Automatic Pet Feeder

Fernando Oviedo
Department of Computer Engineering
University of Central Florida
Orlando, United States

Jose Tapizquent
Department of Computer Engineering
University of Central Florida
Orlando, United States

Devin Besaw
Department of Computer Engineering
University of Central Florida
Orlando, United States

Gershon Prospere
Department of Electrical Engineering
University of Central Florida
Orlando, United States

***Abstract*— This document explains the design and functionality of the Automatic Pet Feeder. The motivation behind this project is to address the common need of feeding accurately, in a timely fashion to a designated pet. This will fill the demand in the market because this Automatic Pet Feeder satisfies the need to dispense both food and water while respecting the possible restricted diet the pet owner might have for their respective pets. This particular Automatic Pet Feeder will dispense food and water and inform the owner the quantity served, the amount left in the container, customized set time through the app designed for the Automatic Pet Feeder, and alert the user if the bowl is empty or tampered. This document will further explain the necessary components needed and the ones we considered in the beginning. We will also show the overall schematic and the overall progression of the Automatic Pet Feeder.**

Keywords—Ultrasonic Sensor, Microcontroller, Serial Communication, Mobile Application, PCB.

I. Introduction

Our inspiration for this project came from the fact that there is no Automatic Pet Feeder in the market that can dispense both food and water simultaneously. With the help of a mobile application that wirelessly interfaces with the Automatic Pet Feeder, our team aims to design, program, and construct a functionally stable Automatic Pet Feeder that will allow the user—in this case, the pet parent—to manage all settings, set a feeding schedule, and stay up to date on their pet's feeding habits from their phone. The feeder's user-friendly interface walks the user through the first steps of setting it up and adjusting its default settings. Anyone will be able to do this, regardless of where they are, and feed their pets. There are various conditions that were put in place to ensure that the gadget is completely functional and performs as automatically as possible because this Automatic Pet Feeder must behave in a simple to use manner and provide a lot of value to the pet parents.

The gadget must also be able to precisely gauge the weight of the food and drink bowls and monitor their contents. The device must automatically check for the last pour cycle once the water or food level falls below a certain threshold, confirm the schedule set in the application, and, if the conditions and rules set by the user are satisfied, pour the predefined amount of food for the pet and refill the water bowl to make sure the pet has enough to eat at the pet parent's preferred meal times.

In order to send the user notifications when certain chores, such as replenishing the primary water and food containers or cleaning the pet bowls, call for assistance, the device must be able to maintain a two-way communication with the mobile application.

II. Engineering Requirements

The Automatic Pet Feeder should be a size that is appropriate for a household item; it shouldn't take up too much room, and it should be simple to integrate with other appliances and technology in the pet parent's home. Pet movement sensors are used to ascertain whether the food can be administered at the specified time. Both the bowl the pet will drink from and the container used to store the remaining water used to fill the aforementioned bowl will have sensors to measure the levels of water. When the interval or predetermined feeding period has passed, the motor will be in charge of moving the dispensing gate and dispense the appropriate amount of food into the bowl. When the main water container's level falls below a certain threshold, a water pump that is linked to it will dispense water into the water bowl, a microcontroller that will manage the motor, read sensor data, and enable two-way communication between the feeder and the smartphone app.

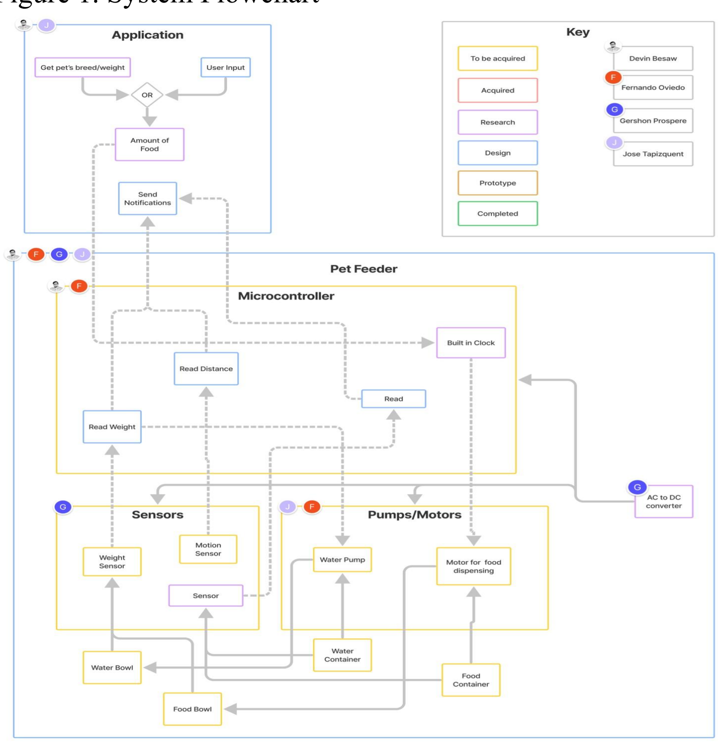
Make sure the Automatic Pet Feeder doesn't disturb other home activities with the noise level it emits. In other words, if the feeder dispenses food in the middle of the night, it shouldn't awaken the entire household. Given that food will be poured from a meal bag, both water and food containers should have funnels at the top to make refilling simpler.

