SQL)	Requires external server	Needs HTTP/REST or intermediate app	No native support	Requires REST API or middleware	Not Ideal
Firebase Realtime Database	**Easy**	**NoSQL (key-value, tree-structured)**	**Cloud-hosted by Google**	**Direct support via REST/Arduino SDK**	**Built-in real-time sync**	**Native SDKs (Flutter, Android, iOS)**	**Excellent (Built-in offline cache)**
MongoDB	Moderate	NoSQL (document-based)	Cloud/self-hosted	Needs REST API or MQTT bridge	Requires custom setup	Requires third-party libraries	Moderate (Manual configuration needed)

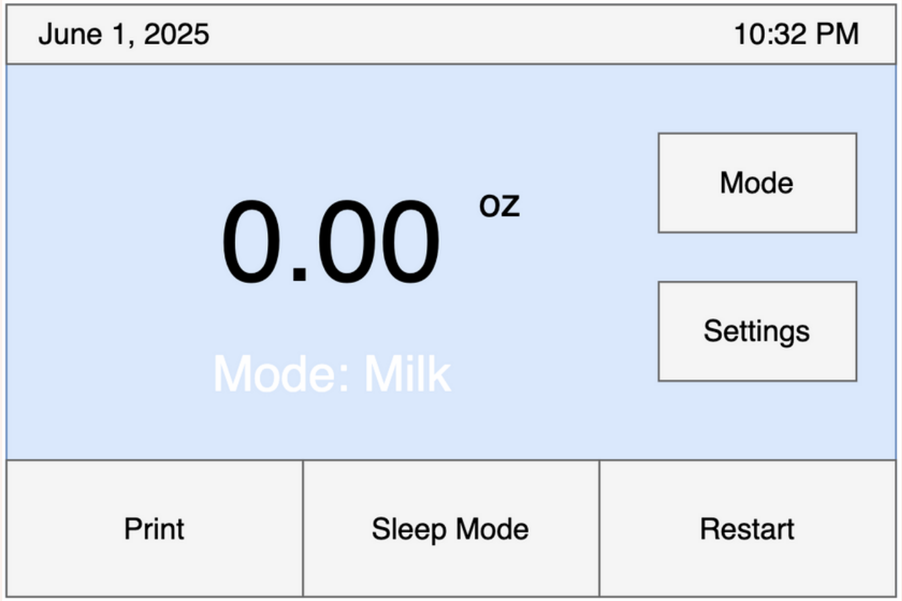
USE CASE DIAGRAM



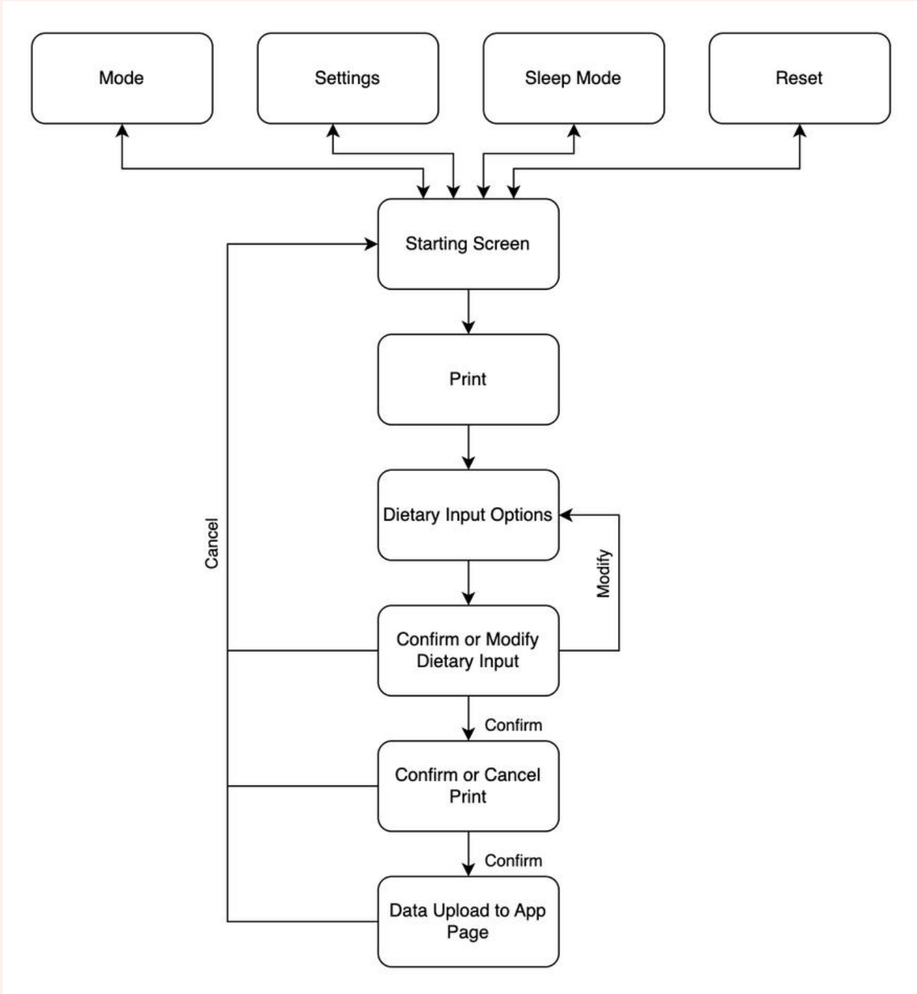
APPLICATION FLOWCHART



GRAPHICAL USER INTERFACE



Homepage Draft



Touchscreen Flowchart

TEST RESULTS

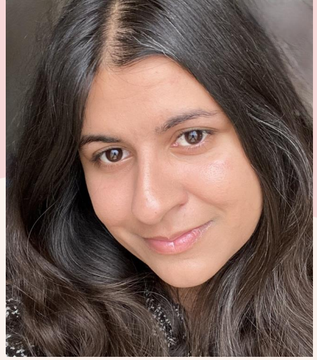
Test Number	Food Scale Weight (Fluid ounces)	Our Scale's Weight (Fluid ounces)	Print Response Time (Seconds)	Software Response Time (Seconds)
Test #1	2.96	2.97	3,28	1.23
Test #2	2.92	2.90	3.05	2.13
Test #3	3.87	3.87	3.41	3.35
Test #4	2.96	2.96	3.98	1.06
Test #5	2.92	2.93	3.89	3.01
Test #6	3.87	3.88	4.58	0.99
Test #7	2.96	2.97	4.14	2.54
Test #8	2.92	2.92	5.21	0.90
Test #9	3.87	3.86	5.28	1.25
Test #10	2.96	2.98	1.23	4.13

RESULTS ANALYSIS

Test Number	Weight Accuracy
Test #1	99.66%
Test #2	99.31%
Test #3	100%
Test #4	100%
Test #5	99.66%
Test #6	99.74%
Test #7	99.66%
Test #8	100%
Test #9	99.74%
Test #10	99.33%

Specification	Average Response Time / Accuracy
Printing Response Time	3.805 s
Software Response Time	2.059 s
Weight Accuracy	99.71%

HARDWARE TESTING



Challenges Faced:

- 5V Regulator Overload
- Hx711 Schematic and Reading Issues
- Peripherals connection to the main PCB board

Solutions:

- Rebuilt new 5V regulator board with new LM2576-ADJ chip
- Scratched ground into the board to match schematic and added direct wire from the data line to ESP32 pin.
- Added solder to jumpers for a more solid connection

Successes:

