

Study Spot Finder

Group 12 - Senior Design Spring 2020

Perla Del Castillo, CPE

Maria Ines Herbas Gutierrez, CPE

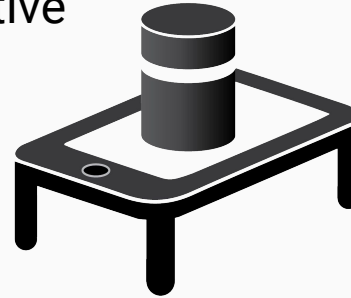
Andrew Kwong, CPE

Darwin Motato, EE

Study Spot Finder

Agenda

1. Introduction
2. Motivation
3. Specifications
4. Standards and Constraints
5. Block Diagram
6. Hardware
7. Software and Hardware Integration
8. Mobile Application
9. Challenges
10. Administrative



Introduction

- Study Spot Finder is a device that helps students locate available spaces on campus to study, either for individuals or groups.
- It implements an online reservation system through a mobile application.
- Users can choose a spot based on number of outlets, capacity, and others.



Motivation

Encourages organization



Reduces wasted time
finding available study
spots



Improves the current
reservation system



Improves students'
experience



Specifications

1	System Specification	Unit
1.1	The power supply provides appropriate voltage to the microprocessor and Wi-Fi module .	5 ± 0.5 Volts 3.3±0.05Volts
1.2	The device consumes minimal current when Wi-Fi is not required	< 130 mA
1.3	The device boots and begin transmitting data to Wi-Fi within a specified time period	7 seconds
1.4	The device changes its signal LED color to reflect its status within a specified time after a status change	6 seconds
1.5	The device checks for status updates at a periodic time interval	5 seconds
1.6	The mobile app reflects any changes to available spots within a specified time after an event trigger	5 seconds
1.7	Any changes to device settings by admin reflects in database within the following time period	5 seconds



Constraints and Standards

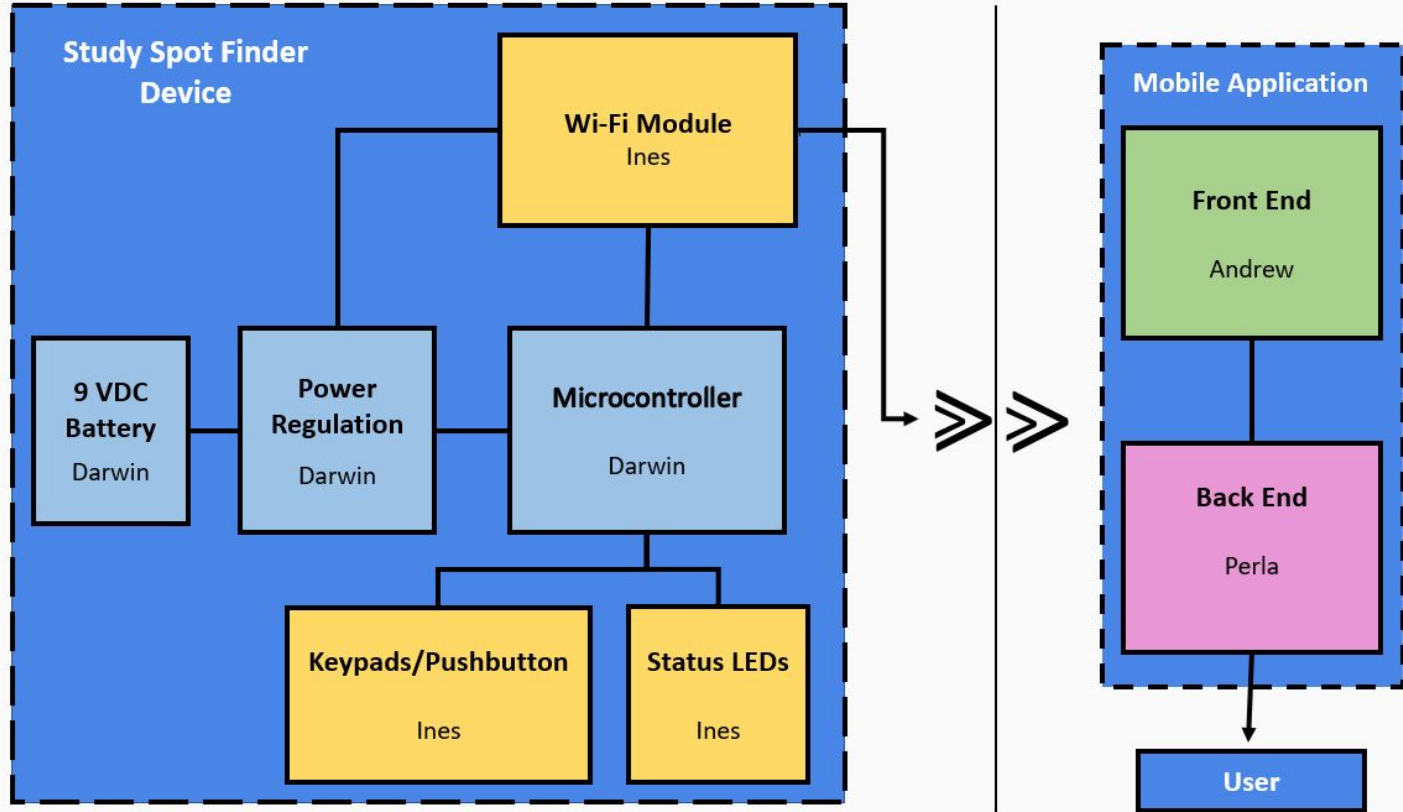
Constraints:

- **Time Constraints**
 - 2 Semesters, 1 for research and 1 for construction
- **Economic Constraints**
 - < \$100
- **Manufacturability constraints**
 - Lightweight, Durability, 3D Printing restrictions
- **Power Constraints**
 - Low power consumption

Standards:

- **Battery Standards**
 - ANSI C18.2M
- **PCB Standards**
 - IPC-221B
- **Wireless Standards**
 - IEEE 802.11

