Shaken Baby Simulator

Software Development Plan & Requirements

March 1, 2022

Version 1.1

Presented To:

St. Luke's Network Simulation Center 3D Print & Innovation Lab

Submitted By:
Tyler Laudenslager, Boyd Lenich, and Andrew Sheidy

REVISION HISTORY

Date	Author	Distributed to	Version	Description
February 27th			1.0	Draft
March 1st	Tyler L.	N?A	1.1	Added more info to Use Cases, Non Functional Requirements, ,Research Sections, and Project Organization sections.
4/10/2022	Tyler L.	N?A	1.2	Added the prototype architecture pictures to the respective section.

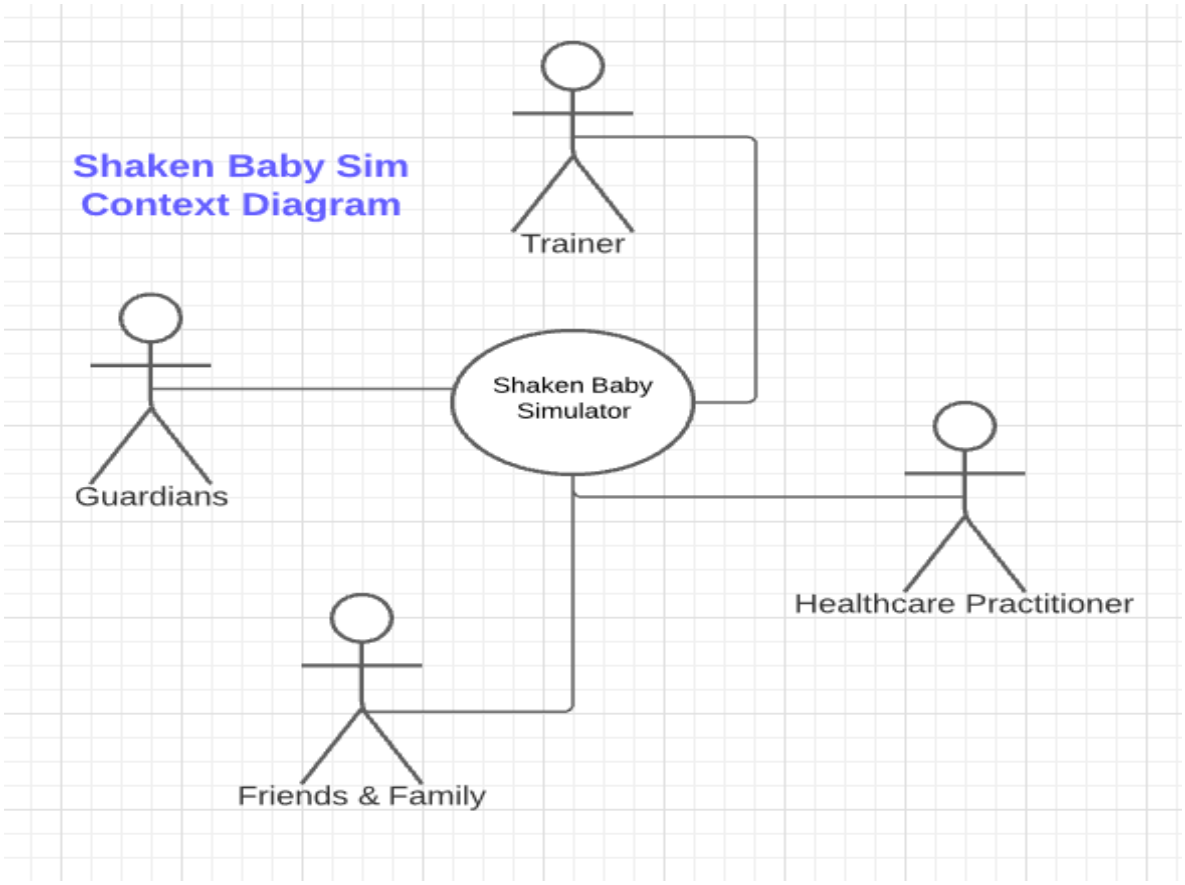
1. PRODUCT DESCRIPTION

We are solving an issue with child care, specifically for guardians who are new to interacting with an infant. The idea is to have a doll that responds to how it is treated to show signs of shaken baby syndrome, an issue that can happen if improper care is given while holding an infant. The doll will give info, to tell the handler that they are handling the baby correctly, or if they are giving the doll too much motion, and it will respond with information on a screen, as well as with qualitative changes, such as sound, and color.