- Fully functioning PCB design
- Increased stability for peripherals connected to main PCB board
- Receiving correct voltage when testing the electronic load

SOFTWARE TESTING

Challenges Faced:

- Incorrect label printing size and feed line
- Significant issues with IOS notifications
- Connectivity to ESP32 via Bluetooth failing
- Struggled with developing the app due to lack of experience and Xcode limitations
- Connectivity to UCF Wifi failure

Solutions:

- Found labels better suited for print spacing to resolve alignment
- Used nRF scanner to diagnose weak Bluetooth signal and utilized user ID rather than passing user's credentials
- Researched and used a trial-and-error approach
- Hotspot connection to Wifi established

Successes:

- Fully functioning IOS application
- Provides visual feedback to users (ie. Notifications)
- Device able to successfully connect to Wifi on campus
- Realtime data sent from device and app to database

BUDGET

Product Name	Expected Cost	Actual Cost
Microcontroller	$10-$35	$40.26
Load-Cell	$6-$10	$20.33
Touchscreen Display	$20-$60	$16.99
Thermal Label Printer	$30-$120	$151.91
Thermal Label Paper	$10-$30	$13.67
Wires	$40	$41.58
PCBs + Tariffs	$80-$150	$306.75
LEDs	$5-$10	$15.68
Load-Cell Amplifier	$40-$50	$21.90
Rechargeable Battery	$9-$20	$41.68
3D Filament	$40	64.02
Prototyping	$100-$150	$139.51
Total Cost		$874.28

WORK DISTRIBUTION

Requirement	Primary Member	Secondary Member
Fusion360 Shell Design	Johan	Simran
Mobile Application Design Development	Mya	Alexandra
Device Software Design & Implementation	Alexandra	Mya
Printed Circuit Board Design	Simran	Johan
Weight Sensor Design & Implementation	Alexandra	Mya
Mobile Application Debugging	Alexandra	Mya
Battery Integration & Management	Johan	Simran
LED Control System	Simran	Mya
Display Integration	Mya	Alexandra
Wireless Communication	Mya	Alexandra
Thermal & Durability Testing	Simran	Johan
Testing & Debugging	All	

THANK YOU!