Block Diagram



Andrew ■ Darwin ■ Ines ■ Perla ■ Hardware Software



HARDWARE

Casing

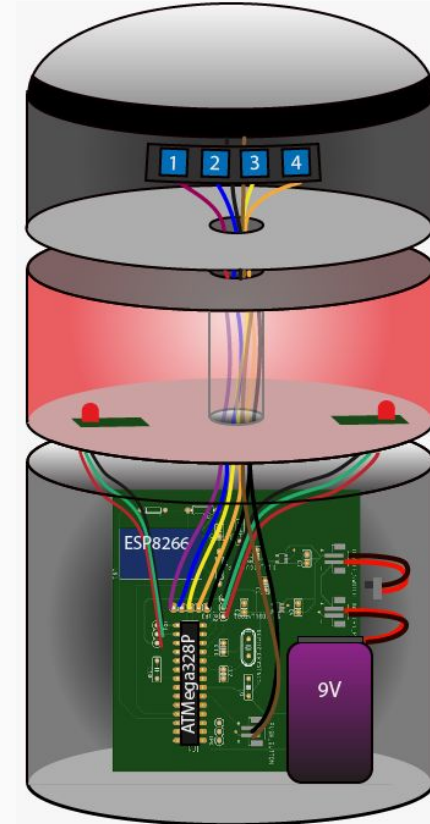
Shape: Cylinder

Material: PVC plastic

Dimensions: 10cm x 10cm x 20cm



Study Spot Finder
Concept design



WI-FI Module | Research Comparison

Decision Matrix	CC3220	ESP-8266	ESP12
Cost	\$59	\$6	\$10
Ease of Use	Too much documentation, Little Forum support	Forum support, some documentation	Forum support, some documentation
Average Current Use	34 mA	10 mA	15 mA
Firebase Communication	No	Yes	Yes
Supply Voltage	2.1-3.6	3.3-3.7	3.3-3.7
Flash Memory	1 MB	1MB	1MB

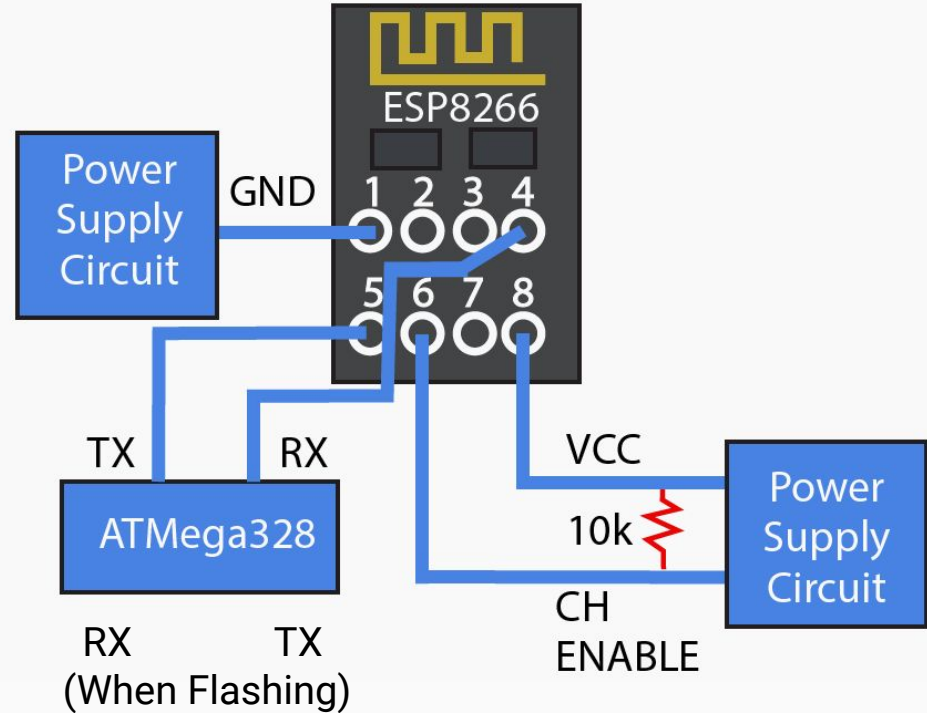
- Final Decision

ESP8266

By Expressif

Cost: \$5.99

- Compact, Low-Cost
- Serial UART Communication
- Support from Arduino Libraries for effective coding
- Better range compared to Bluetooth



Microcontroller | Selection

Decision Matrix	Raspberry Pi Zero W	ATMega328P	MSP430G2553	STM32F103C8T6	NUCLEO-L011K4
Cost	\$10	\$2.08	\$2.09	\$1.69	\$10.32
Ease of Use	Raspberry Forums	Arduino Library	Heavy Documentation	Some forums	Some Forums
Power Use	80 mA	16 mA	0.35 mA	1.19 mA	20 mA
Wi-Fi Enabled	Yes	No	No	No	No
Firestore Communication	No	Yes	No	No	No
Flash Mem	Expandable	48kB	128kB	128kB	16kB
Pins	40	14	47	38	38