2. PRODUCT REQUIREMENTS

Describe the users and user environment.

Context Diagram



Personas

Healthcare practitioner - Someone who would want to use to learn about child handling in a healthcare sense rather than a guardian sense

Family and Friends - Connected to the guardian, but does not need as in depth coaching on child handling as they would not have as close proximity

Guardians - Wants an in depth look on how to handle a child safely

Educator / Trainer - Someone who is using the doll as a teaching tool for others

User Stories

- Healthcare Practitioner
    - As a nurse, I want accurate data representation so I can know more about child handling at a more detailed level, so I can have more knowledge when handling a child.
- Family and Friends of the Baby
    - Being a friend to new parents I would like to be less anxious about holding a child for the first time and to assure my new parent friends that I can hold a child properly to have them feel less anxious about me holding their child. Some of the fears I have is not properly supporting a child when holding it or some of the dangers that could be present that I am unaware of when holding the baby.
    - As a grandparent I would like to show my competency in holding a baby to my children, so they feel less anxious about me handling fragile packaging.

    - As a six-year-old I would like to show everyone that I am capable of important things such as being able to properly show I am all grown up to handle a baby now.
    - I am a babysitter who would like to train other babysitters how to properly hold a baby to provide confidence to them and the parents of the baby before they need to watch the baby.
- Guardians of the Baby
    - As a parent, I want to know more about child handling, and how safe I need to be when handling them so that I can be a better parent.
    - As a parent, I want my older children to be able to understand how fragile a child is so they can safely interact with their new siblings.
- Educator / Trainer
    - As an Educator / Trainer, I want to teach parents about child handling in a way that gives them more information and experience, so that I can do my job more efficiently and train them better.

High Level Use Cases

Rotate Baby - rotating the baby in various horizontal and vertical positions.

Shake Baby - shaking the baby gently, moderately, and violently.

Drop Baby - dropping the baby from various heights.

Read Condition - read the condition of the baby (good/dangerous/bad).

Use Case Descriptions

1. Name: **Rotate Baby**
2. Participating Actors: Healthcare Practitioner, Trainer, Baby Handler..

3. Entry Condition: Charged Battery.
4. Exit Condition: Baby Makes Sound.
5. Flow of Events:
    a. Pick up baby sim.
    b. Rotate baby sim.
    c. Emit sound/no sound from baby.
6. Special Requirements:
    a. If the baby is upside down, have the baby cry until the baby is upright.

---

1. Name: **Shake Baby**
2. Participating Actors: Healthcare Practitioner, Trainer, Baby Handler
3. Entry Condition: Charged Battery
4. Exit Condition: Baby Displays Light from LEDs
5. Flow of Events:
    a. Pick up baby sim
    b. Shake baby sim
    c. Emit light from LEDs
6. Special Requirements:
    a. If the baby is shaken at max force level, display red
    b. If the baby is shaken at moderate force level, display yellow
    c. If the baby is shaken at gentle force level, display green.

---

1. Name: **Drop Baby**
2. Participating Actors: Healthcare Practitioner, Trainer, Guardian, Baby Handler.
3. Entry Condition: Charged Battery.
4. Exit Condition: Baby Cries and Displays Light from LEDs.
5. Flow of Events:
    a. Pick up baby sim.
    b. Drop baby sim from a relative height.
    c. Emit sound.
    d. Emit light.
6. Special Requirements:
    a. If time to fall is below a half a second display yellow.
    b. If time to fall is above half a second display red.

