a) Monday, June 12
b) Finish my minty boost starter project.
c) Finished starter project, trained for soldering. I finished soldering all the components to the pcb(resistors, diodes, capacitors, etc.) I'm also able to solder now that I finished the soldering training.
d) I learned that when you're testing the continuity for a resistor, the multimeter won't beep unless the resistor is under 20 ohms. The multimeter beeps when it detects a closed circuit but with a resistor. It doesn't detect the voltage. When power is going through the leads it will beep if the power will come back into the multimeter. Because resistors resist voltage, the there won't be the same amount of voltage coming out of the resistors back into the multimeter. Because the voltage is decreased, the multimeter won't detect a closed circuit.
e) Finish the documentation for starter project and begin my vacuum robot. I think I'm supposed to document the starter project. If not, I'm just going to do what Erik tells me to do.

-

a) Tuesday, June 13
b) Finish documenting my minty boost starter project. I also what to start my vacuum robot.
c) I Finished up my research on all the components. I figured out what inductors do. I also finished up my video and documentation on the website. I talked about the components and what they do, and some of the struggles I went through with the project.
d) I learned about USB plugs. USB stands for Universal Serial Bus and that port 1 supplies 5 volts, ports 2 and 3 are used for the in/ out data ports, and port 4 is the ground.
e) I plan to start my vacuum robot. I'm still wait for my 3d printed parts but I can still solder up the wires and do more research on the parts and what they do.

The minty boost is a small portable charger. It can charge many mobile devices like your phone. As long as you have a usb cable you can charge your device. Let's start with how it works: The power source is made up of two AA batteries. Each battery provides about 1.5volts. When you couple both batteries in series, you will get the sum of 3 volts. That 3 volts will go into the PCB(Printed Circuit Board, soldered with all sorts of components. The battery leads will go into a pair of electrolytic and ceramic capacitors. There is one ceramic and one electrolytic capacitor for each positive and negative power leads. Their purpose is to stabilize the voltage from the battery going into the IC chip. After the capacitors, the current will flow into a inductor. The inductor will resist change in current flowing the 5 volt boost converter chip. The purpose of the chip is to convert the 3 volts from the two AA batteries into 5 volts. The only downside by increasing voltage is that the current will decrease to about 0.5 amps. This is twice as slow as the normal wall chargers but this one is probably. The 5 volts will then go through a diode, which will only allow current to flow in one direction. In this case, current will flow right into pin 1 of the USB(Universal Serial Bus) plug. Pin 1 will provide 5 volts for your device. Pin 4 of the USB will be the ground of the IC chip. There are also 5 volts going into a pair of 75k and 49.9k resistors.

This will later go into the the two data pins, pins 2 and 3. The purpose to the data pins is to tell the mobile device that it's being charged. In the end, you get a small portable usb charger. The overall build of the minty boost is fairly easy. If you just learned to solder, this is a great first project. I had trouble with figuring out why I'm not getting continuity when testing a resistor using a multimeter. I later found out that multimeters won't beep unless the resistor is under 20 ohms. I also had trouble figuring out what inductors do. I know now that inductors resist change in current.

-

a) Wednesday, June 14
b) 1. Start my vacuum robot. I just want to get as many things done today such as my infrared sensors.
c) I was able to get most of my parts. All I have to do is to get my 3d printed parts and some pushbuttons, etc. I also was able to get the only 1k resistor in the pile of a million other resistors. I was also able to get all the screws and nuts. I wasn't able to start my actual build of my robot but I got most of the parts.
d) I learned how a dpst switch works. It stands for a dual pole single throw switch. There are a total of 4 pin and you are able to turn two different circuits on and off just by using one switch.
e) I plan to work on creating the schematic for my robot because I can't really start my project without the 3d parts. I can also work on the code for the robot if I have the time.