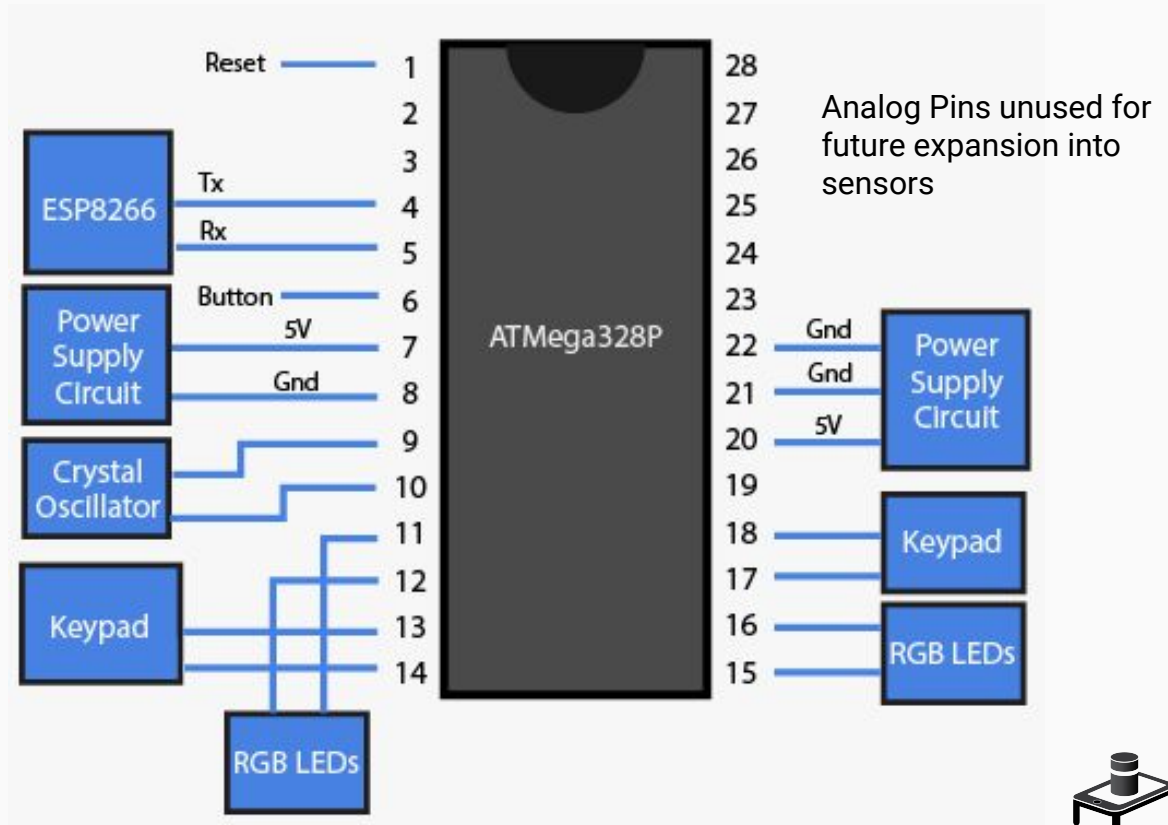
- Final Decision

ATmega328P

By Atmel

Cost: \$2.08

- Cheap and Energy Effective
- Easy communication with firebase library
- Simple flashing & debugging with development board



Serial Communication

- In order to send multiple values of information at once, a data packet was created for both the ATmega chip and the ESP Module.
- The keycode integer was sent using the *highByte* and *lowByte* command in the Arduino IDE
- Once received by the ESP Wi-Fi module, the high byte was shifted by 8 and an OR operation was performed with the low byte to obtain the full Keycode.

Device	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Arduino => ESP	Header	Keycode High Byte	Keycode Low Byte	Status	Footer
ESP => Arduino	Header	Valid	Status	Footer	

Voltage Regulation

9 V rechargeable Battery

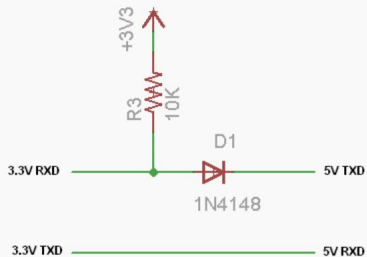
2 Voltage Regulators

- 3.3V (ESP8266)
- 5V (ATMega & Buttons)

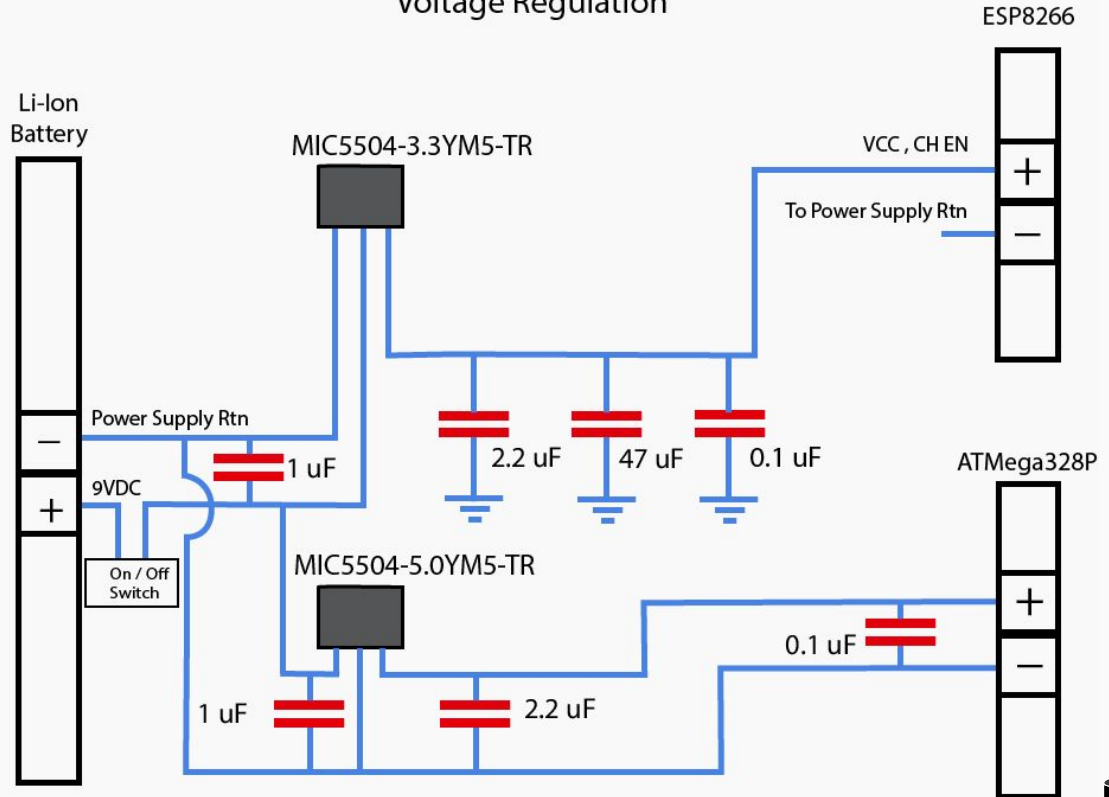
Decoupling Capacitors based on Datasheets

