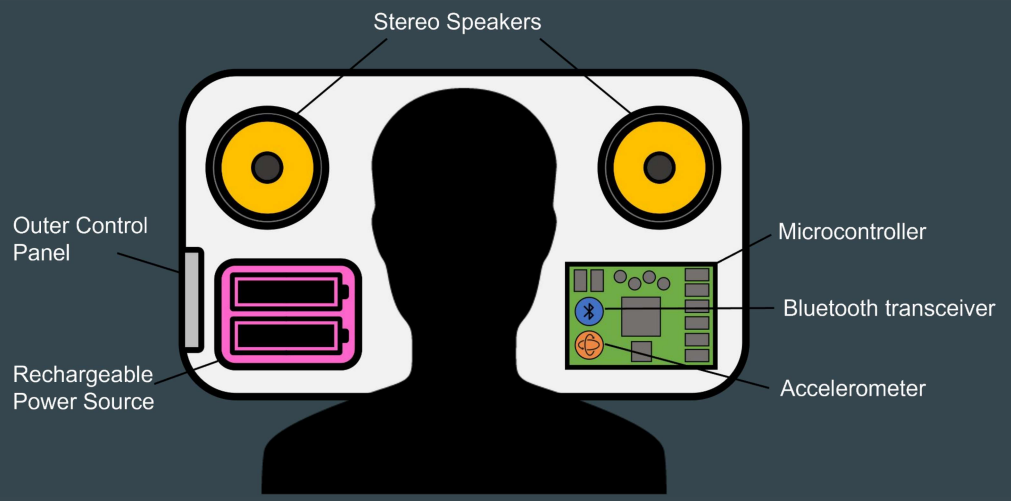
Group 46

Christian Tomlinson	Computer Engineering
David Kipikash	Computer Engineering
Jason Davis	Computer Engineering
Kevin Perez	Computer Engineering

Project Introduction

- What is Snoozie?
  - “Smart-Pillow”
  - Hardware product & Software application
- What does it do?
  - All in-one sleep-wellness tracking solution
  - Alarm clock
  - Media player
- Components
  - Stereo Speakers
  - Bluetooth Transceiver
  - Light Sensor
  - Pressure Sensor
  - Gyroscopic Sensor
  - Temperature Sensor



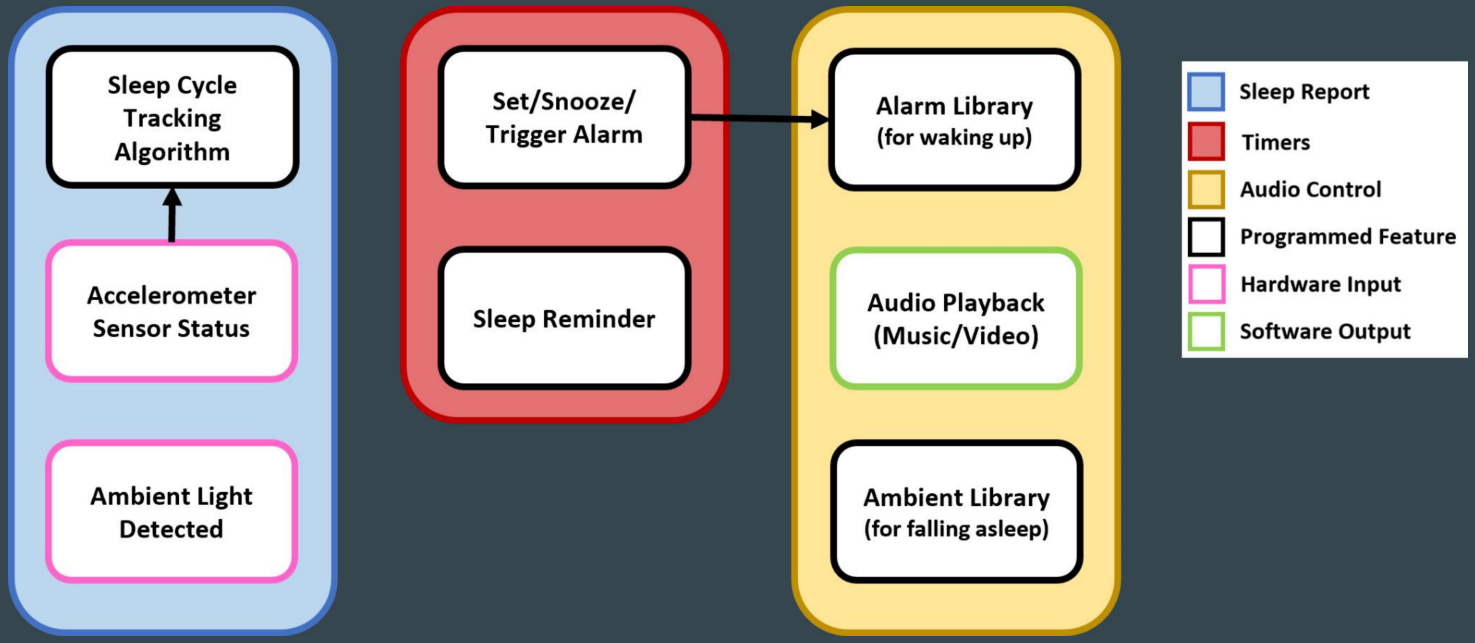
components not to scale

Project Inspiration

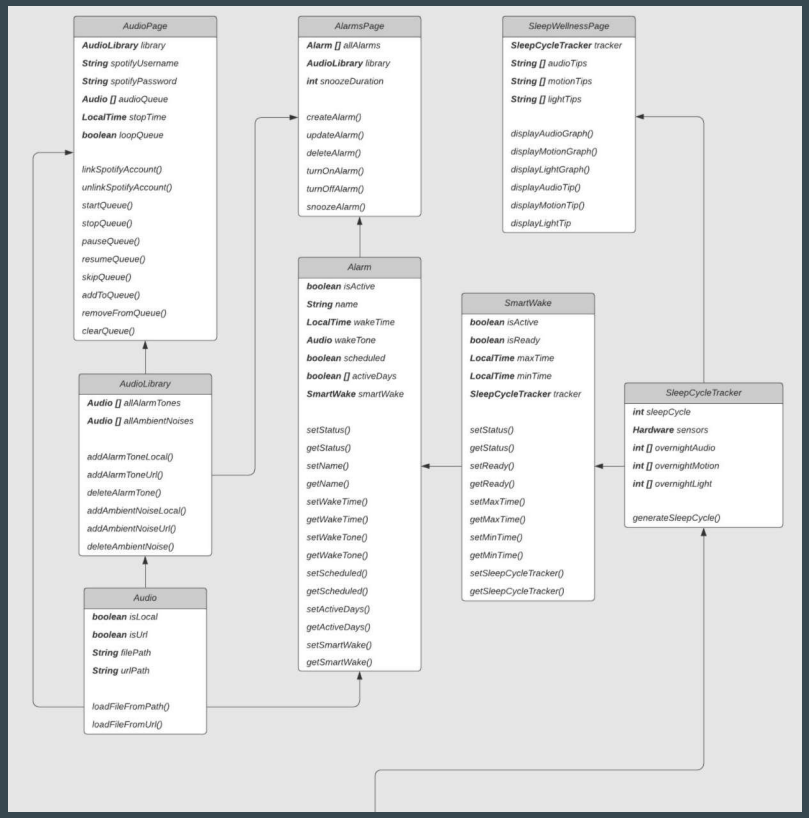
- Team of sleep-deprived engineering students
- Beginning
  - Speakers in a pillow -> much more!
- Wanted to benefit those who struggle with sleeping
  - Educate people on good sleeping habits
- Hundreds of sleep tracking apps that are NO GOOD!
- Wanted to provide a complete, integrated solution

