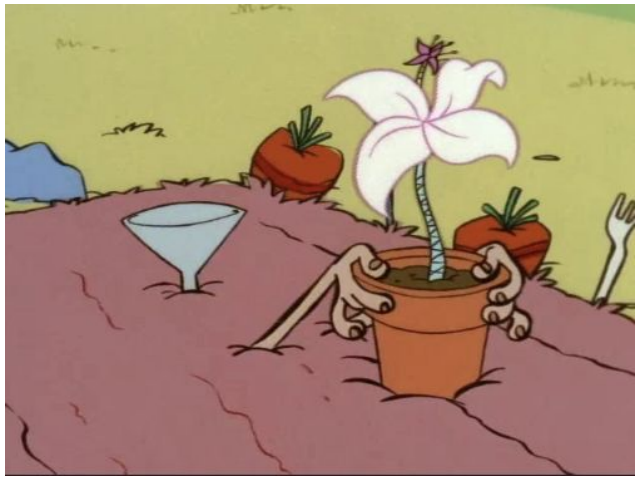
Automatic Irrigation System

Jenny, Daria, Vera, Okasha & Juan



What is an Automatic Irrigation System?

- a system that automatically waters plants.
- we have a soil moisture sensor.
- when the soil moisture is low, the water pump turns on.
- water is extracted from a supply through tubing

Everyday Life

Automatic irrigation systems can be integrated into everyday life such as:

- Large-scale: agriculture farming with less manual labor and time used
- Smaller-scale: maintaining a lawn or outdoor garden with automatic sprinklers
- Improving crop quality
- Control System and regulation