ESP8266 Has strict voltage fluctuation tolerances

Serial Communication also requires 5V to 3.3V Logic Conversion



Voltage Regulation



LED String Type	Operating Voltage (V)	Light Color	Additional Features
Flexible LED circuit	12V	RGB	Flexible, wide PCB
Full Color Flexible LED circuit	12V	RGB	DMX 512 compatible, flexible
Driverless RGB LED Strip Light	12V	RGB	Driverless, cuttable segments
Ring Lights WS2812B	5 V	Vary	Stability and efficient
CornholeRing Lights	4.5 V	Vary	N/A
Lumex	2.1V	R, G, B	Simple, cheap, accessible

PCB Design Overview

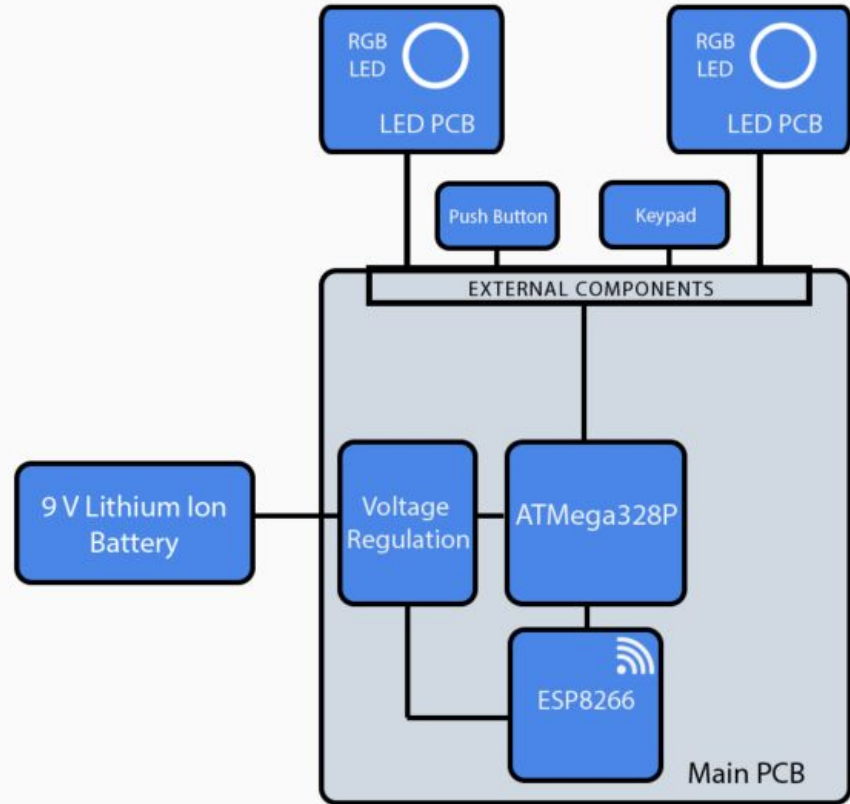
2 different PCB Boards:

Main PCB

- Processing
- power regulation
- Wiring to external components

LED PCB

- Holds an RGB LED
- Separate for convenient placement





- Final Decision

5mm RGB LED

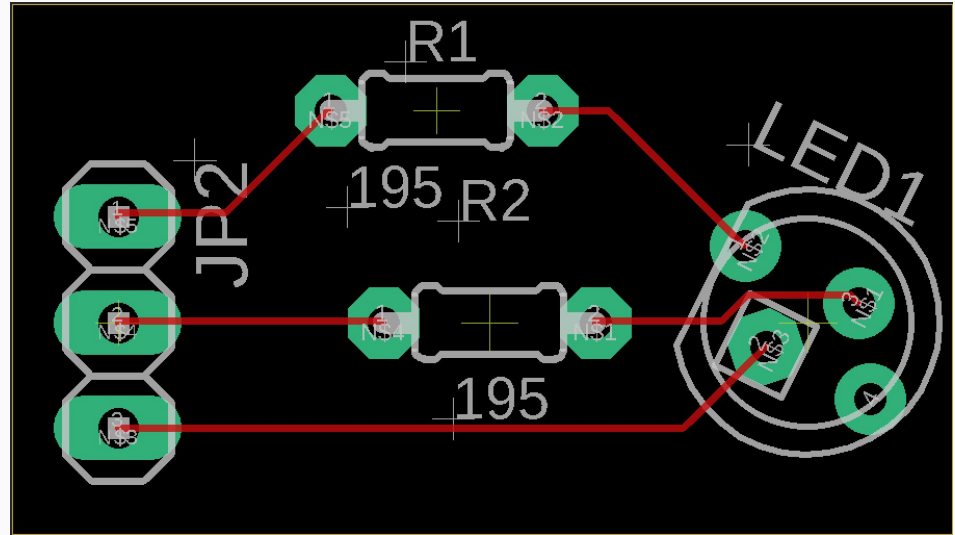
By Jameco

Cost: \$0.39

Reservation

Available

Awaiting Confirmation



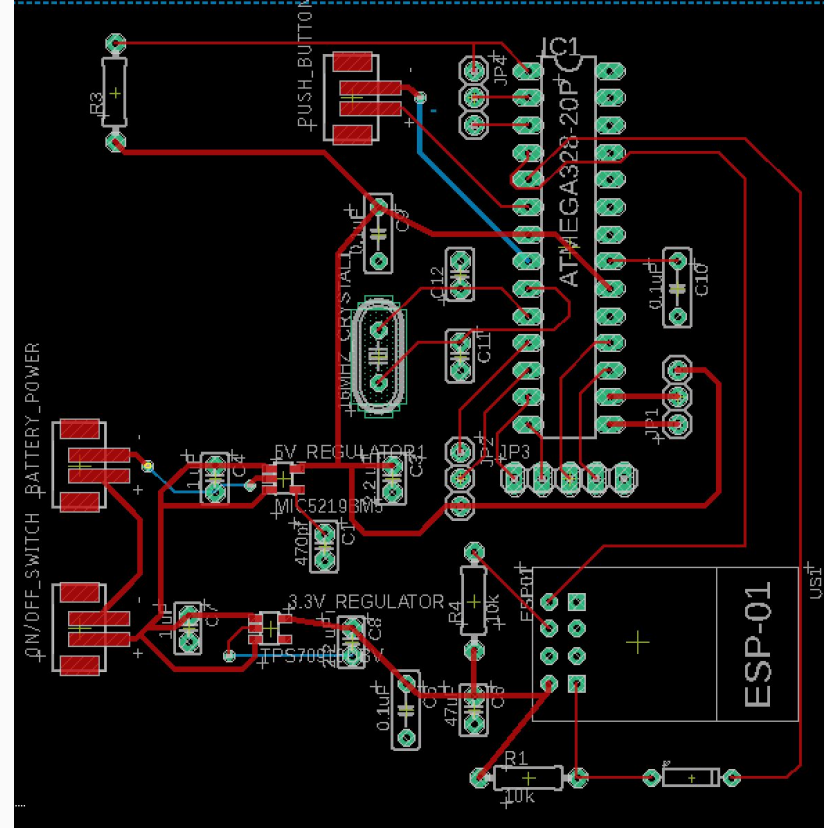
- RGB LED
- Forward Current required => 20mA
- Blue pin will not be used, only Red and Green.



Main PCB

Components:

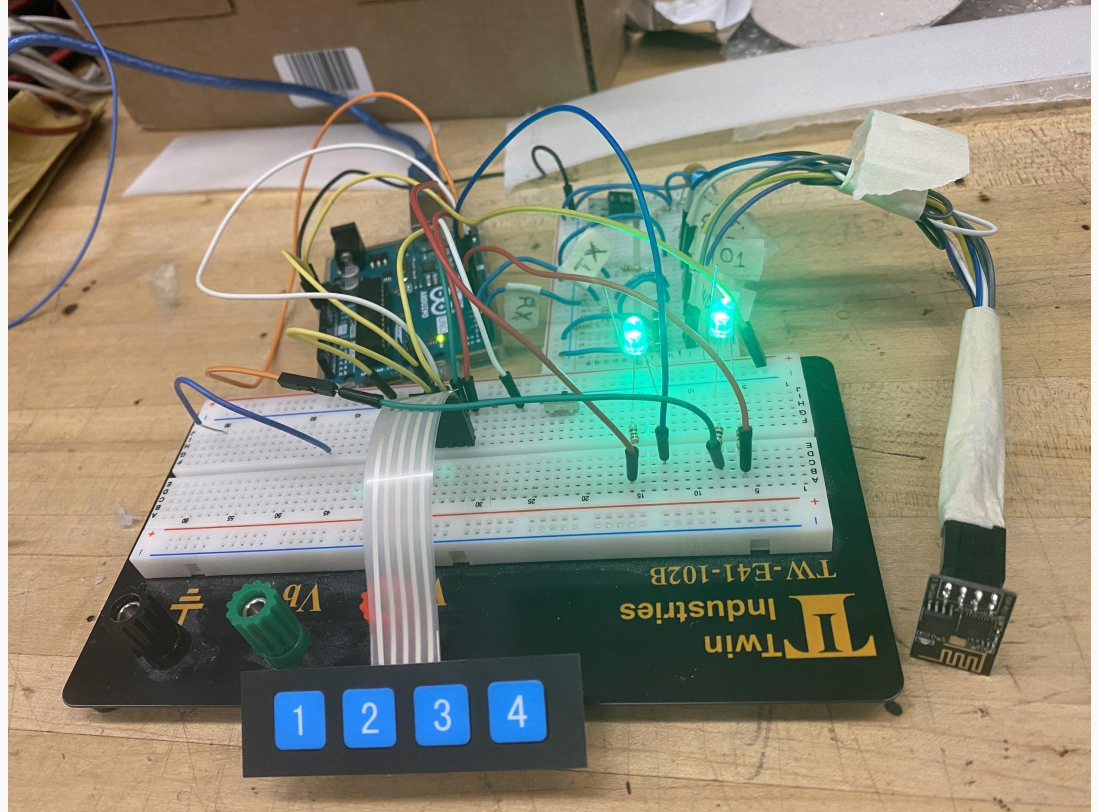
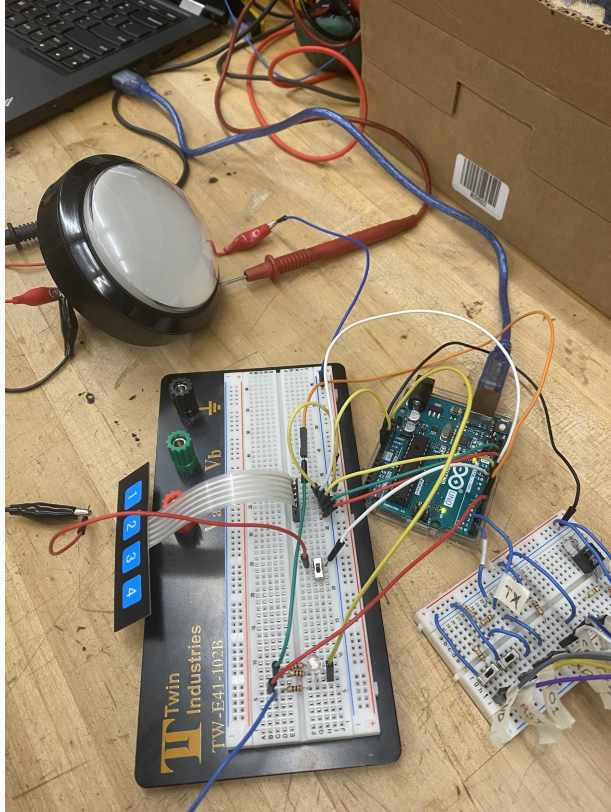
- 3.3V voltage regulator
- 5V voltage regulator
- Capacitors
- Resistors
- ATmega 328P
- ESP 3266
- Crystal Oscillator
- Keypad
- Push button
- Connections to LED PCB