-

a) Thursday, June 15
b) 1. Start creating the schematic 2. Continue thinking of the overall build process.
c) I was able to finish my schematic. At first, it was very messy, then I was able to bend the wires so it looks a lot more cleaner. I just need to double check if it's correct. I also need to get it checked with the instructors.
d) I learned how to create a schematic using fritzing. I was able to make my own electronic such as the IRF520 MOSFET Driver Module. It's very simple, just search up the component then drag it into the workspace.
e) I plan to get my schematic checked and work on 3d printing my vacuum robot. Because the 3d workspace is smaller than the actual robot. This means I will have to split the cad file in halves and glue the pieces together in the end.

-

a) Friday, June 16
b) 1. Start 3d printing all my parts.
c) I was able to test all the electronics. I was able to reverse my motors and adjust the speed of them. My LEDs and push buttons work as well.

d) I learned how to control/program dc motors. I was able to connect it using the h-bridge motor driver and I can change the speed of the motors using PWM(Pulse Width Modulation). I can make it go at full speed 255 or stop at 0. I also learned how to change the direction of the motors. There are two pins for each motor. If you set one High and the other at Low, then it will spin in one direction. If you switch those two values, High to Low and Low to High then it will spin in the opposite direction.
e) I plan to finally start 3d printing my parts, I will have to slice the robot body in half in order for it to fit on to the print bed. After that, I will be able to connect everything together.

Erik and Andrew 6/16 1-on-1 meeting
Starter Project: it works, awesome!
What did you struggle with?: the research took a great deal of time and work.
What was the learning experience?: Researching take a lot of work to do it effectively
Intensive Project:
What do you anticipate being a struggle?: Building the actual robot will be a struggle. The programming is going to be difficult because I don't have much experience.
What is the next step?: 3D print parts right now
Program
Don't like: not getting a direct answer to a question
Like: looking at the progress in the first week

-

a) Monday, June 19
b) 1. Start 3d printing all my parts 2. Maybe start building.
c) I found someone that can 3d print my parts in pla. I had to contact several people to see if they can print some of my bigger parts. I was also able to get in contact will the original builder and we are planning to make some cool modifications. I also was able to design a mount for my IR sensors.
d) I learned how to make some of my own 3d printed parts using CAD software. It stands for Computer Aided Design. It allows the user to create 3d files. I also learned how to use the makerbot slicer software to turn my .stl files into g code. G code is the code that 3d printers and cnc machines understand. There is a line of code for each layer of filament.
e) I plan to order my 3d printed parts and start prototyping.

-

a) Tuesday, June 20
b) 1. Order 3d printed parts 2. Get trained for all the tools.

c) I finally got my 3d printed parts ordered. I was also able to find the correct screw size for my ball caster. I was able to hook up my two infrared sensors to my arduino. I was also able to program it as well. I started creating a new program that will stop the motors if the sensors detect an obstruction.

d) I learned how to use sketchup to find out how thick my robot chassis is. It's very simple, all you have to do is use the measuring tool a drag it along the part you're measuring. Something more interesting that I learned was programming my IR sensor. I had to plug in the wires to the 5v, Gnd, and A0. A0 is the first analog pin. Then you can just use the analogRead function, to read what info the sensor is telling you.

e) I plan to continue programing and building my robot.

-

a) Wednesday, June 21

b) 1. Program my sensors and motors together 2. Hook up my motors and wheels together.

c) Today, I was able to finish my program for my motors and sensors so that the motors will stop moving when the infrared sensors detect an obstacle. I was also able to connect the wheels to the motors. It was a very hard friction fit. I had to sort of jam it in. I was also able to get my fan working and my IRF520 Mosfet Driver Module working.

d) I learned how to use the if and else conditional statements. You write if then (). Whatever is in the parenthesis is the condition qualifier then what's in the {} is the code to perform if the condition qualifier is true. If the if condition is false then it will run the code in the else if condition qualifier . If that is also false, then it will run the code in the else.

e) I plan to connect my IR sensor and motors to the 3d printed parts. Then I will finish documenting create my video.