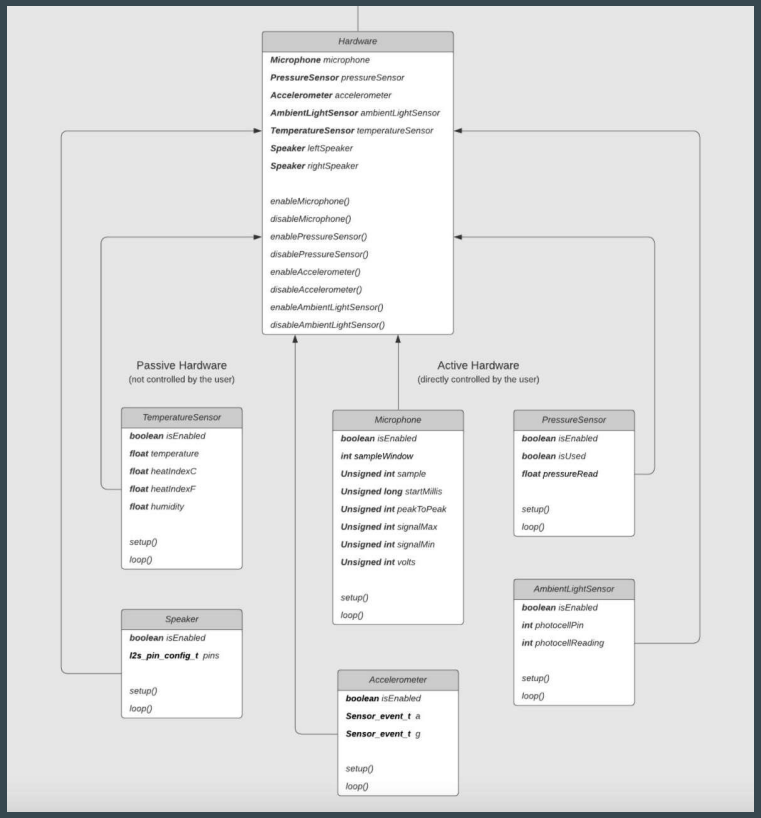
Project Software

Project Software Block Diagram



Project Software Class Diagram

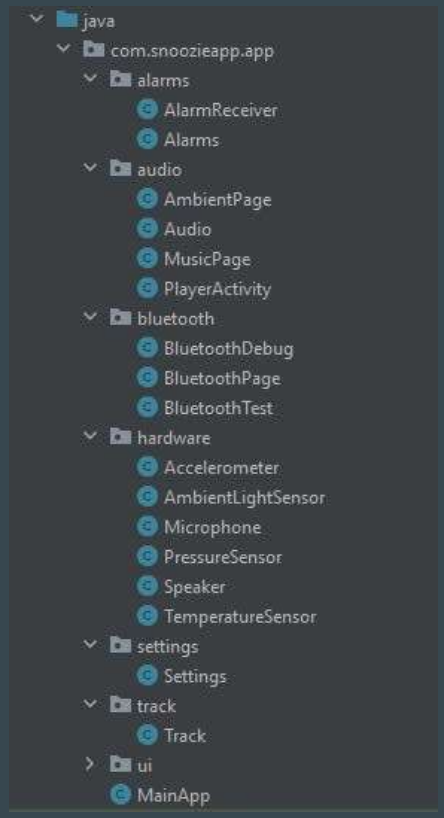




Java (Classes / Individually Testable)

- Since we want to individually test our classes, we decided to separate them in different components.

- This way we can avoid any classes from depending on one another, allowing us to individually test the classes with ease.



Software Testing Example

Alarm Testing

- Check to make sure the app is creating new alarms, updating, and deleting existing alarms.

- Methods:
  - testCreateAlarm()
  - testCancelAlarm()

Firmware Testing Example

Temperature Sensor Testing

- We test that the conversion of Celsius and Fahrenheit works correctly. We also make sure that the sensor returns valid results for humidity and temperature.

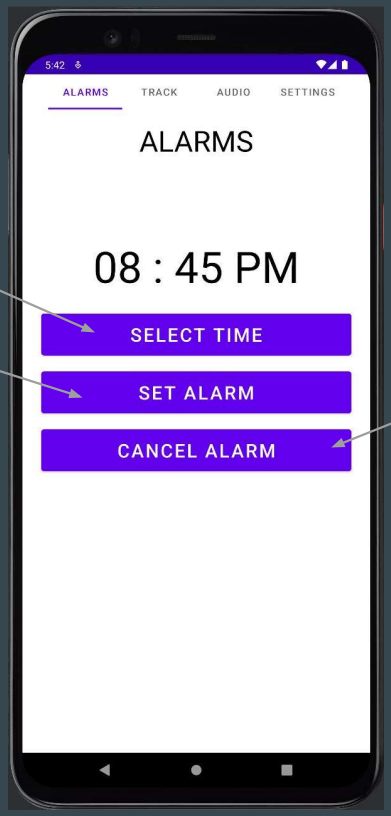
- Methods:
  - testDevice()
  - testValues()

- Fields:
  - boolean isEnabled - Set True if sensor is active
  - float temperature - Get temperature reading
  - float humidity - Get humidity reading
  - float testVal - Check conversion values