III. Block Diagram and System Concept

The block diagram gives us a comprehensive understanding and big-picture perspective of how all the components of the device interact to effectively address the issues raised throughout this paper. The diagram displays the gadget as a whole, with each of its parts interconnected to demonstrate how data is sent from one component to the other. The diagram also clearly distinguishes between the hardware parts of the gadget and the software program that lets pet parents communicate with it.

We can see from the design that the application's core features allow for simple functions that either allow for the explicit input for the amount of food that should be dispensed based on the user's preferences for that meal cycle, or the automatic calculation of the amount of food to be dispensed based on the pet's weight and breed previously registered by the user. The processor creates the message and decides how much should be delivered to the machine via the communication module after receiving this information from the application. There are currently three main clusters in the design. The microcontroller is the first component and acts as the brain of the system by processing messages from the mobile application, reading sensor input, calculating the weight of the food and water bowls, and communicating with the motors and pump to dispense the proper amounts based on the settings previously set by the user and the instructions received. The sensors are the second major cluster we come across; they are weight sensors for each of the bowls, sensors to gauge how much food and water are in the containers, and sensors to gauge how far the pet is from the gadget. The pumps and motors, which are responsible for reading the data from the microcontroller and delivering the proper quantity of water and food to the bowls during the predetermined feeding cycles, are the last of our components as shown in the figure below.

Figure 1. System Flowchart



IV. Schematic & PCB Design

Before putting the pieces together to construct a prototype and test our concept, we had to create a schematic and specify the connections for our Arduino Uno and Raspberry Pi. We spent some time online researching different schematic construction software programs before settling on EasyEDA because it is both free to use and has a PCB design feature that we could purchase later. There are two schematics for each microcontroller since we are using two of them. We debated whether to use two power sources or try to make the two microcontrollers share a single power supply.

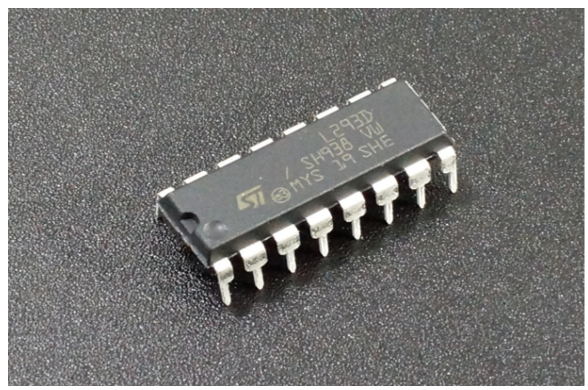
After extensive research and some trial-and-error, we found that we could use the USB components on both boards to power the Arduino with the Raspberry Pi. The raspberry pi was able to manage powering the arduino and the arduino was able to operate correctly. Due to the fact that the only connection between the two would be the USB connection that would power the arduino, we came to the decision that the schematics for both the arduino and raspberry pi could be completed independently at first and then integrated after completion. These are the sections that are part of the PCB of the Automatic Pet Feeder.

A - Voltage Supply
The greatest required voltage is 12V, so the decision was made to use a 12 V AC converter as the power source. This device will convert the voltage to DC power and supply 1 A of current. It also offers the convenience of various connectors, guaranteeing that it will connect regardless of the input port on the circuit.

B - Voltage Regulator
Most of the parts required to finish the Automatic Pet Feeder, as previously indicated, only need 5 V, but because the pump needs 12 V, it would be wise to utilize a 12 V input dc to our circuit. The Automatic Pet Feeder will employ a 5v regulator to use that voltage to power the Raspberry Pi, which is being used to power the rest of the system, in order to prevent any of the components from being damaged.