-

a) Thursday, June 22

b) 1. Adding more to my code 2. Finish documentation

c) Today I was able to add to my IR and motor code. I made more if else conditional statements so that the motors will move a certain way if the sensors detect something. When both sensors detect an object, then the robot will move backwards then turn left. When only the right sensor detects an object the robot will move backwards then turn right. When only the left sensor detects an object the robot will turn right. When both sensors don't detect an object then the robot will move forwards.

d) I learned that you can just use solid core wire and strip the two ends as a replacement of the male to male jumper wires.

e) I plan to get my 3d printed parts and attach the electronics.

-

a) Friday, June 23
b) 1. Work on the code 2) Attach some of the motors and sensors.
c) Today I was able to work on the code a little. I added some pwm which will allow me to slow down the motors without using the enable pins. The enable pins take up too many pins on the arduino. Now I have space to add more accessories. I was also able to get my 3d printed parts so I started to attach my motors and sensors.
d) I learned how to use pwm to slow down the motors. You have values from 0-255, not moving-moving at full speed.
e) I plan to finish my first milestone and finish debugging the code.

-

a) Monday, June 26
b) 1. Finish attaching the ball caster. 2. Finish debugging my code.
c) Today I was able to get my code working! I realized that my motors were spinning the opposite direction than I intended it to do. I was also trained to use a power drill.
d) I learned how to debug my code. I had to check line by line to find out why my robot wasn't doing what I wanted to do.
e) I plan to finish my first milestone.

-

a) Tuesday, June 27
b) 1. Finish my 1st milestone.
c) Today I was able to finally finish my first milestone. I was able to drill out the bigger holes for my h-bridge to fit with the M3 screws. I was also able to start soldering and connecting the buttons to the bumper.
d) I learned that my 3d printed parts were too small for my M3 screws and buttons. So I had to spend a long time filing and drilling.
e) I plan to finish connecting the bumper and solder up my wires so it looks cleaner.

-

a) Wednesday, June 28
b) 1. Finish mounting the bumper 2. Finish soldering all the wires.
c) Today I was able to mount my buttons for the bumper and CA it down. I was also able to mount the wall sensors. But I'm still working on the floor sensors.

d) I learned how to use an x-acto blade and a hand saw. I used them for cutting my chassis .
e) I plan to finish mounting all my sensors together.

Today I completed my first milestone which was to build and program my IR sensors and motors. I connected the l298 dual H-bridge motor drive to the arduino, the 2 motors, battery and the 2 IR(Infrared) sensors. The 1300mAh 3s Lipo(Lithium Polymer) battery will be used to power all the electronics. At full charge, the battery will provide up to 12.6 volts into the motor driver. The 12.6 volts will be used to power up the two metal gear motors. The power from the Lipo battery will also go into a 5 volt step down voltage regulator that will decrease the voltage down to 5 volts. The 5 volts will be used for powering up the two IR sensors and the arduino. Then there is the ground that will be connected to all the electronics. Now let's look into the pins on the arduino. Analog pin 0 and 1 will be connected into the two IR sensors. I am also using pwm pins 3, 5, 6, and 9 for the motors. I am using PWM(Pulse Width Modulation) so I can control the speed of the motors. The H-bridge has 2 inputs for each motor. This allows you to change the direction of the motor. If you're confused, you can check out the schematic below. So how does the arduino know what it needs to do? It's in the code (link). When both sensors detect an object, then the robot will move backwards then turn left. When only the right sensor detects an object the robot will move backwards then turn right. When only the left sensor detects an object the robot will turn right. When both sensors don't detect an object then the robot will move forwards. This code will help my robot avoid walls and other obstacles. But had some trouble with coding the arduino. Once I finished coding, I found out that my robot wasn't moving in the correct direction. I thought it was my sensors that wasn't working. In end, it was the code. I had to debug it for hours to later find out that my motors were traveling the opposite direction then what I expected. When the robot was supposed to go right, it went forwards. So after fixing the motor directions, it worked!