Alarms Page

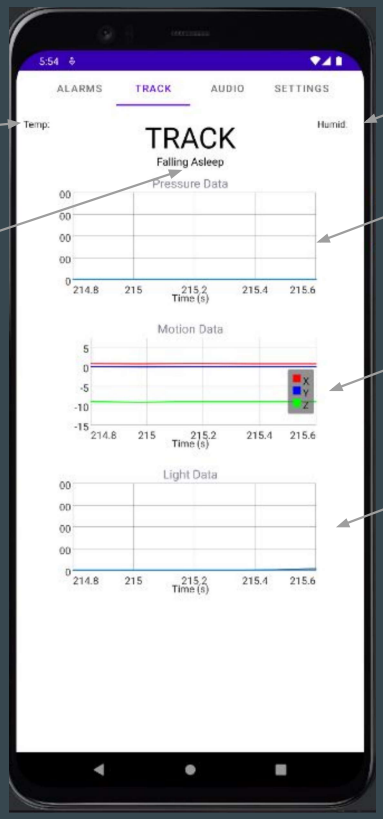
Select alarm

Set alarm



Cancel Alarm

Track Page



Temperature Reading

Sleep Stage

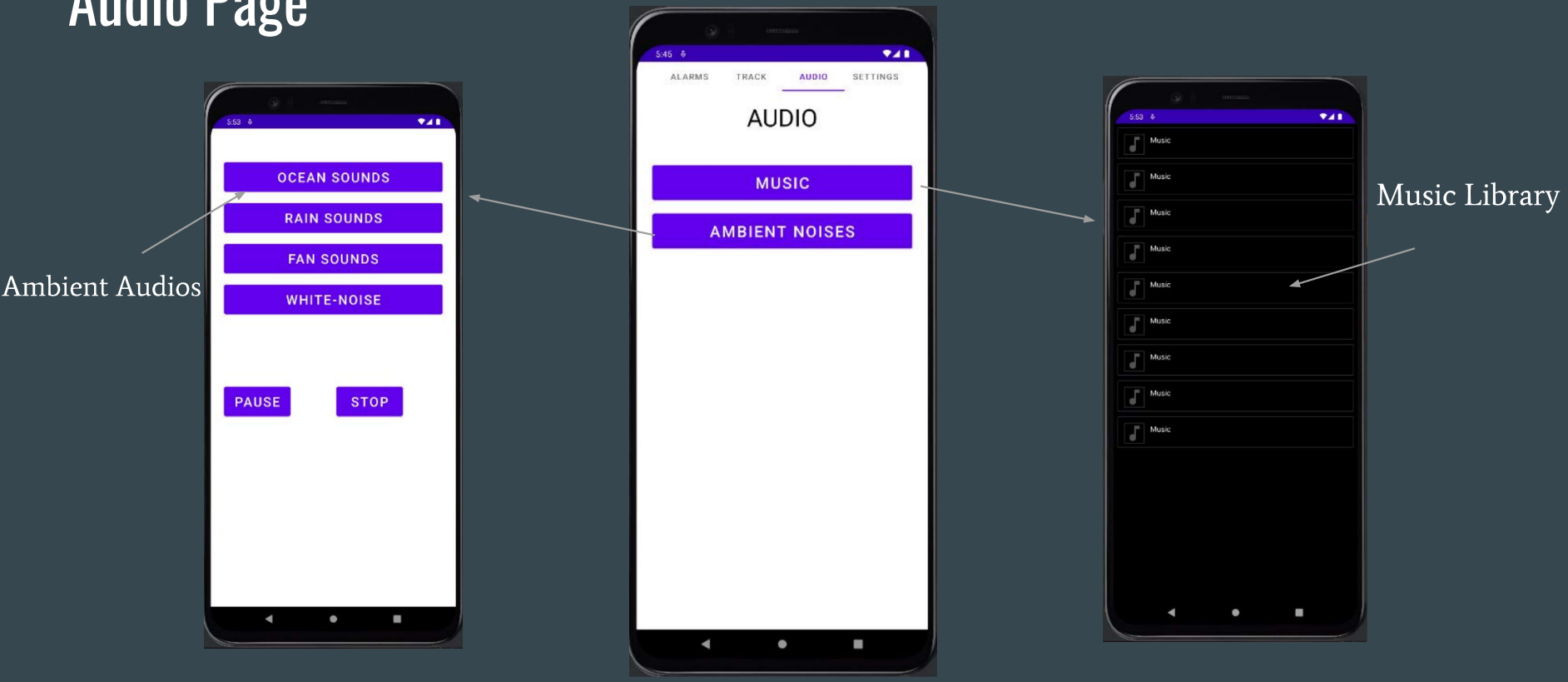
Humidity Reading

Pressure Graph

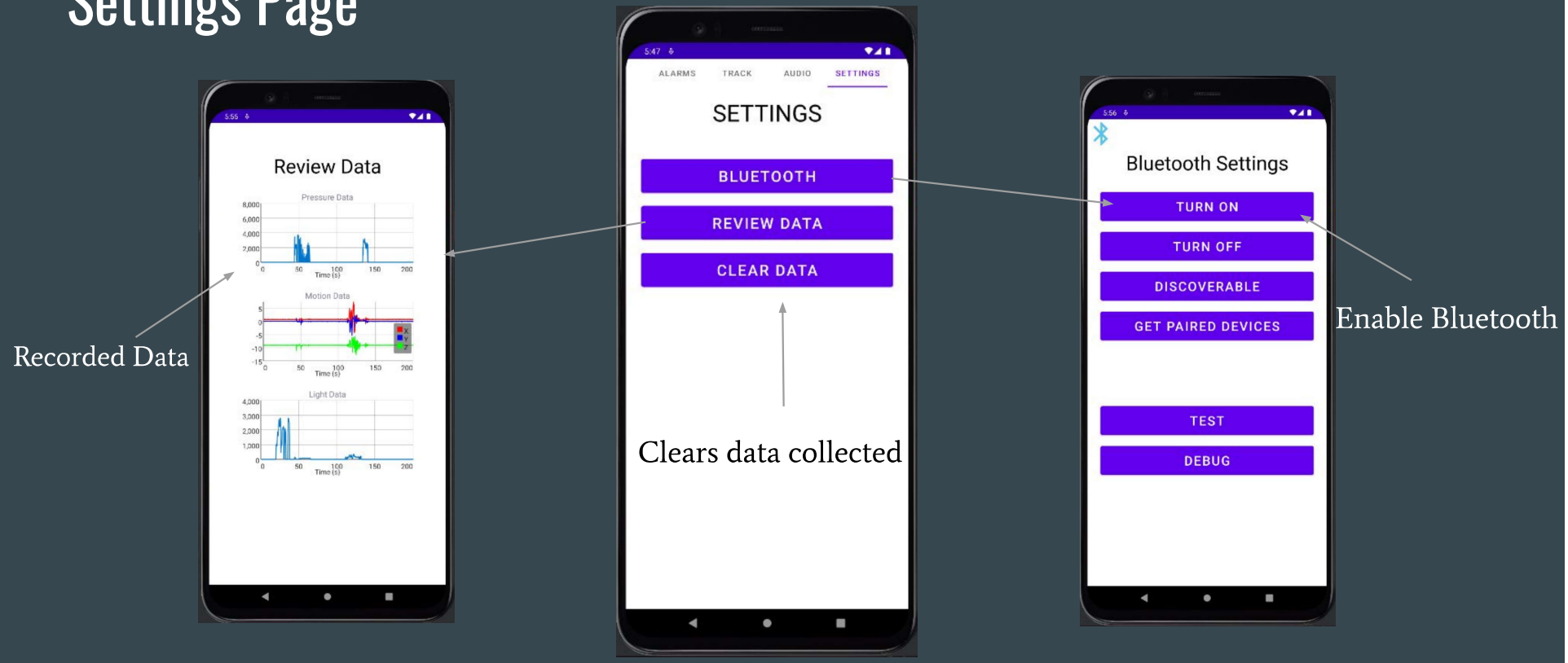
Motion Graph

Light Graph

Audio Page


Ambient Audios
OCEAN SOUNDS
RAIN SOUNDS
FAN SOUNDS
WHITE-NOISE
PAUSE
STOP
ALARMS
TRACK
AUDIO
SETTINGS
AUDIO
MUSIC
AMBIENT NOISES
Music Library
Music

Settings Page



Sleep Cycle Algorithm

- How to track a user's sleep cycles
  - As users transition through the various stages, they bodies change behavior (trackable)

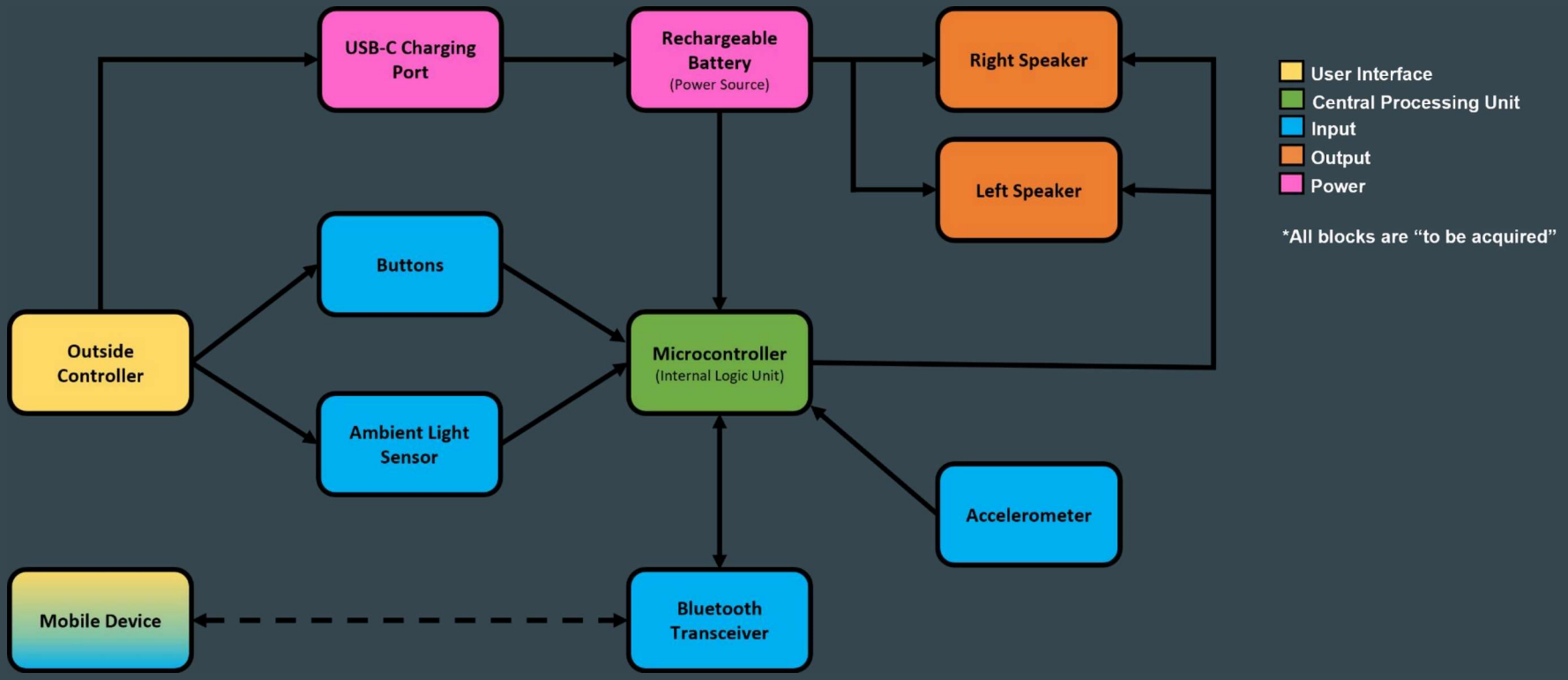
- Initial Plan - Machine Learning Algorithm
  - Too time demanding
  - Not enough data collection

- Setback - No Audio Data
  - Algorithm became fully dependent on pressure sensor and accelerometer