C - Motor Driver
As shown below, the L293D Motor Driver is one of the parts that is necessary for the Automatic Pet Feeder to work properly. It will be utilized for both the water pump and the motor in this product. Although they will function nearly identically, there are two motor drivers since the motor and water pump require different input voltages. We need two different drivers because the motor driver requires two input voltages to operate properly: one for the motor and one for the devices it drives.

Figure 2. Motor Driver



D - Water Pump
Peristaltic liquid pump is required for this Automatic Pet Feeder. As long as the tubes can inhale the water and pour it into the bowl, it doesn't really matter where the pump is placed (with this exception of being submerged in water) because its weight and size aren't as important. This pump squeezes the tubing and circulates water through it utilizing rotating action. To move water both ways, a rotor is employed with attached rollers that may be spun in either direction. The key advantage of this is that the pump never actually comes into contact with the fluid.

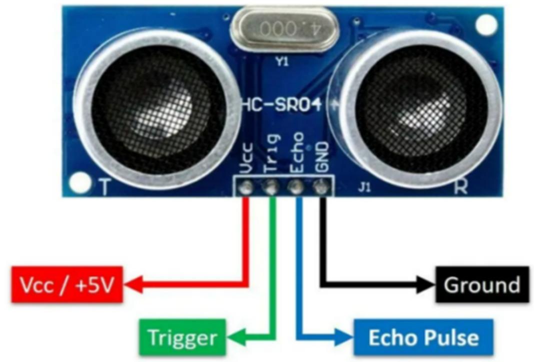
This enables us to use FDA-approved food-safe tubing, giving us the confidence that the water is drinkable as it only flows through the tubing. Here, in the figure below, a tube with an exterior diameter of 4mm and an internal diameter of 2mm is being used. This pump requires 5V to 6V DC power, which can be provided by a power transistor or a motor driver chip. Additionally, the pump doesn't require priming and can self-prime with very little water (half a meter).

Figure 3. Peristaltic Liquid Pump w/ Silicone Tubing

E - Sensors & Motor

The Automatic Pet Feeder requires the use of two sensors for accurate servings and proper connection with the API. One of them is the Ultrasonic Sensor (HC-SR04), which will be placed on the top of the casing of the food and water and will alert the user if the container is empty and transmit a signal to the API about the amount of food and water within. The time difference between transmission and reception is used to calculate the distance between the sensor and the object. In Figure 4, it shows that in order to measure distance using the HC-SR04, pulse the sensor's trigger pin for 10 microseconds. The transmitter circuit then emits eight pulsed ultrasonic waves at a frequency of 40 kHz. As a result, the electrical signal is changed by the transmit circuit into an 8-pulse 40kHz burst. This enables the signal sent to the API to be extremely precise and adjust the distance exhibited in a statement that will display to the user how much food or drink is needed depending on how near or far the product is from the HC-SR04.

Figure 4. UltraSonic Sensor HC-SR04



Under both serving bowls will be the Round Force-Sensitive Resistor. The weight of the bowl, which controls how much food or water is in it, will be made clear to the user. This will also let the user know whether the bowl is full or empty. The Arduino IDE is used to program it, and it operates as follows. The program sets off a flag and stops feeding if the pressure sensor detects a specified level of pressure. The program will notify the user and begin distributing food and/or water if the pressure sensor sends a signal that is close to 0. It is advantageous to have it programmed in this way since it helps manage how much will fill the bowl if it is halfway full; otherwise, it will fill only to the specified level that the pressure sensor is configured to work at.

The DC motor was chosen above the stepper and servo motors for this project because it is not only the simplest to operate, but it also comes with the pre-existing Pet Feeder. We chose a motor with greater placement options because the size and location of the motor are quite important. It is extremely portable and takes up little room. Even at that little, it provides us with 4000RPM, proving that no matter how small it is, it can still do its function. The DC motor has enough power to accurately and quickly dispense food, with a very low possibility of becoming stuck in the middle of a serving.