I will be building a vacuum robot. This is a fun DIY project to work over the summer. It's an autonomous robot that will avoid walls and drop-offs. The robot will also help you understand how the arduino works. This is a super useful robot if you don't like vacuuming the floor. Just lay it down and let it clean!

-

a) Thursday, June 29
b) 1. Finish mounting all the sensors 2. I also want to solder up the 5v and Gnd wires together as well.
c) Today I was able to finish mounting the front, side, and bottom sensors for my vacuum robot. I also glued my button on accident so I have to fix that. I was also able to solder up the wires too.
d) I learned that my container filter doesn't fit very well. So I have to do more filing… Yay…
e) I plan to fix up the button and fix the container.

----

a) Friday, June 30
b) 1. Fix the button 2. Fix the container. 3. Finish up the other steps of the build.
c) Today I was able to Finish soldering male pins to all the sensors and buttons. I was also able to make the wires look neater. I also attached a switch so I can turn the robot on and off using a switch. I made a plug for my battery as well. This allows me to keep the xt-60 connector and still able to connect the power to the other electronics.
d) I learned that you should always use hot glue for your wires. They serve as an insulator and prevent your wires from moving around.
e) I plan to continuing soldering up my electronics.

----

a) Monday, July 3
b) 1. Test the sensors 2. Program
c) Today I was able to finish up my soldering and making the extensions for the male to female connectors. I was also able to connect most of the electronics such as the motor inputs and the IR sensors. I'm still waiting for the longer screws for the fan motor and the zip ties so I can bundle up all the wires. I was also able to work on another program to make it simpler and still function perfectly. I just deleted the line of code that created the variable for the values of the IR sensors. Then I just added it into the void();
d) Today I learned that color coding your wires (ex: yellow, orange, etc.) helps a lot with the organization of your parts so you prevent connecting them to the wrong ports.
e) I would like to continue programming documentation.

----

a) Wednesday, July 5
b) 1. Program
c) Today I was able to work on my program so that my fan motor will be able to move. When I finished that, I realized that I ran out of analog pins of my arduino uno. I was using all the arduino analog pins for the 6 sharp distance IR sensors. They provide the arduino with analog inputs so you can find out how far aways the sensor is from an obstacle. But I needed an analog pin for my voltage divider so I can measure the voltage of the lipo battery. After A couple of hours I created a great idea! I can use the sensors in the digital pins. I couldn't use the wall sensors or the ground sensors because I need to be very precise with those. But the front sensors can be digital because they either detect a wall or not. Then I started to make a voltage divider.

d) Today I learned how to use Tin snips. There fairly easy to use. It’s just a big scissor!

e) I would like to finish my voltage divider.

-

a) Thursday, July 6

b) 1. Finish calibrating my voltage divider.

c) Today I was able to finish up the code for the voltage divider. With the help of Erik, I was able to calibrate my voltage divider using the potentiometer. I first calculated the accurate voltage of the battery using a voltmeter. Then created a code so the arduino will send the voltage it’s receiving to the serial monitor. This allowed me to change the resistance from the potentiometer so I can get the correct voltage reading.

d) Today I learned that using println(); in your code is great for debugging code. Because I have so many conditional statements, I don’t know if they’re true. When using the serial monitor, the arduino will tell me what conditional statements are true.

e) I would like to finish programming and documentation for my second milestone

----
-

a) Friday, July 7

b) 1. Finish programing and documenting for my second milestone

c) Today I was able to finish my second milestone! I finished the code and soldering up the voltage divider to the battery. The new program has a conditional statement turning the fan motor on if the battery is full and turn it off if the battery is low. I finished up adding all the comments to help anyone else that needs help understanding the code. I also finished up the documentation for the second milestone. Next week I will finish up the video and website. Then I can begin programming the other sensors.

d) Today I learned about static int’s. They are just like normal in’s but static. For example, If you use them in the loop, the arduino will create the integer during its first loop, then ignore it during the second run. This is perfect for my conditional statements for my wall sensors. I need the minDistance to increase after each sweep so it’s more accurate. So I will use the static int in the loop and increase it my 100 every time the robot hits the end of the wall for wall tracking.

e) I want to finish the video and website for my second milestone. Begin my third milestone.