---

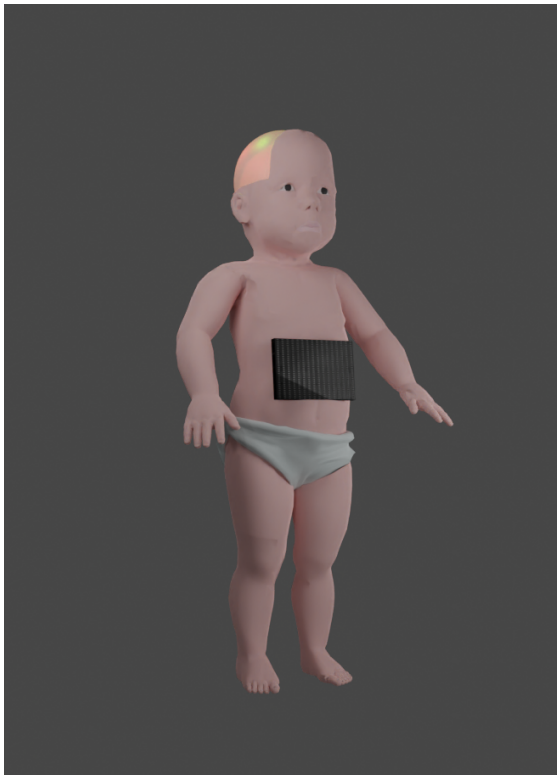
1. Name: **Read Condition**
2. Participating Actors: Healthcare Practitioner, Trainer, Baby Handler
3. Entry Condition: Charged Battery and (Shake or Drop or Rotate event)
4. Exit Condition: Condition Understood
5. Flow of Events:
    a. Perform either a Rotate or Drop or Shake operation on the baby sim

b. Read the condition

c. Reference the User Manual

d. Understand the condition

3. USER EXPERIENCE WIREFRAMES



4. NON FUNCTIONAL REQUIREMENTS

Other Systems

What other systems to interface with? Dependent on?.

The system is self contained and does not depend on outside systems to operate with full usability.

Security

There will be an emphasis on protecting end users from any damage that can occur when dealing with any lithium-ion battery packs. There will also be considerations to dealing with making it difficult to accidentally be in direct contact with the system itself. We will provide this functionality by encasing the system within the baby itself.

Performance

Up time - We are planning on having the system be up and fully usable only when required to be and off when necessary.

Response time? - The response time of the system should be near real time responsiveness with a plus / minus a half second delay.

Maintainability

How will it be maintained? - The system will be maintained remotely when necessary to incorporate other features requested by the customer using a version control system such as Github with full documentation to provide full disclosure of the changes made to the system over time for the customer and users alike.

Budget

Shaken Baby Simulator Prototype Shopping List				
Product Page	**Description**	**Price (in dollars $)**	**Quantity to Purchase**	
https://www.adafruit.com/product/2809	**Adafruit LIS3DH Triple-Axis Accelerometer (+-2g/4g/8g/16g)**	**$4.95**	**1**	
https://www.adafruit.com/product/4884	**Black Adafruit Feather RP2040**	**$11.95**	**1**	
https://www.adafrui	**Raspberry Pi Pico RP2040**	**$4.00**	**1**	

t.com/product/4864				
https://www.adafruit.com/product/2671	2mm Pitch 40-Pin Break-apart Male Headers - Pack of 5	$2.95	1	
https://www.adafruit.com/product/159	Diffused RGB (tri-color) LED - Common Anode	$6.00	3	
https://www.adafruit.com/product/3006	Adafruit I2S 3W Class D Amplifier Breakout - MAX98357A	$5.95	1	
https://www.adafruit.com/product/3968	Speaker - 40mm Diameter - 4 Ohm 3 Watt	$4.95	1	
https://www.adafruit.com/product/258	Lithium Ion Polymer Battery - 3.7v 1200mAh	$9.95	1	
https://www.adafruit.com/product/1954	Premium Female/Male 'Extension' Jumper Wires - 20 x 6"	$1.95	1	
https://www.adafruit.com/product/902	Adafruit Bicolor LED Square Pixel Matrix with I2C Backpack	$15.95		
	Price Total	$68.60		
	Optional Choices		1	
https://www.adafruit.com/product/2738	Raspberry Pi Sense HAT - any Raspberry Pi with 2x20 Connector	$39.95		

*Some products may be discontinued and / or unavailable due to supply shortages / demand.