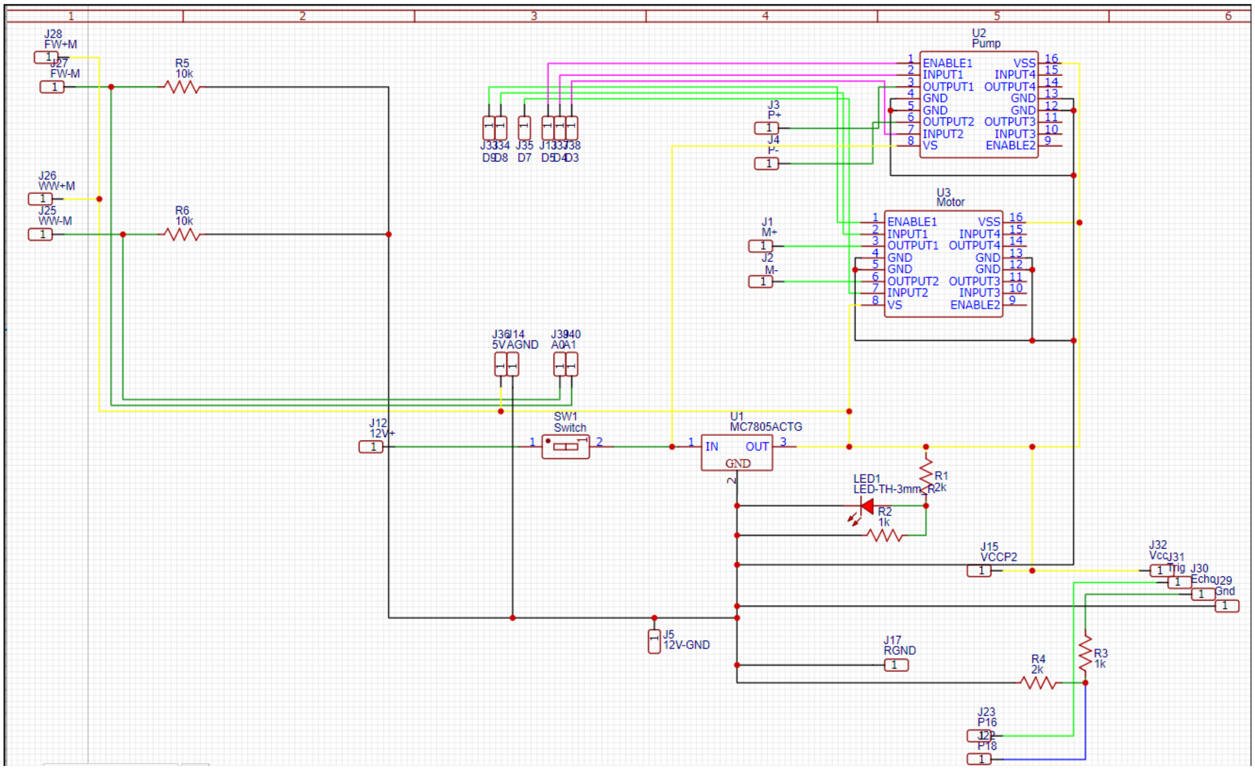


Figure 5. Schematic of the PCB

F - Raspberry Pi Camera Board v2

The Raspberry Pi Foundation has just officially published the newest camera board. It is an improvement above version 1, which used a Sony IMX219 image sensor and 5 megapixels. This camera can produce static images with a maximum resolution of 3280 x 2464 pixels. The IR filter is removed from the NoIR version of this camera, enabling you to use infrared to see in the dark. The interoperability of this camera with the Raspberry Pi is what makes it most advantageous. The Raspberry Pi features a specific software library called libcamera that aims to support complicated camera systems through the Linux operating system in addition to the straightforward connection of this camera. Directly from ARM-powered open source code, the library controls the camera system. Additionally, Libcamera offers programmers access to a C++ API for camera configuration and image frame requests. The system's RAM contains what are also referred to as image buffers.

V - Microcontrollers

The Raspberry Pi 3 and Arduino Uno each include two microcontrollers that will be used in the automatic pet feeder. They have enough pins to connect all of the components, which is one of the reasons behind this. Since they both require 5 volts, it is simpler to power them because the Raspberry Pi will supply power to the Arduino. There is no need to be concerned about running out of memory because both have enough memory to retain the instructions that will be given to them. The Arduino will be used to connect and program almost all of the components. The Raspberry Pi will power the Arduino as well as serve as the link between the Arduino and the software.