7/7/17

Project Status: second milestone functions! Filming video monday, schematic and code commenting needs to happen first.

Third Milestone: program and plug switch in - finish adding sensor functionality.

Looks awesome!

Problems: not at the moment

Today I accomplished my second milestone. I added a couple of new electronics to my vacuum robot such as 4 more IR (Infrared) sensors, a bumper, a fan, and a voltage divider. My milestone was more focused on the fan and voltage divider. I still need to program the wall, ground, and bumper sensors for my third milestone. The fan will be used for sucking in the dirt and dust on the floor. It's connected into a Mosfet Driver Module so I can turn it on and off. I also added a removable container to hold in the dirt it sucks. The container also has a filter, similar to the ones in vacuum bags. All the dirt and dust will be caught by the filter and the air will escape through the back. The main purpose of the voltage divider is to decrease the voltage coming into the arduino. Because my Lipo battery will provide up to 12.6 volts, it will damage the arduino due to the large amount of voltage. The decreased voltage will go into one an analog pin. The arduino will then read the voltage coming from voltage divider and then calculate the real voltage coming from the battery using a formula. I also have a conditional statement turning the fan motor on if the battery is full and turn it off if the battery is low. It's important to measure the voltage of the battery so you don't damage it. If the cell voltage of any Lipo (Lithium Polymer) battery goes below 3 volts, then it will be destroyed. This is called over discharging. When you over discharge a battery, it will start to puff like a balloon and when it pops, it becomes a huge fire ball of flames and smoke. Obviously, I don't want this to happen to my battery so that's why I made a voltage divider consisting of a 1k ohm resistor and a 2k ohm potentiometer. The potentiometer allows me to calibrate the voltage divider so the arduino gets an accurate reading. To do this, you simply calculate the actual voltage of your battery using a voltmeter. Then write a simple code so you can see the amount of voltage coming from the arduino using the serial monitor. I also had a huge amount of trouble with the IR sensors. After hooking up all the sensors to the arduino, I realized that I ran out of analog pins. I only have 6 analog pins and have 6 IR sensors and a voltage divider. So after a couple of hours on the internet I came up with a great idea! I could use the front IR sensors in the digital pins. The problem with using digital pins is that the only values you get from the arduino is HIGH (5 volts) or LOW (0 volts), the IR sensor either detects something or doesn't. The main reason I used the analog pins for the sensors is because they are distance sensors. You can measure the distance from the sensor and the wall, this is an analog signal. But because the front sensors don't need to calculate the distance like the wall and ground sensors so it would work perfectly. This would free up 2 more analog pins used for the voltage divider. I had trouble programming it because conditional statements are different due to using the arduino pins. Here is my new code for the vacuum. I added the ability to control the fan motor, measure the voltage of my battery, and avoid obstacles using the IR sensors connected to the digital pins.

----

a) Monday, July 10
b) 1. Finish second milestone 2. Program
c) Today I was able to finish my second milestone! Before starting the video, I had some bad problems. My robot wasn't working. So I had to do some changes in the code in

order for that to work. Then I filmed the video. After the video, I began programming the wall sensors. I still can't get them to work…

d) Today I learned that it's important to properly prepare for the video. I didn't' prepare as much as I used to, so It didn't turn out as well as I wanted.

e) Finish programming the wall sensors.