**Prices subject to change.

5. PROJECT ORGANIZATION -

Breakdown of major tasks and schedule

Key
NC - Not Comfortable
C - Comfortable
P - Partial Experience

Matrix of Responsibilities / Skills

Skills Needed	Skills Acquired		
	Drew S.	Boyd L.	Tyler L.
Circuit Python	NC		
Github / Git	P	P	P
User Interface	P		P
Product Design	P		P
Baby Research	P	P	
Knowledge of Microcontrollers	NC		
Knowledge of Accelerometers	NC		
Knowledge of Speakers	NC		
Two-Dimensional Array Programming	NC	P	P
Debugging / Troubleshooting	P		P

Skills Needed	Skills Acquired		
Testing	P		P

Responsibilities	Drew S.	Boyd L.	Tyler L.
Testing	P		
Anatomical Research	NC	X	
LED Matrix Programming	NC		X
Accelerometer Programming	NC	X	X
Hardware Research	P		X
User Interface Design	p	x	
Circuitry Design	NC		X

PERT / Gantt Chart

Shaken Baby

By : Tyler, Drew, Boyd

Project Start Date 1/24/2022 (Monday)

Project Lead Team

TASK	LEAD	START	END	DAYS
Planning	Team	Mon 1/24/202	Sun 1/30/22	9
Risk Management	Team	Mon 1/31/22	Sun 2/6/22	9
Designing	Team	Mon 2/7/22	Sun 2/13/22	9
Protoype	Team	Mon 2/14/22	Sun 2/20/22	4
Administration	Team	Mon 2/21/22	Sun 2/27/22	4
Implementation	Team	Mon 2/28/202	Sun 3/6/22	5
Sprint 1	Team	Mon 3/7/2022	Sun 3/20/22	14
Sprit 2	Team	Mon 3/21/202	Sun 4/3/22	14
Sprint 3	Team	Mon 4/4/2022	Sun 4/17/22	10
Sprint 4	Team	Mon 4/18/202	Sun 5/1/22	7
Testing	Team	Mon 5/2/2022	Fri 5/8/22	7

6. VALIDATION PLAN

Test Strategy

What is the definition of done?

The definition of done is defined as all components have been integrated together within the system and the system has been fully tested for each expected input output of every use case.

What does success look like?

Success is the measure of how effectively we convey to the user of the system how to properly handle a baby given the force applied in various situations using clearly defined indicators built into the system.

If the user can instantly tell when they have applied too much force to the baby in any given situation this is how we will know we have achieved success.

7. RISK MANAGEMENT PLAN

What are the known risks and how will they be handled?

Research

Hardware / Software Research

In this part we are going to discuss all the hardware decisions as related to the design of the shaken baby simulator. We will discuss the differences between a microcontroller versus a micro-computer. After a brief discussion we will discuss all the major components of the system and why we decided to choose some components over others. We will conclude with a brief discussion of the software that is needed to control the hardware of the system.

Microcontroller Vs. Micro-Computer:

There is a distinct reason we used a Micro-Controller such as an Arduino versus a Micro-Computer such as a Raspberry Pi in this implementation of the system. A Micro-Controller has a dedicated program that runs in a loop that provides a specific functionality and is very suitable for connecting various components such as

temperature, acceleration and light sensors and emitters. A Micro-Computer is more for variable uses across a variety of different applications depending on the needs of the user at a particular time.