A - Arduino UNO R3

The ATmega328 microprocessor is used in the high-performance, low-power Arduino UNO microcontroller board. The beautiful thing about it is that there are lots of online resources we can utilize to learn more about it because it's so well-liked. Its simplicity is a significant plus because it makes coding simpler. It contains a 16 MHz crystal oscillator, 23 general-purpose input/output pins, and a USB port. Around 2.7 to 5.5 V is the operational voltage. Six of the board's 23 input/output pins are utilized for Pulse Width Modulation (PWM), while six are used as analog pins. The 10-bit ADC provided by this microcontroller, which governs the precision of the sensors' measurements, is another crucial component. Additionally, it has a low power requirement. 1.5 mA at 3 V and 4 MHz would be the current in active mode. It can produce 1 uA at 3 V while in power-saving mode. The Arduino software can be used to program this microcontroller.

B - Raspberry Pi 3

The Raspberry Pi 3 is an improved version of earlier Raspberry Pi gadgets. One, two, or four gigabytes of RAM are available as options. It offers a Bluetooth interface with Bluetooth Low Energy to its consumers (BLE). The board has a total of 40 general purpose input/output (GPIO) pins. UART can use up to six pins, I2C up to six pins, and SPI up to five pins. When connected to a USB-C, the Raspberry Pi requires an input voltage of 5 V at 2 A. Additionally, the Raspberry Pi has excellent documentation and a variety of software libraries. A Linux-based operating system called Raspberry Pi OS has also been developed for the microcontroller (previously known as Raspbian). As a result, the user experience is enhanced and the hardware is more intrusive. The development environment is the greatest of any microcontroller, and it supports Python, a contemporary language that emphasizes on making software development simple.

Table 1. MCU

	Arduino UNO R3	Raspberry Pi 3
GPIO	40 pins	14 pins
Input Power	5V DC via GPIO	5V
Memory	1GB	32KB

VI - Housing

We ultimately decided to use a pre-built pet feeder for the food and a custom-built water container for the housing. This choice was made in order to avoid spending time and resources on creating a reservoir and food-dispensing system. Instead, we would like to put that time and effort into creating the software and hardware for our pet feeder. Since we will be utilizing a pump to move the water, all we really need is a container that is watertight. Since the area doesn't need to be very large, we first thought about creating a water reservoir, but we had problems waterproofing it; as a result, we bought an existing one and installed the pump to it. As shown in the figure below, the water reservoir has one of the hose from the Peristaltic pump into the container while the other one is attached to the bowl.

Figure 6. Food and Water Housing

VII - Application Design

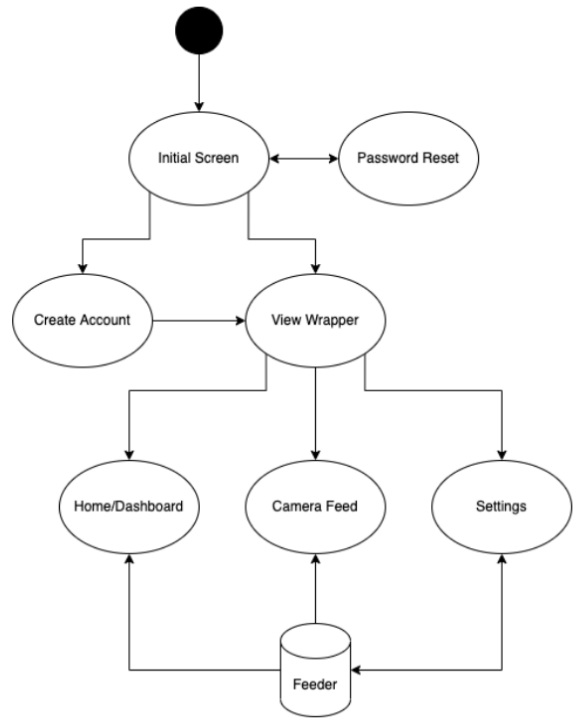
Our project's software, which is divided into two components, is what will connect everything. Our mobile application, which is for the user experience, is the initial component. Here, we'll be able to provide data and let people customize it as they see fit. The primary channel of communication between the pet parent and the Automatic Pet Feeder will be the smartphone application. With the exception of turning the feeder on and off, almost all interactions will be made through the app. Messages from the device will also be sent to the mobile application. Our embedded systems, which will link our mobile application to our hardware, make up the second component. Access to the camera feed and any additional sensors from the feeder are also made easier by the firmware. The microcontroller's software will also act as the feeder's central processing unit.

The firmware simplifies the initial configuration of the device, connects, and regulates the handling of each event posted by the mobile application. By calculating the number of rotations based on the amount of food set by the user through the mobile application, it will also manage how the motors rotate to distribute the food. Once the bowl is at the proper level, it will command the water pump to pour water.