Power Consumption

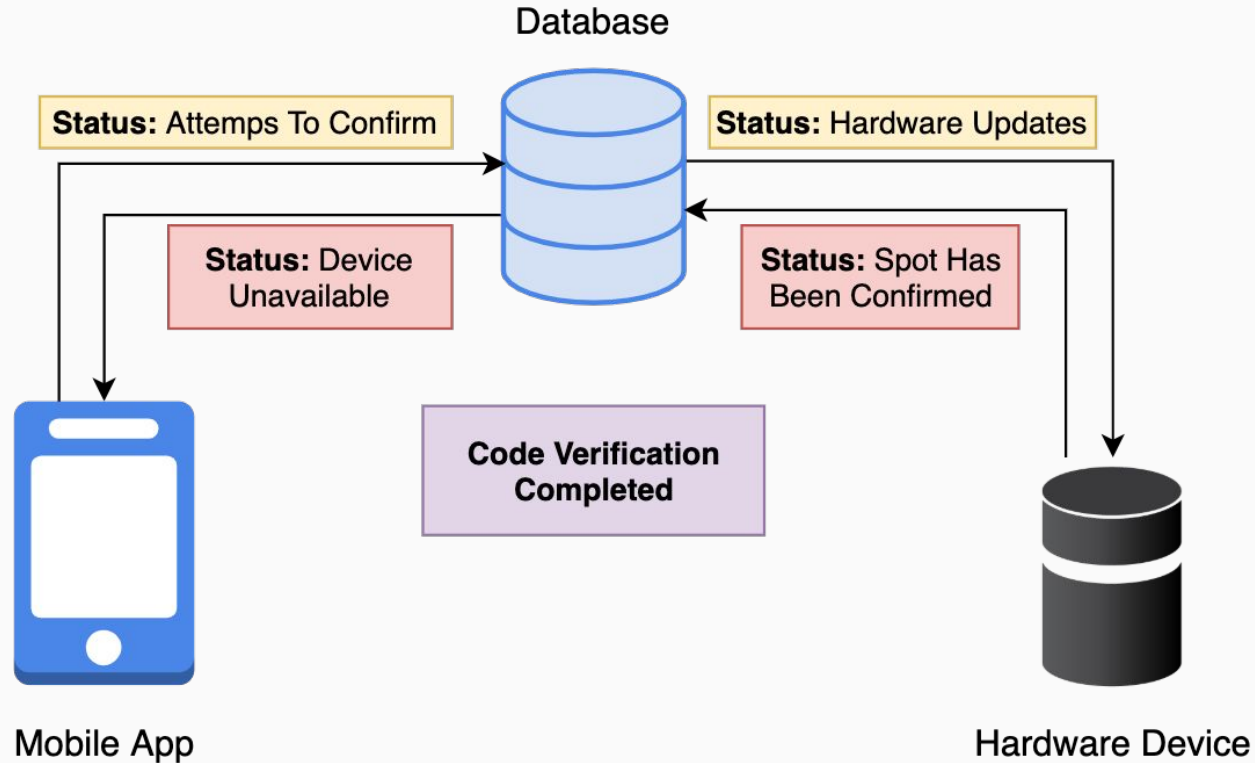
Power Modes	Total Current Consumption	Rechargeable 9V 1200 mAh Time
Regular	40 mA	34.3 Hours
Transferring Data (~1mS)	450 mA	2.6 Hours





INTEGRATION

Software and Hardware Integration



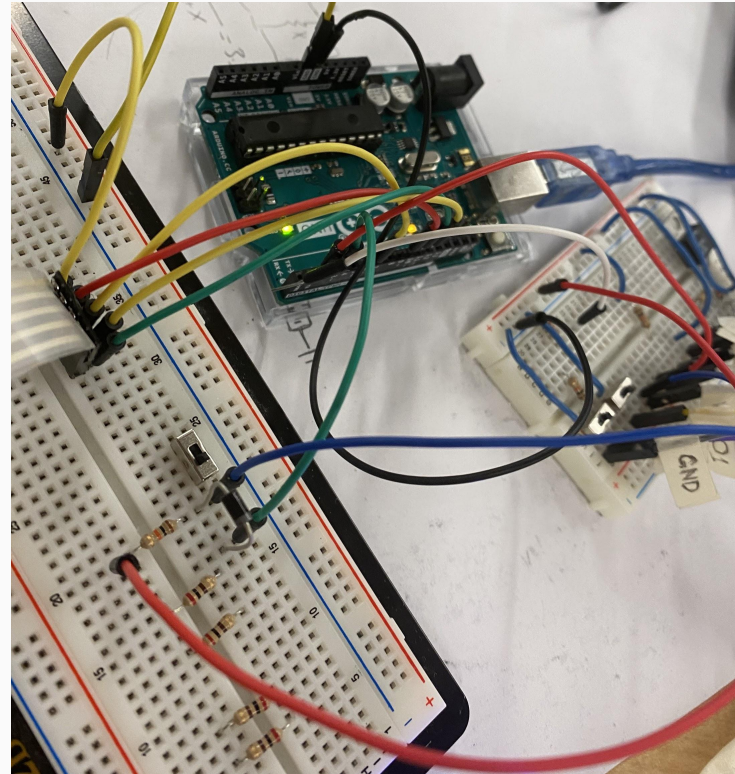
Integration

- Every hardware device has a unique ID that will allow the software to interact with multiple devices simultaneously
- Hardware device will be periodically checking for new information as well as pushing status updates
- Software will be able to bring back information from this real time database to the user



Arduino-Firebase

- ESP8266 requires a flashing software & Arduino IDE in order to set up and flash the Wi-Fi module before running.
- ATmega talks to the ESP8266 through UART Serial Communication at 9600 Baud using SoftwareSerial.h library.
- Firebase-Arduino library is used to send and manipulate data in a realtime Firebase database through Wi-Fi.