- Final Implementation!
  - Timed sleep cycles + Pressure sensor data
    - Wake detection

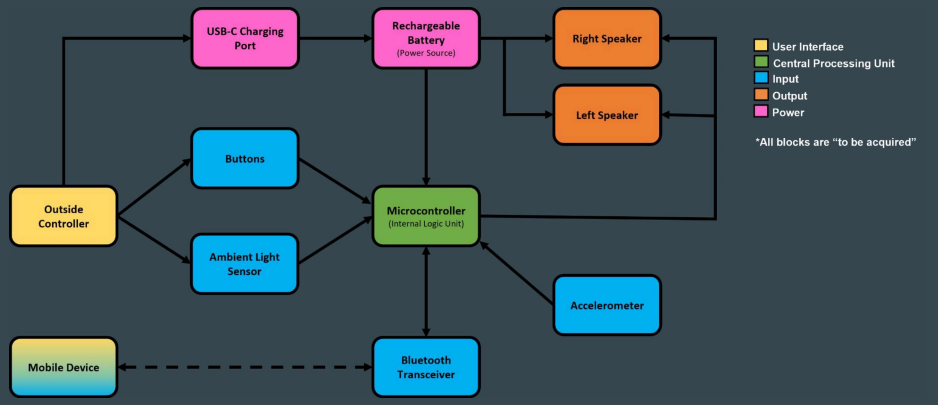
Project Firmware

Hardware Block Diagram

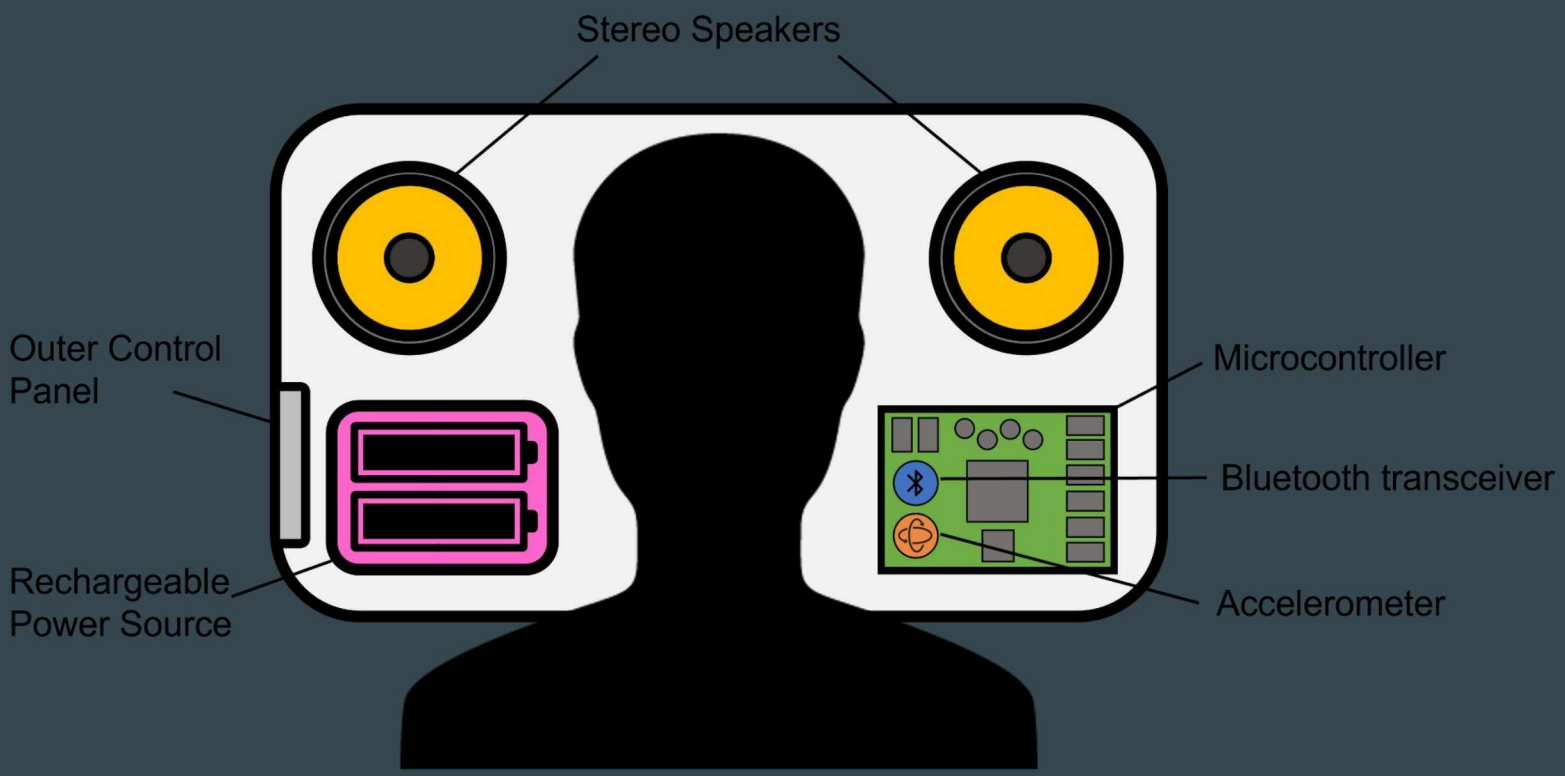


Hardware Block Diagram

- User interface
  - Allows users to directly interact with device
  - Done through the outer control panel
- Input (Blue)
  - Composes of light sensor, bluetooth, accelerometer, and buttons
  - Directly interfaces with microcontroller to provide feedback for user input
  - Collects data on user's sleeping patterns
- Microcontroller (Green)
  - Brain of the device
  - Manages inputs and outputs of Snoozie
- Power source (Pink)
  - Rechargeable battery
  - Exposed charging port on the control panel
- Output (Orange)
  - Composed of both left and right speaker
  - Broadcasts audio
  - Connected to rechargeable power source

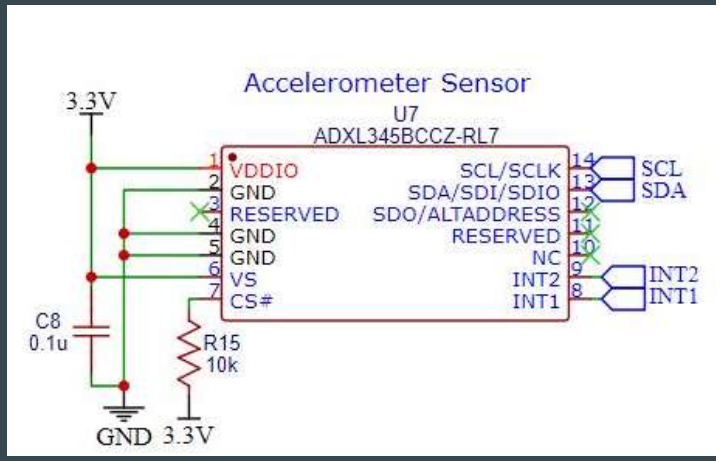


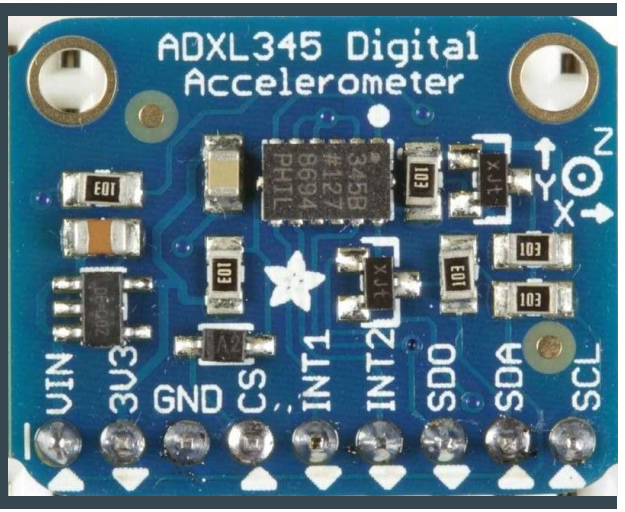
Pillow: Top-View



Accelerometer

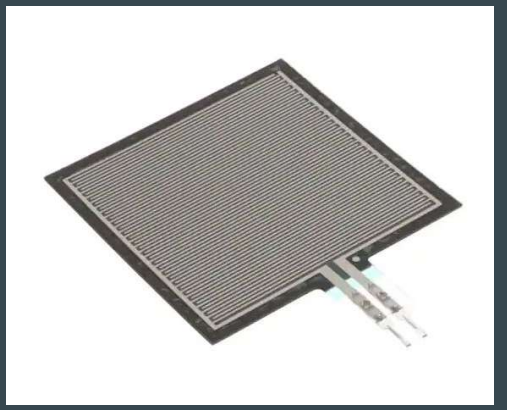
- Part chosen: ADXL-345
- Records motion during the night and data acquired will be transmitted to the app (via Bluetooth)
- Part information
  - Price: $9.99
  - Power: 2V-3.6V
  - Interface: I2C, SPI
- Advantages
  - Allows us to carry out planned functions and receive data based on user's movements
  - Compatible with microcontroller
- Note on testing
  - Needs to be connected to dev board and contain the module output numbers we can translate into necessary data