----

a) Tuesday, July 11

b) 1. Finish programming the wall sensors.

c) Today I was able to create code for the wall sensors and ground sensors. Both of them didn't work. I still have to debug all the code by tomorrow. So I'm very stressed out. I started creating my status lights for the robot. I have a green and a red led. The red light will blink for 5 seconds letting the user know that you need to wait. Then it will turn green letting the user know that it's good and will start moving. Then the red led will turn solid if the battery is low. I still have to create the code for that though.

d) Today I learned that the ground sensors are too close to the ground to give me the correct reading.

e) Finish soldering status lights.

----

a) Wednesday, July 12

b) 1. Finish soldering status lights. 2. Finish Programing Ground Sensors.

c) Today I was able to finally finish the building portion of my robot. I finally finished the code for the bumper and ground sensors. I used bitwise or for my new function. The function starts off at by reading the IR sensors. Then there has a delay for 100 milliseconds. Then there is another reading and another delay, then one more reading. The reason why I had to take so many readings is because the sensor is so close to the ground and is very sensitive that it won't give me an accurate reading. It displays thing like 0101010110010110 unsted of, 000000 for cliff and 111111 for ground. So because it gives random numbers when on the ground and solid 0s when there is a clif. I just had to return the value of my three readings using bitwise or.

d) Today I learned about using bitwise or. In my function, if all the values are 0s(cliff) then it will return cliff. If one is 1 and the others are a 0 it will still return as ground.

e) Finish up the code and schematic.

----

a) Thursday, July 13

b) 1. Finish coding and new schematic

c) Today I broke two arduinos. I think I burnt the bootloader when programing it. The microcontroller still works but my computer can't communicate with it. I think the reason it's not working is because when I program it, I have the fan and motor driver connected.

It's possible that it's pulling to much volts out of the 5volts usb that it stopped communicating with it as a safety procedure. I also wasted my time working a new code to realize that it doesn't work, so I had to stick with the old code with not as good.

d) Today I learned how to create a function that will blink my status lights. I have the parameters for how long it will blink and how bright
e) Finish the new schematic and hopefully the video

Today I finished my third milestone. I am finally done with my autonomous vacuum robot. I did some small modifications to the hardware, such as some new status lights and a switch. The new status lights allow the user to know what the robot is doing. For example if the LED (Light Emitting Diode) is blinking when you turn it on, then you will know that the robot will need to calibrate for 5 seconds. The IR (Infrared) sensors need time to get a clear reading when powering it up. Then, the light will turn solid letting you know that the battery is full and will start vacuuming the house. If the battery dies, then the led will start blinking again as the buzzer goes off. My switch is just a normal switch to open and close the circuit, turning the robot on and off. I also added and programed the bumper. The bumper consists of two push buttons so if either button gets hit, then it will move away. I finally added a battery alarm, so if the voltage divider doesn't work, then the buzzer will also let you know that the battery is dead. I finally made changes to the code. I made my own function for the ground sensors and the LEDs. I made a function allowing me to blink an LED and change the brightness using PWM. I used two for loops in the function that will keep looping until the time is over. So for one second it will turn on, and the other second it will turn off. For the ground sensors, I made a function that will take readings from the ground sensors. And if it returns that the sensors detect a cliff, then it will turn back. Because the ground sensors are so close to the ground and is so sensitive that it won't get an accurate reading. So my function will take one reading, then a delay for 50 milliseconds. Then repeat that process until it gets a total of 4 readings. Then my function will return either ground or cliff using bitwise OR. So if one of the readings is a 1(ground) then the whole function will return ground. It will only return cliff if all of the readings are a 0(cliff). That is all the modifications I made to vacuum robot. I had so much trouble creating the function for the ground sensors and was very frustrated because I had figure it out before my parent night presentation. I also had trouble fitting all the parts in the robot because the chassis is small. You can also find my code here.