Project

Study Spots

0001

Name: "Table 2"
DeviceID: "0001"
BuildingName: "Harris"
Floor: "2nd Floor"
SpotType: "Group"
Capacity: "Up to 4"
Outlets: "2"
DesktopAvail: "No"
Status: " 1"
Valid: " 0"
KeyCode: "0 "

- Firebase will be handling the data with a real time database
- Admin will set up the study spot/device from the admin application and this will appear in the database
- Hardware device has access to 3 fields:
 - Status
 - Valid
 - KeyCode
- Admin has the ability to change other fields of the device



MOBILE APP

Mobile App | Development Selection

Decision Matrix	iOS (Native)	Android (Native)	React Native (Cross-Platform)
Programming Language	Swift	Java	JavaScript
Performance	Fast	Medium	Fast
Cost	99\$/year	\$25	Depends
Development	Fast	Slow	Fast
Flexibility	Limited	Open-Source	Open-Source
Testing	Limited	Extensive	Moderate

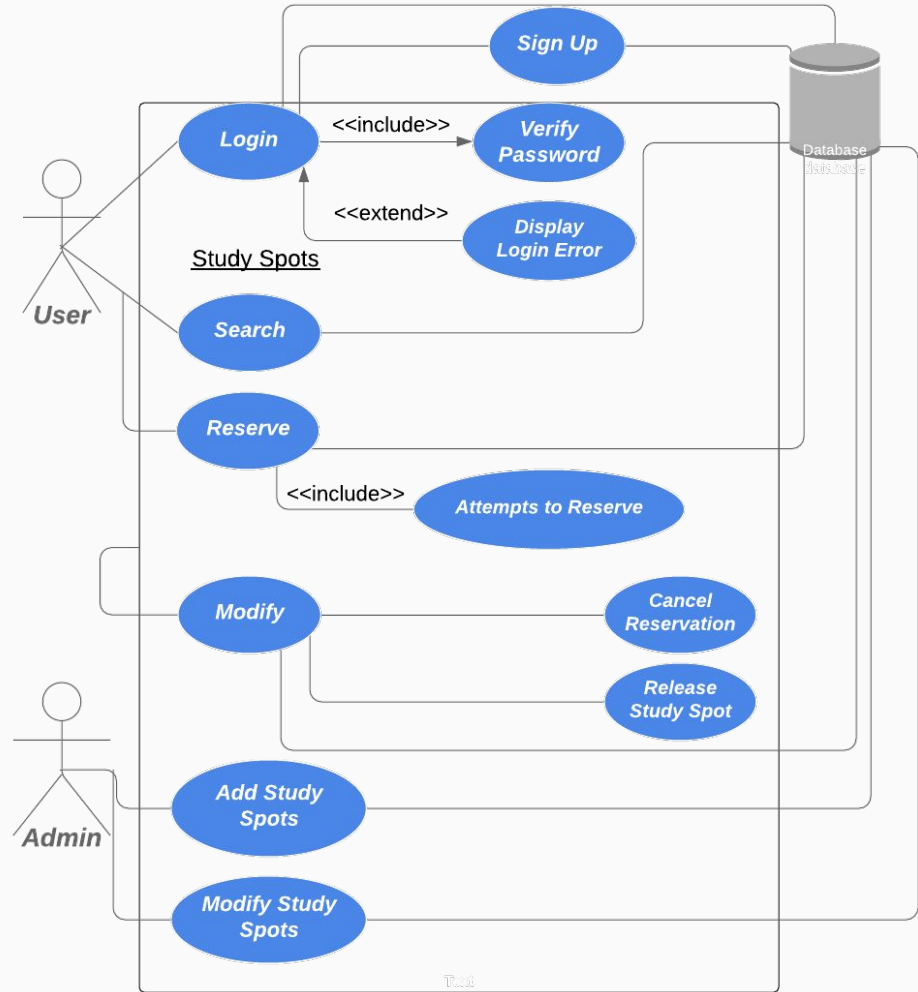


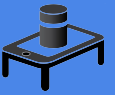
Mobile App | Development Selection

Decision matrix	AWS	Azure	Firebase
Cost	Free 5 GB stored/month, 15 GB served/month	Free 1 GB Disk Space Up to 10 apps	Free Hosting Up to 1 GB 5 GB storage
Ease of Use	Easy to get started	Requires some prior knowledge	Easy to get started
Performance	Medium	Fast	Fast
Support	Extensive	Extensive	Moderate
Updates and Maintenance	Limited	Moderate	Extensive



Use Case Diagram



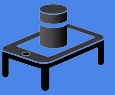


Reserve A Study Spot

<p>Table 3</p> <p>Library 2nd Floor</p>	<p>Table 4</p> <p>Student Union 1st Floor</p>
<p>Spot 2</p> <p>Library 3rd Floor</p>	<p>Table 6</p> <p>Eng 2 Atrium 1st Floor</p>
<p>Table 34</p> <p>Library 2nd Floor</p>	<p>Spot 1</p> <p>CSB 1 1st Floor</p>
<p>Spot 2</p> <p>Eng 2 Atrium 1st Floor</p>	<p>Study 1</p> <p>Tech Commons 1st Floor</p>


Reserve A Study Spot

<p>Table 3</p>	<p>Table 4</p>
<p>Table 6 ✕</p> <p>Location Eng 2 Atrium 1st Floor</p> <p>Type Group </p> <p>Capacity Up to 4</p> <p>Outlets 2</p> <p>Key Code 0</p> <p>Reserve</p> <p>Release Spot</p>	
<p>Eng 2 Atrium 1st Floor</p>	<p>Tech Commons 1st Floor</p>




List Of Devices

Table 3 Library 2nd Floor	Table 4 Student Union 2nd Floor
Spot 2 Library 3rd Floor	Table 6 Eng 2 Atrium 1st Floor
Table 34 Library 2nd Floor	Spot 1 CSB 1 1st Floor