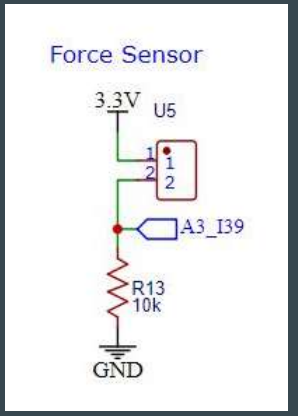




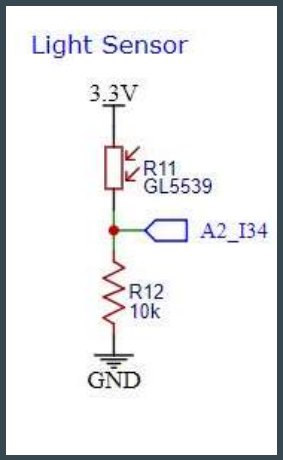
Pressure Sensor

- Analyze user's breathing patterns & muscle activity
- Large changes in pressure = movement
- Decreasing threshold on the sensor allows for the pillow to detect small variations of a user's breathing pattern
- Force Sensitive Resistor (FSR)
  - Group of sensors that detect pressure
  - Composed of 2 layers
  - Once the sensor is pressed, resistance will go down
  - Price: $7:00
  - Force range: 0-20lb.
  - Power supply: Any power supply is compatible
- Advantages
  - Low cost
  - Simple to use
  - Synchronizes with our goal of having the best pillow at the best price possible





Ambient Light Sensor



GL5537 Photocell

We decided to go with a photocell since they are basically a resistor that changes its resistive value depending on how much light is shining onto the surface.

Advantages:

- Very Cheap and Available
- Simple Setup
- Reliable Performance

Disadvantages:

- Very inaccurate
- Not able to be sold in other countries

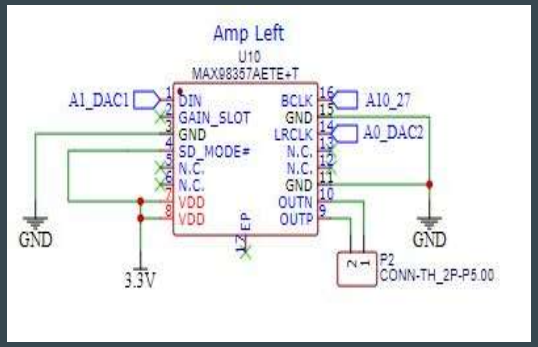
Price: $1

Interface: ADC

Sensitivity: 400 nm - 600 nm

Resistance: 200KOhms - 10KOhms

Speakers



- Gikfun 2” 4Ohm 3W Full Range Audio Speaker
- Snoozie needs a speaker that’s both compact and effective
- Run with a MAX9835 amp
- Part information:
  - Dimensions: 4.5 * 1.1 * 2.5
  - Max output power: 3 Watts
- Advantages
  - Works well in various frequencies
  - Works very well with our design due to they’re small size and amount of space they create for other materials

Temperature sensor

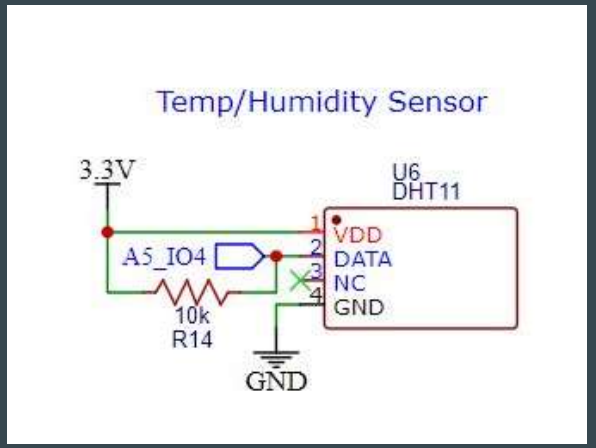
DHT11 Temperature & Humidity Sensor

The DHT11 will be used to collect temperature data while the pillow is being used.

Allows us to supervise the device's temperature and detect if the device is being used.

Advantages:

- Very Cheap
- Accurate
- Easy Implementation



Price: $5

Interface: GPIO

Accuracy: 0-50C +-2C

Sampling Rate: 1 Hz (1 Second)

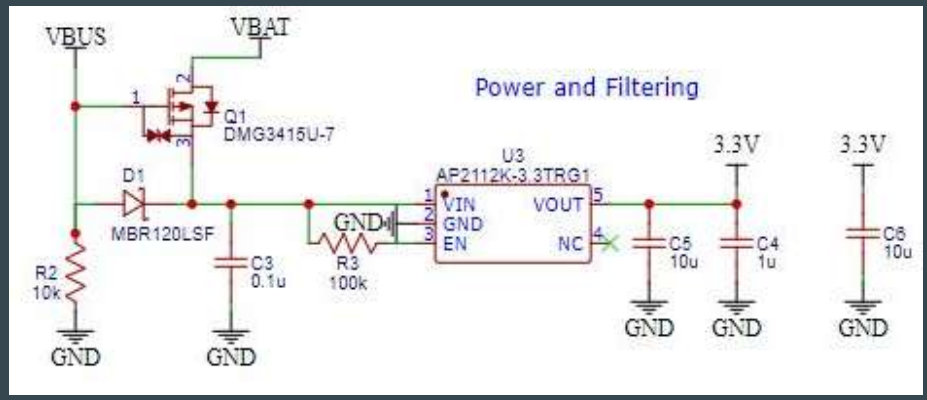
Power Regulation

AP2112K-3.3 Linear 3.3 voltage regulator

This will take power from the 5V USB port and regulate it to a constant 3.3V that the Microcontroller will need.

Advantages:

- Included in dev board
- Available
- Can use usb as input



Price: $0.15

Input Voltage Range: 2.5 - 6.0V

Output Voltage: 3.3

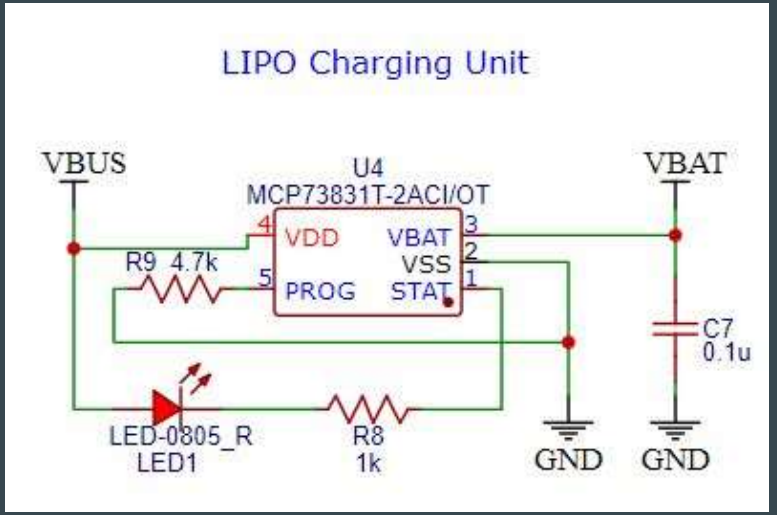
Battery Charger

MCP73831T lipo charging unit

This component will take input from the 5V USB bus and charge the lipo batteries and power the rest of the components.

Advantages:

- Cheap
- Integrated with the dev board



Price: $2

Supply Voltage: 3.75 - 6V

Start Threshold: 3.45V

Stop Threshold: 3.38V

Battery

Lithium Ion Battery Pack - adafruit

We will use this battery giving us approximately 3 days of battery life at the average current draw of the device.