----

a) Monday, July 17
b) 1. Fix the ground sensor code 2. Finish up Milestone #3 Website.
c) Today I was able to finally get a code for the ground sensors to work! Instead of using my more complicated code I used a simpler one. I just had to create my code that will read the values from my ground sensors using an analogRead();. Then comparing the values it gets when on the ground and the values when off the ground, I got the number 150. 150 will be the threshold for the ground sensors. If the ground sensor returns a value greater than or equal to 150, then it's on the ground. If the value is less than or

equal to 150, then it’s off the ground. So depending on what sensors detect the ground or not, the robot will simply turn away.

d) Today I learned that I only have 3 minutes to talk about my project. That's not a lot of time!
e) I want to finish up troubleshooting so my robot is ready for parent demo night.

----

a) Tuesday, July 18
b) 1. Get ready for parent night
c) Today I was able to get my robot prepped for the parent night. All I did was make some modifications to the code so it works well. It still doesn’t do what I want it to do. I would like to work on the code for the voltage divider. The battery voltage will slowly sag. The voltage decreases tremendously when under a great load. Then over time, it will get back up again. It’s a weird characteristic that Lipo Batteries have. I’m working on a code that will ignore the voltage sag and not turn off the fan when it does sag.
d) Today I learned that my code only takes 2 seconds for the arduino to read. This is created because I have other delays for the motors so they have time to move. That means the arduino can process my code very fast.
e) Continue to work on voltage divider code.

----

a) Wednesday, July 19
b) 1. Work on the voltage divider code.
c) Today I worked on the voltage divider code. I’m creating something that will ignore the voltage sagging from the lipo battery. I brought my voltage checker at home that will allow me to measure the voltage coming from each cell. I found out something very interesting The voltage will decrease like crazy when first turning it on. Then after turning it off, then the voltage will go back up again. So in conclusion you can’t ignore the voltage sag without powering it off.
d) I learned that I never had to spend hours on the voltage divider code because of the interesting voltage sagging characteristic.
e) Work on the app

----

a) Thursday, July 20
b) 1. Work on the android app
c) Today I created an app using the mit app inventor. It’s an online app development environment. I created the app so I can connect it with my bluetooth module, control it with arrow keys, press a button to start cleaning, and voice control. I completed getting the app look nice, but now I have to work on the code behind it. It’s challenging to work on the block coding because I’m used to programming in C. It is easier, but just takes time to adjust.

d) I learned how to use mit app inventor. It’s very easy, little typing and a lot of drag and dropping.
e) Finish the app

----

a) Friday, July 21
b) 1. Finish up the app
c) Today I finished the app for my android. You can also control the robot with a computer because they all have bluetooth. I made some improvements to the app so it tells me what the robot is doing. If you don’t have visual of the robot, this a perfect feature for you. I also created a code that will change the clean image button. When you press it, the circle around the word clean will change from grey to green. And will change back when pressed again. This will start and stop the robot from cleaning.
d) I learned the the SH-HC-08 is only a slave module. This means it can only receive data and not transmit. This mean I will need to get a slave and master bluetooth module so the robot can send information back to the user that the battery is dead or is done cleaning.

I completed my final milestone for my vacuum robot. In the end, it vacuums the house for you! besides sucking up dirt and hair, it can avoid walls, stairs, and chairs. It has a 30 minute battery life and will let you know when the battery is low and needs a recharge. Some of the new features that were not shown in the video was the bluetooth capability. I added a little bluetooth module so I can wirelessly control the robot using my phone or computer. For the phone, I created a simple app that can move the robot forwards, right, left, and backwards. It also added a voice recognition capability so you can tell it to clean and move around. Finally, there is the cleaning button where you can start and stop the robot from cleaning your house. To control it from your computer, you can just use the arrow keys and press “c” on the keyboard to start cleaning! One challenge that I faced would be connecting the bluetooth to my phone or computer. I never used bluetooth low energy before and never created code for a bluetooth module. With a little research, I was able to get it up and running.