A - Mobile Application

The mobile application will ask the user to sign in with an existing account or create a new one when it is initially started. The user will be asked to couple the feeder with the app after logging in. If they are a first-time user, a guide will be shown after pairing to walk them through basic setup. The mobile application will enable the user to register all of the pet's information, including name, age, and weight, once the device has been paired. The user will then have the ability to modify the feeder's various settings and scheduling. The process of the mobile application that is in line with this project is depicted in the following flowchart in Figure 7.

Figure 7. Mobile Application Flowchart



In order to establish its schedule, dispense food or water, or monitor the video feed, the Automatic Pet Feeder receives messages from an external device running the mobile application and processes the commands. Therefore, the user must first download the mobile application to their mobile device and couple it with the Automatic Pet Feeder in order to control every part of the feeder.

B - Communication

It's crucial to first clarify that there are actually two different communication protocols in use. The Raspberry Pi and PCB communicate serially during the first communication, while Wi-Fi is used for the second communication, which connects to the internet.

The Raspberry Pi and PCB should always communicate using UART serial connection. What is being communicated between the two components must first be determined. Since the camera is the primary purpose for using the Raspberry Pi, the microcontroller must then respond with the view and commands from the camera.

VIII - Conclusion

We have encountered several challenges that have impeded our progress as we have worked on this project consistently, but we have been able to get beyond practically all of them. We chose a pet feeder because we all recognize the value that pets have for families all across the world. This project is a crucial component of our engineering education. After spending years learning about various subjects relevant to our degree, this is where we get to work practically and gain engineering experience.

References

"Engineering Subject Guide: Standards." *LibGuides*, https://libguides.trinity.edu/engr/standards#:~:text=A%20standard%20can%20be%20defined,on%20engineering%20components%20or%20equipment.

projects, Contributors to Wikimedia. "Peristaltic Pump." *Wikimedia Commons*, Wikimedia Foundation, Inc., 10 Mar. 2019, https://commons.wikimedia.org/wiki/Peristaltic_pump.

Eslam, et al. "HC-SR04 Ultrasonic Sensor Interfacing with Pic Microcontroller - Distance Measurement." *Microcontrollers Lab*, 25 Oct. 2020, https://microcontrollerslab.com/ultrasonic-sensor-interfacing-pic/.

"HC-SR04 Ultrasonic Range Sensor on the Raspberry Pi." *The Pi Hut*, 22 Oct. 2013, https://thepihut.com/blogs/raspberry-pi-tutorials/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi.

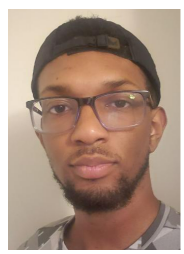
"Raspberry Pi Documentation." *Camera*, https://www.raspberrypi.com/documentation/accessories/camera.html.

Last Minute Engineers. "In-Depth: Control DC Motors with L293D Motor Driver IC & Arduino." *Last Minute Engineers*, Last Minute Engineers, 18 Dec. 2020, https://lastminuteengineers.com/l293d-dc-motor-arduino-tutorial/.

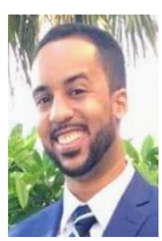
Biography

Jose Tapizquent is an undergraduate student at the University of Central Florida pursuing a Bachelor of Science degree in Computer Engineering. By learning to code through online courses and school, he was able to obtain a position as a Software Engineer at UKG, a Saas company based out of Weston, FL.

Devin Besaw is an undergraduate student at UCF pursuing a B.S. in Computer Engineering. Currently, he is interning as a Client Engineer for the startup company SightPlan. In high school his favorite subjects were math and web design which led him to choose this degree and the position on the SightPlan team as a Client Engineering Intern (Front-end web development).



Gershon Prospere is an undergraduate senior at UCF pursuing a B.S. in Electrical Engineering and will graduate in August of 2022. He started his passion for electrical engineering as a kid, discovering all of the circuitry inside technology and has stayed curious on how these devices work and how they keep improving as time continues to go on. After graduating, he plans to work at Texas Instruments.



Fernando Oviedo is an undergraduate senior at UCF pursuing a B.S. in Computer Engineering. He originally was interested in Information Technology but then he wanted to know exactly how a computer works and what makes it function, when does it fail and why is it happening. After graduation, he plans on getting a job as a Systems Engineer and then pursuing his masters degree in Hardware Security/Cyber Security.