With the previous discussion in mind therefore we decided to choose a Microcontroller instead of a Micro-Computer for the project implementation. The specific Micro-Controller that we chose was a decision that was based on a few aspects of the project such as availability of hardware due to the on-going chip storage as well as the cost of the Microcontroller as well as the API (Application Programmer Interface) to interface with the various sensor components we needed to develop the project well. We ended up choosing the Adafruit Feather RP2040 for a few reasons over other variations. First is that we needed to have a battery connected to the Microcontroller that was housed inside the baby simulator. The second is that the Feather RP2040 supports built-in charger regulation using USB-C which is becoming the standard in charging ports on various devices. The third is that it has plenty of GPIO ports to interface with various components using Circuit Python.

Hardware Decisions:

There are a few main components to the project that will be described in detail here. We will also describe a few advantages that we can leverage by choosing these specific components. The main system is built by having an accelerometer connected to a feather RP2040 which is also connected to a 8x8 pixel grid plus a speaker and led light for sound/light respectively. The reason we chose the feather RP2040 instead of another microcontroller is due to the ease of use in connecting various other components needed for the project using Circuit Python libraries which are controlled using the Python programming language. Also the Feather RP2040 has the Raspberry PI 2040 processor chip which has PIO ports (Programmable Input Output) . These ports allow you to write software to interface properly with all kinds of different components that we would otherwise not be able to do. Everytime the feather RP2040 is powered on by either a dedicated power source such as a USB-C charging cable or a portable lithium ion battery pack the script that is saved runs automatically and nearly instantaneously. The 8x8 led grid we have chosen perfectly displays all three of the colors we wanted to show: green, red, yellow. These colors correspond to the various levels of danger with green being the least worrying to red being the most force being applied to the simulator. The backpack that attaches to the 8x8 led grid is necessary to interface with the 8x8 grid with only four wires instead of sixteen. This reduces the amount of wiring in the project. We went with a speaker that was not going to break with extended use of the system instead of getting a fragile speaker for cheaper.

Software API Overview:

The main component to programming the project comes from the various libraries provided by the Circuit Python framework which allows you to interface with various components using a lot of microcontrollers by simply programming in a relatively familiar and known programming language Python. The documentation can be found here at this website https://circuitpython.org/.

The getting started with Circuit Python page can be found here: https://learn.adafruit.com/welcome-to-circuitpython/circuitpython-libraries.

The overview of the feather RP2040 can be found here as well:

https://learn.adafruit.com/adafruit-feather-rp2040-pico.

Risk Identification

The biggest risk is an accurate range of force to damage reading, measuring force accurately on a doll can be difficult with the tools we have available.

Competition, being able to be a viable counterpart to the other shaken baby simulator we found is important

Making sure the doll is able to go through testing and last, with both battery, and being able to last without damage

Risk Prioritization

Prioritized list (biggest risk -> lowest risk)

1. Accuracy
2. Damage Possibility
3. Battery
4. Competition

Risk Mitigation

1.Accuracy: Can be mitigated by thorough testing and research. Testing can help us dial in measurements and ensure that the data collected is accurate, and research can ensure we know the ranges of force that the doll should respond to.

2.Damage: This can be mitigated by testing different actions and force on the doll and seeing how the technical components handle it, we can work through

designs and different measure to ensure that the doll can survive the force needed to test, such as other accidental damages such as dropping.

3.Battery: This should not be an issue, as batteries in general can last a long time, but this should still be a general consideration for the doll, as having a long battery life will be important for medical demonstrations and training sessions

4.Competition: We can research competition to come up with ways to differentiate in both price and features to insure that our doll stands apart from other competitors that currently exist.

8. CONFIGURATION AND VERSION CONTROL

Github will be used for configuration and version control.

www.github.com

9. TOOLS

Mu - Python Text Edition -> https://codewith.mu/

Must Have (This is due to the full writes when saving as explained further here **:** **https://learn.adafruit.com/welcome-to-circuitpython/recommended-editors)**