Register a New Device



Name of Device

Device ID

Building Name

Floor Number

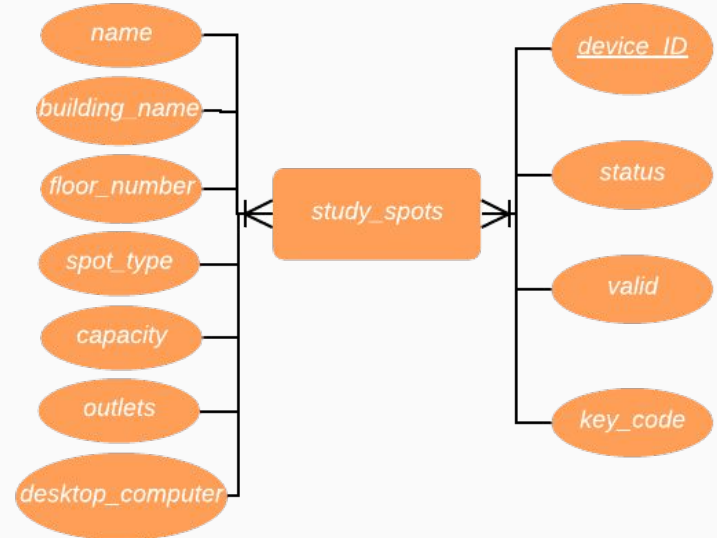
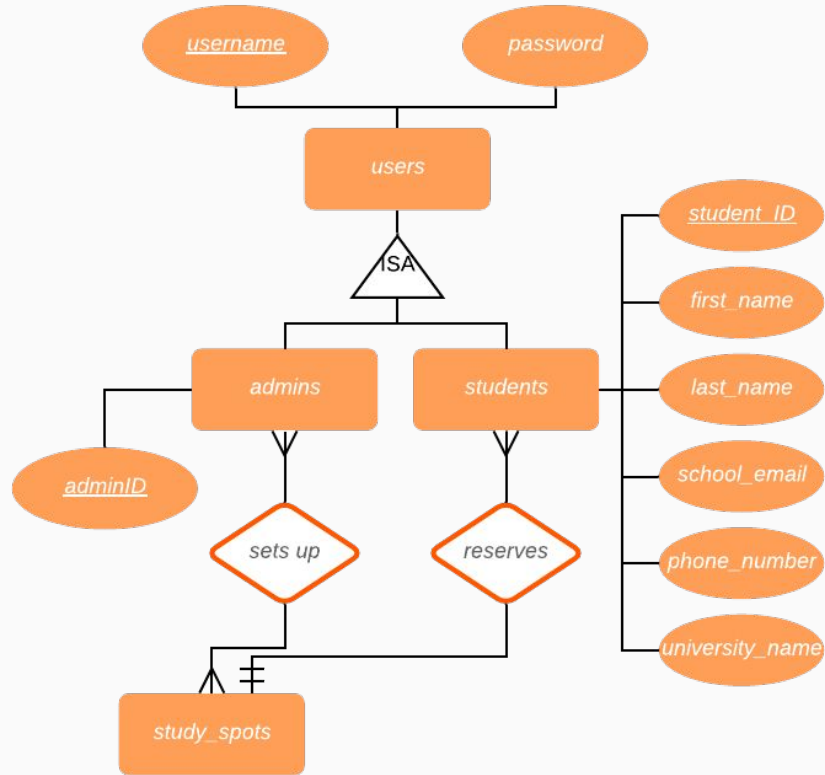
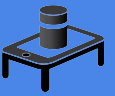
Spot Type

Capacity

Number of Outlets

Desktop Available

Entity Relationship Diagram



CHALLENGES

Hardware Challenges

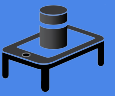
Challenge	Solutions
Esp Resetting	Used more decoupling capacitors to stabilize voltage spikes
Serial Communication between ESP and Arduino	Serial Monitor debugging, Firebase Librar's Footprint, proper logic voltage conversion
Material supply	Ordered materials in advance, and ordered more than necessary
PCB design Errors	Re-orders and careful revisions of PCB designs
Manufacturing processes	Switch process to an available one
Social Distancing Order	Communicated via online platforms, and maintained healthy environment
Time	Scheduling to account for error and unforeseen circumstances

Software Challenges

Challenge	Solutions
Application Emulation	Use Expo and XCode as main sources of emulation, so we can be consistent across the board
Implementing APIs between front end and back end	Use Google Firebase documentation to understand how to carry out server operations
Design and Feature Priority	Identify the most critical features outlined in the requirements and implement them
Debugging Errors	Use Google and friends
Time constraints	Use a requirement tracking solution (Jira)
Creating and implementing unique ID values	Use Google Firebase documentation in order to implement a different approach of adding new devices with this unique ID value

ADMINISTRATIVE

Cost



Level	Description	Manufacturer	Part No.	Quantity	Unit Cost	Total Cost
1	Microcontroller	Arduino	ATmega328P	1	\$2.50	\$2.50
1	Wifi Module	Espressif	WRL-13678	1	\$6.95	\$6.95
1	Voltage Regulator	Microchip	MIC5219-TR	2	\$0.89	\$1.79
1	Keypad	Adafruit	4861332	1	\$2.95	\$2.95
2	PCB	JLC PCB	-	1	\$2.0	\$2.0
2	LED	Jameco Value Pro	2228957	2	\$0.39	\$0.78
2	PVC	MH Build Series	MMUULTCG	1	\$12	\$12.00
1	Connectors	JST PH 2 pin	VUPN924	2	\$2.9	\$5.8
1	Button	Jiu Man		1	\$9.99	\$9.99
1	Switch	Judco	J-188A-1	1	\$1.85	\$1.85
1	Battery	Battery Junction	RAYOVAC-R9VL	1	\$8.78	\$8.78
Total						\$55.39

Work Distribution

Team Member	Front-End	Back-End	Software Integration	PCB	LED	Hardware Integration	Wi-Fi Communication
Andrew	Primary	Secondary	Primary				
Perla	Secondary	Primary	Primary				
Darwin				Primary	Secondary	Primary	Secondary
Ines				Secondary	Primary	Secondary	Primary



Thank You!