Advantages:

- Decent Capacity
- Discrete size

Price: $25

Capacity: 6600 mAh

3 cells in parallel

3.7V LiPo

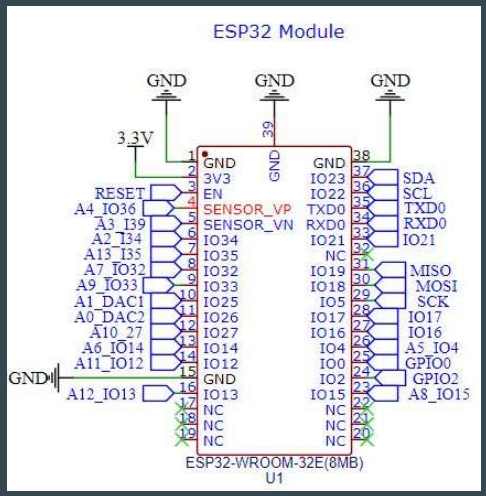
Microcontroller

ESP-32-WROOM-32E Microcontroller

This device will run and collect the data from the sensors and send it to the app. Also used for audio input and output.

Advantages:

- Cheap and Available
- Bluetooth capabilities
- Sufficient processing power



Price: $7

448 KB ROM

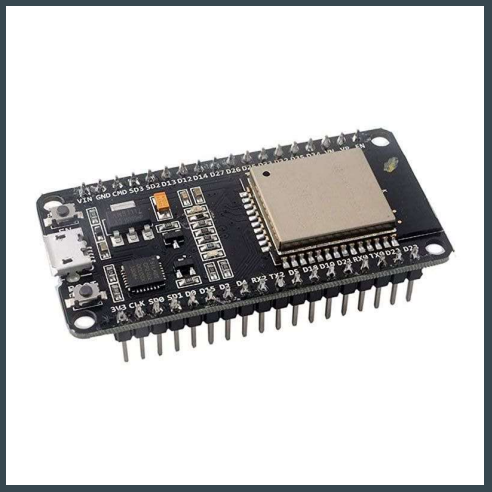
520 KB SRAM

Xtensa dual-core 32-bit LX6 microprocessor

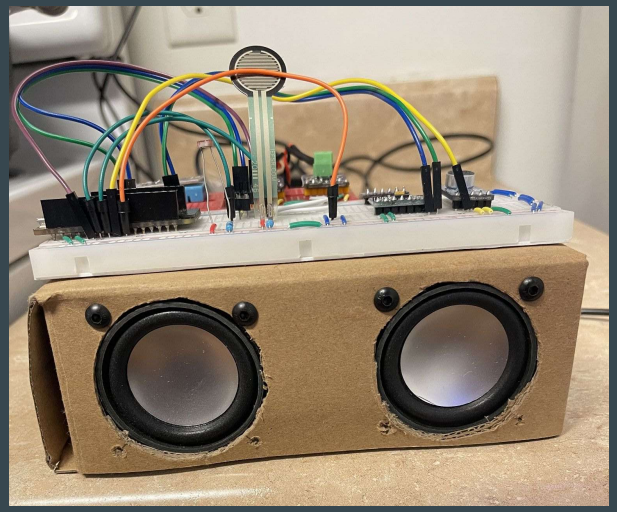
Operating voltage: 3.0 ~ 3.6V

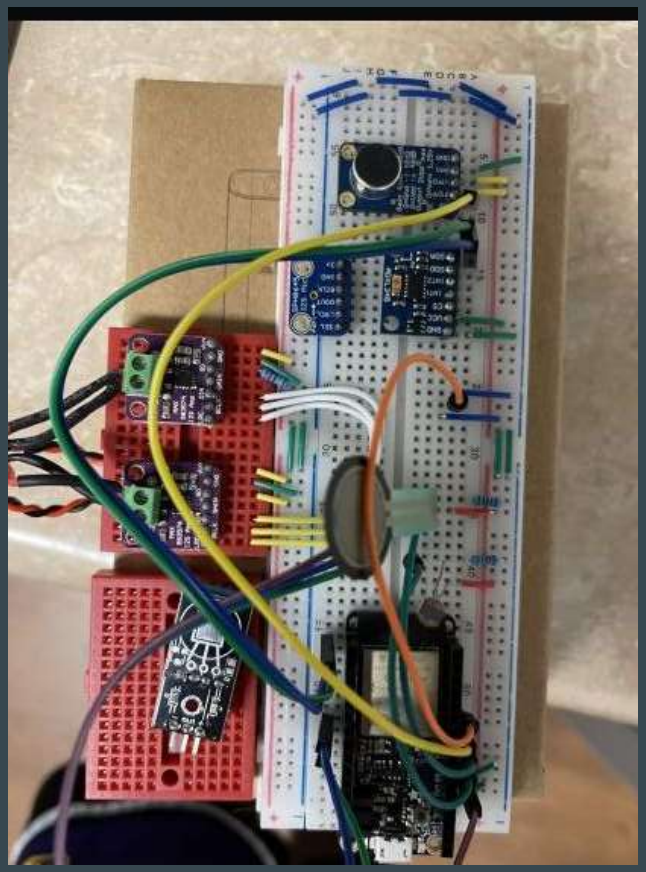
Bluetooth

- Bluetooth transceiver will connect with mobile application
- ESP 32 has an innate dual channel bluetooth
- Range will be sufficient to reach from the pillow to the nightstand
- Arduino IDE will test Bluetooth performance with our board
- Bluetooth Power Draw
  - Controller will have its power tested and without the BLE turned on (under 260 milliamps)
- Bluetooth range
  - Criteria is that the range has to reach at least 3 meters