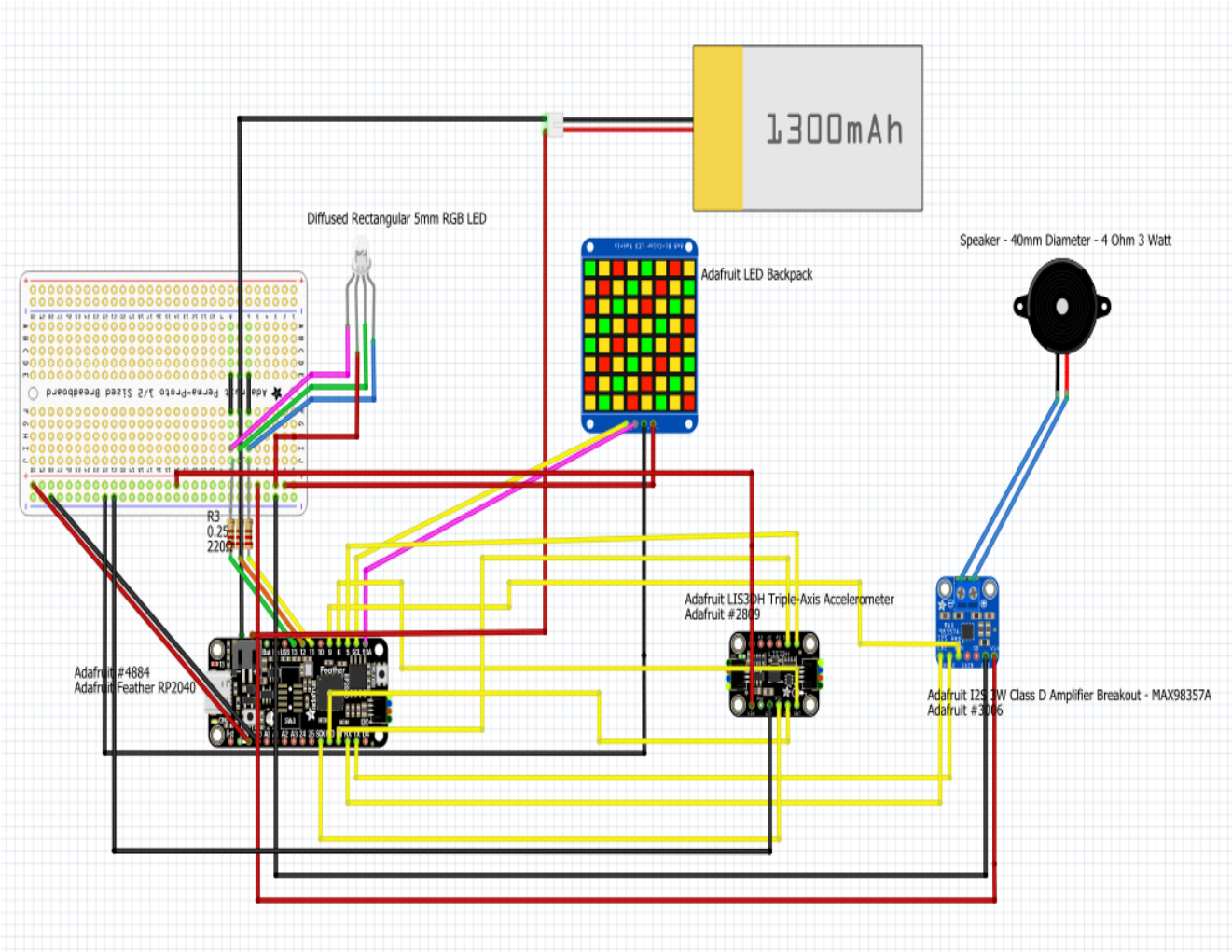
Github (Software Version Control / Project Management) -> https://github.com/