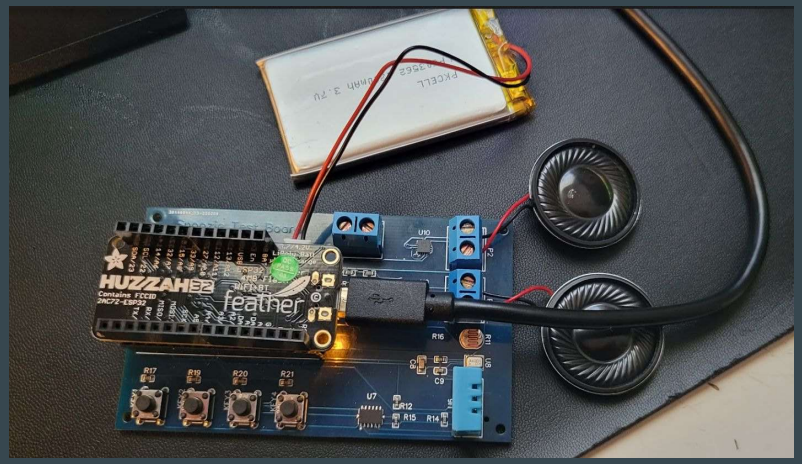


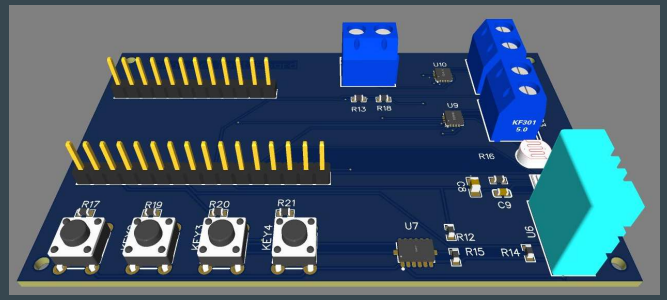
Initial Prototype

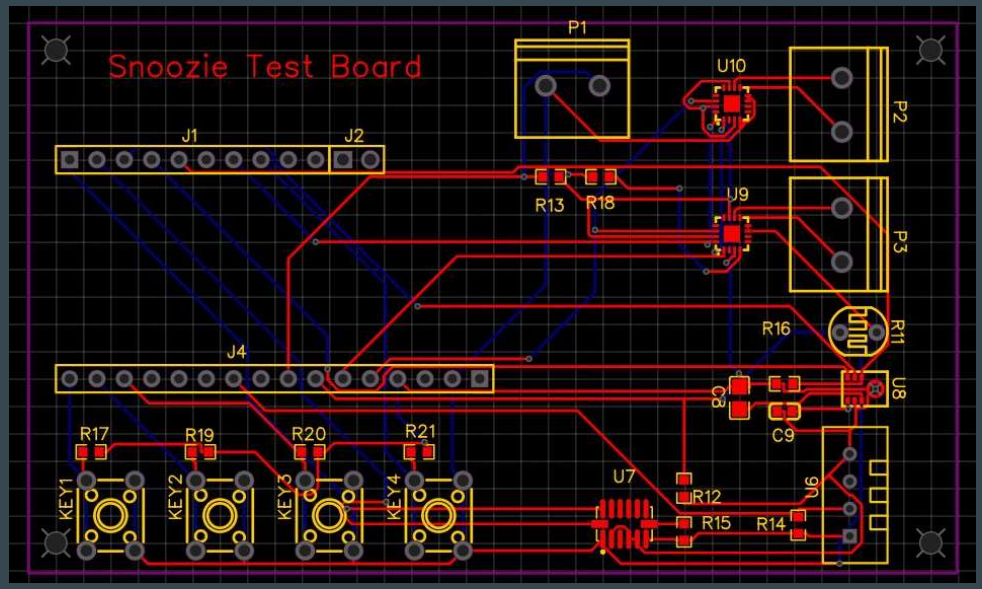




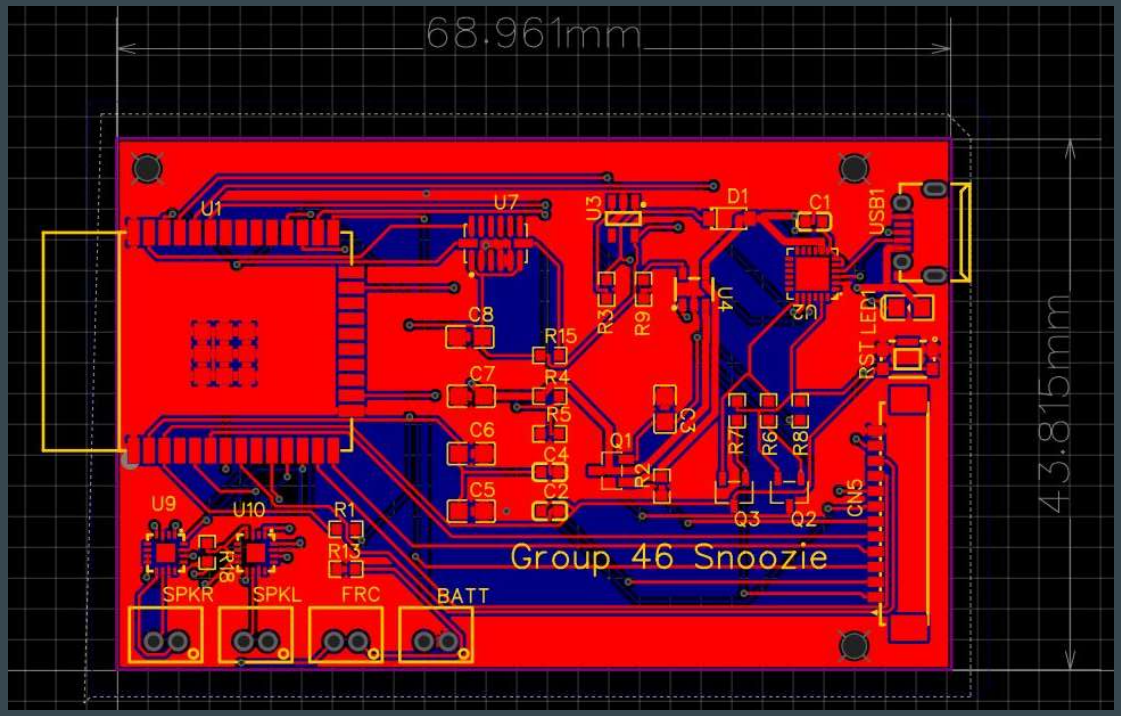
PCB Prototype

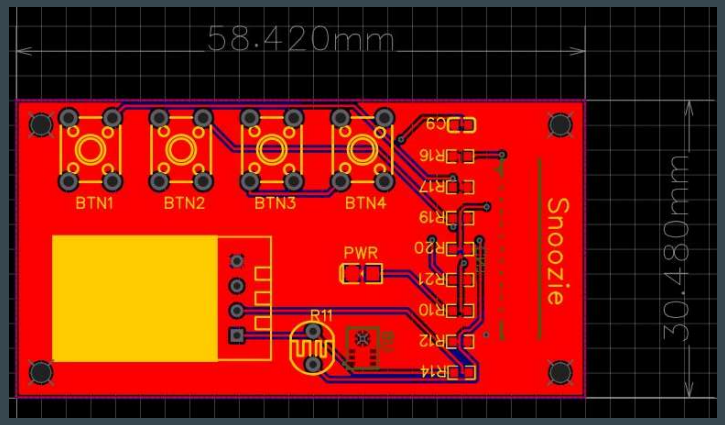




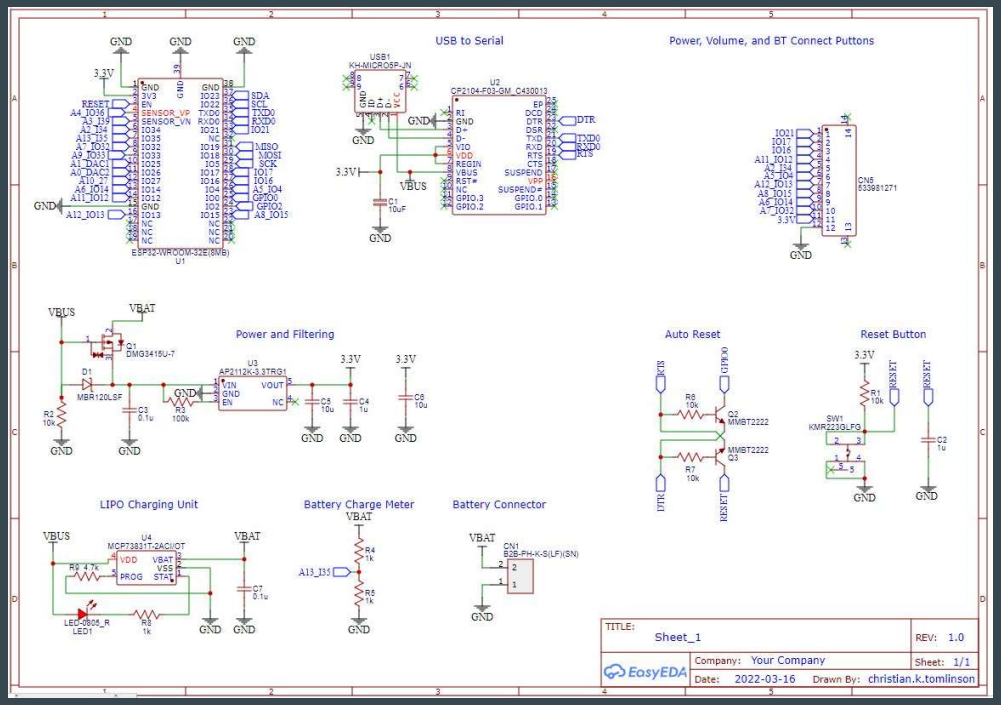


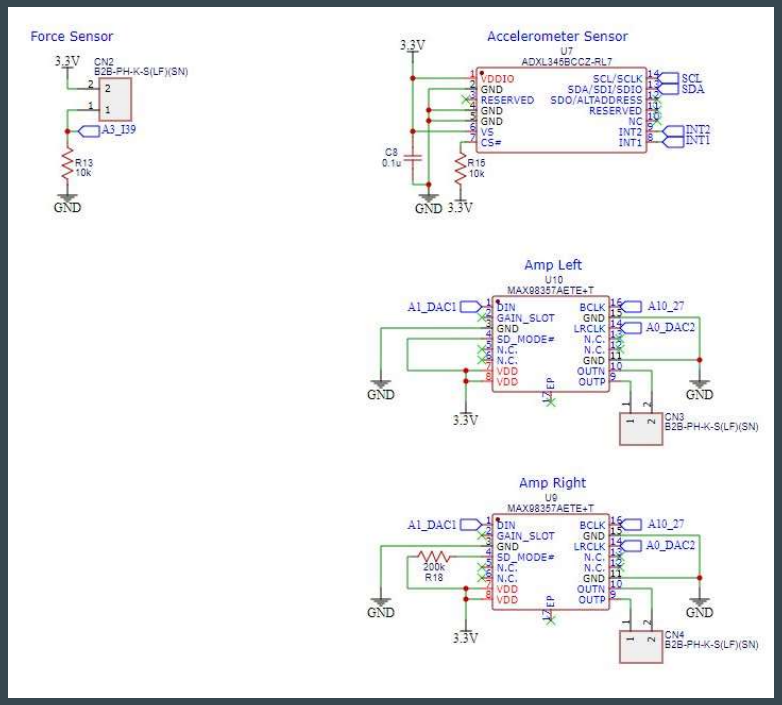
PCB Schematics



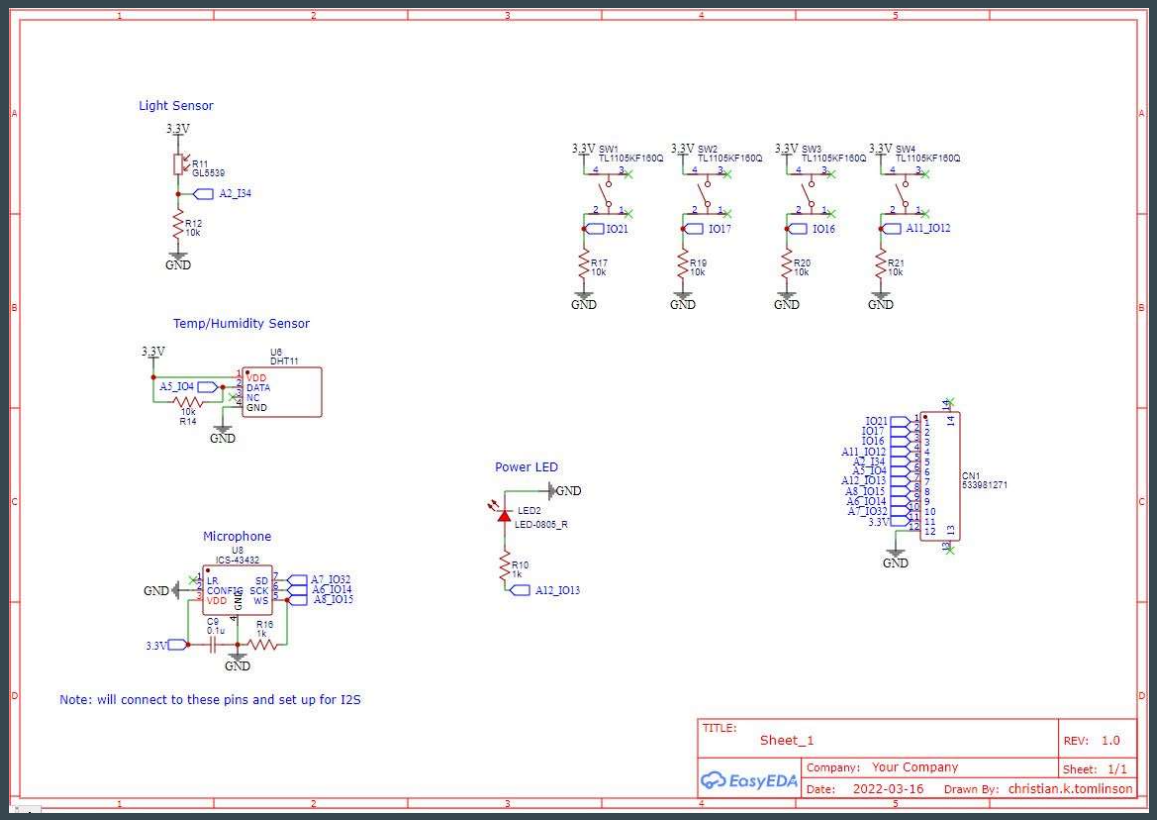


Schematic sheet





Panel Board Schematics



PCB Design


3914694A_Y8_220322
U1
U7
U3
D1
C1
USB1
U4
U5
LED1
RST
C8
R15
R3
R9
C7
R4
C3
R7
R6
R8
C6
R5
Q1
C4
R2
C5
C2
Q3
Q2
CN5
U9
U10
R1
R13
R18
SPKR
SPKL
FRC
BATT
Group 46 Snoozie


BTN1
BTN2
BTN3
BTN4
PWR
R11
Snoozie


CN1
U8
Snoozie
3914694A-Y5-220322

Final Design

Project Plans

Project Budgeting

Part Name	Role?	Price	Source
ESP32	Microcontroller/ Wireless Communication	$7.39	Espressif.com
MAX9814	Microphone	$8	Adafruit.com
cp2104	Serial Communications	$6.29 (2)	Amazon
Force Sensing Resistor	Pressure Sensor	$11.93 (2)	Amazon
PCB Parts and assembly		$250	

DHT11	Temperature Sensor	$5	Adafruit.com
Photocell (CDS Photoresistor)	Ambient Light Sensor	$1	Adafruit.com
ADXL-345	Accelerometer	$9.99	Adafruit.com
Quick Charge 3.0, Anker 18W 3Amp USB Wall Charger	Power Source	$15.99	Amazon.com
Amaitree USB C Cable	Power Cable	$9.28 for 3	Amazon
Adafrui lipo 3.7 3 cell battery pack	Battery	$25	Amazon.com
NaturePedic Organic Cotton/PLA pillow	Pillow	$79.00	Amazon

Future Plans

- Expand Compatibility: Allow app to be usable across both Android and iOS devices
- Improve Sleep Pattern Detection: Add a machine learning algorithm to accurately measure sleep patterns
- Improve Pillow Design: We can improve airflow within the pillow to avoid it from heating up and being uncomfortable.
- Acquire better hardware specs: We can improve accuracy from the sensors with more expensive hardware.

Successes and Failures so far

Successes:

- Able to get a large portion done early
  - Hardware
  - Software
- Good group communication
- Able to fix issues as they came up

Failures:

- Had to remove bluetooth mic
- Re evaluated the difficulty of a machine learning language
- Had to change some parts to accommodate for availability

Questions?