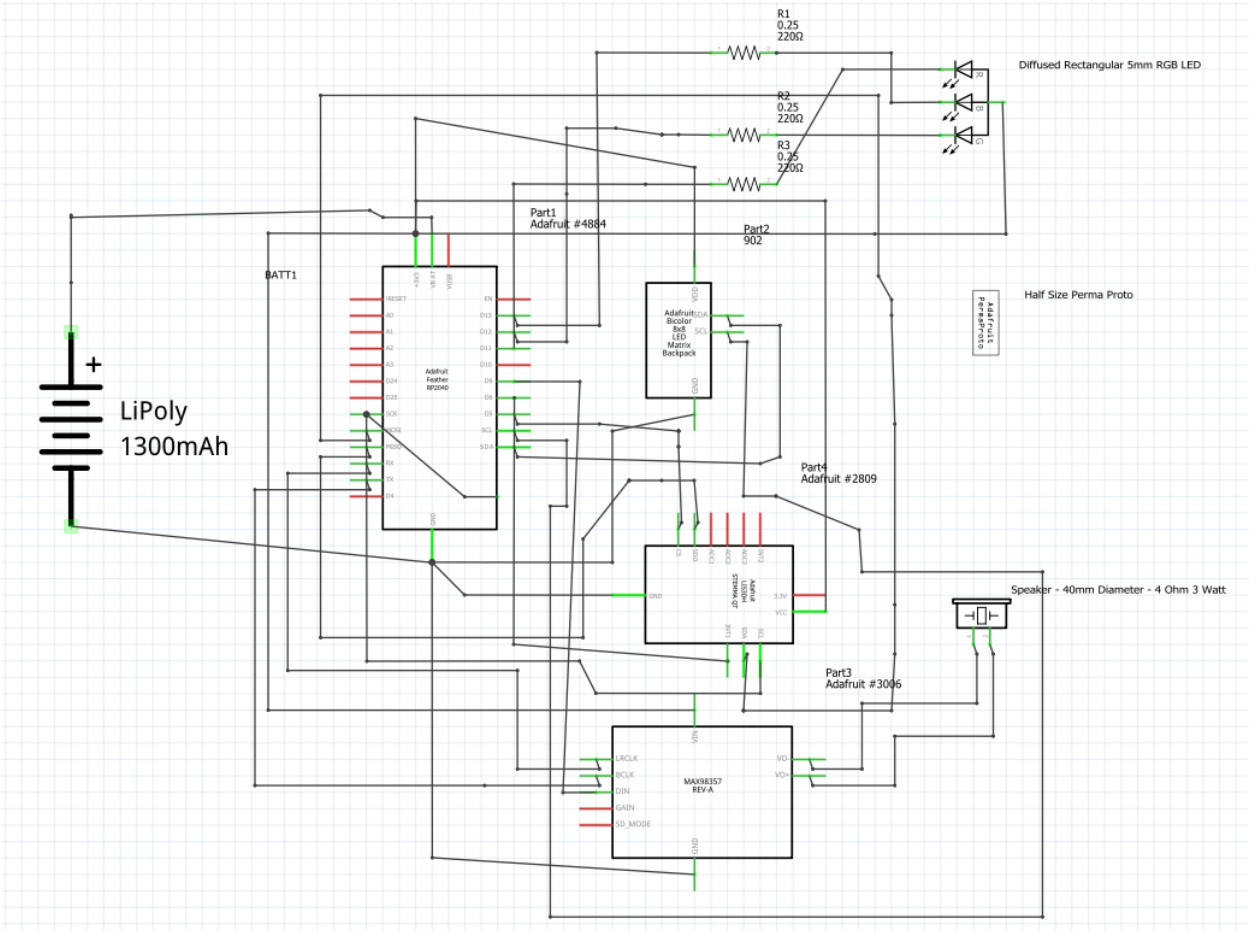
Circuit Python -> https://circuitpython.org/

Adafruit Learn -> https://learn.adafruit.com/

Fritzing Circuit Builder -> https://fritzing.org/

10. ARCHITECTURE





11. RELATED LINKS

https://codewith.mu/

https://learn.adafruit.com/welcome-to-circuitpython

https://learn.adafruit.com/adafruit-feather-rp2040-pico

https://learn.adafruit.com/adafruit-lis3dh-triple-axis-accelerometer-breakout

https://learn.adafruit.com/adafruit-led-backpack/1-2-8x8-matrix

https://projects.raspberrypi.org/en

https://www.recantha.co.uk/blog/?p=20950

https://fritzing.org/