Slides?

- Goals of the Project
  - What is it supposed to do? **- David**
  - How does this benefit users? **- David**
  - Inspiration **- David**
- Software Design (multiple slides)
  - Block Diagram **- David**
  - Class Diagram **- David**
  - Java (Classes/Individually testable) **- Kevin**
  - Sleep Wellness Page **- David**
  - Alarms Page **- Kevin**
  - Audio Page **- Kevin**
  - ML algorithm **- David**
- Hardware Design (multiple slides)
  - Block Diagram - **jason**
  - Accelerometer - **jason**
  - Pressure Sensor - **jason**
  - Light Sensor **- Kevin**
  - Microphone - **chris**
  - **Speakers - jason**
  - Temp **- Kevin**
  - Power systems - **chris**
  - Microcontroller - **chris**
  - Bluetooth - **jason**
  - Schematics - **chris**
- Budgeting **- Kevin**
- Future Plans
  - Current Progress **- jason**
  - Prototyping - **chris**
  - Testing **- Jason**
  - App Development **- Kevin**
- Successes and Failures **- chris**
- Q&A **- chris**

Contents:

- project title, group number, project group members, and administrative introduction
- statement of project goals, objectives, specifications, and requirements
- overall project block diagram, or system design diagram
- discuss the project design approach and proposed implementation
- discuss and support significant design decisions and significant component decisions
- discuss project successes and difficulties for each project subsystem
- show the project budget and financing to date, and to the end of the project
- clear statement of current progress made by the group (% done by category)
- discuss the group's immediate plans for the successful completion of the project

Part Connections:

- Light Sensor
  - Model: GL5537
  - Reading Pin: 1 ADC
- Pressure Sensor
  - Model: PRESSURE SENSING RESISTOR
  - Reading Pin: 1 ADC
- Temp Sensor
  - Model: DHT11
  - Reading Pins: 1 GPI
  - Note: needs a pullup resistor on out pin
- Accelerometer
  - Model: ADXL345
  - Reading Pins: SDA/SDL
- Audio out
  - Model: MAX98357A (x2) -> speakers (x2)
  - Reading Pins: I2S Bus
  - Note: will require 3 lines (LRC,BCLK,DIN)
  - Note: SD pin open -> mono, SD pin 0.77V-1.44V -> Right, SD 1.44V-up -> left