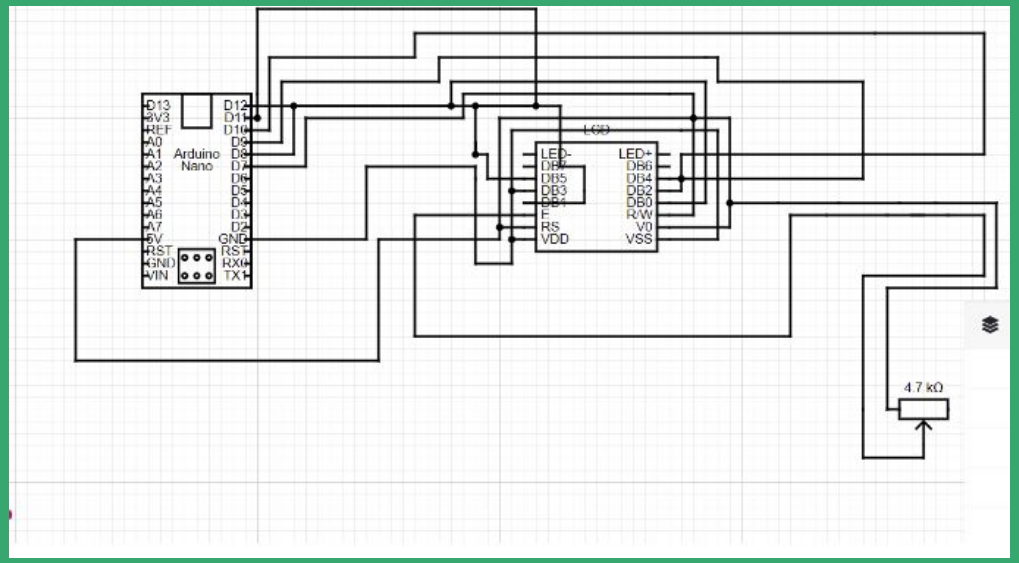
Problems It Solves

- Short-Term vacations
- Forgetfulness
- Efficiency Issues
- Overhydration

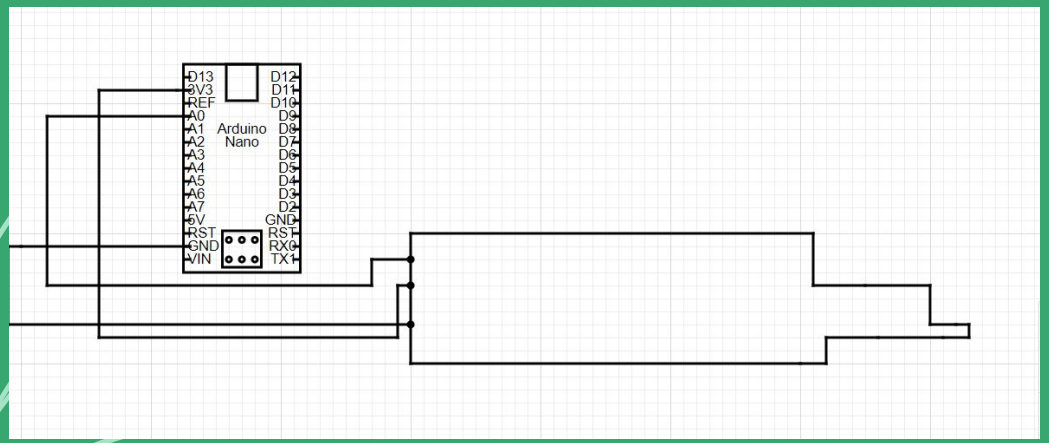
Our Designs

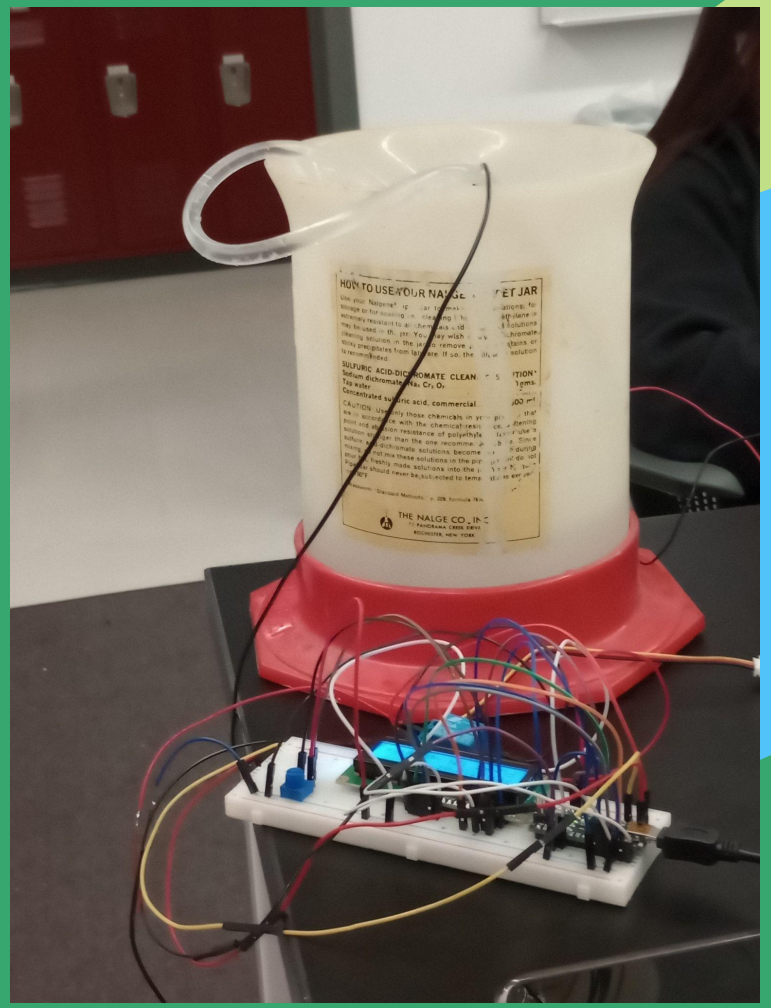
The greenhouse protective case was 3D modeled in Fusion 360. It was then 3D printed on a Prusa 3D printer in the UMB Makerspace.

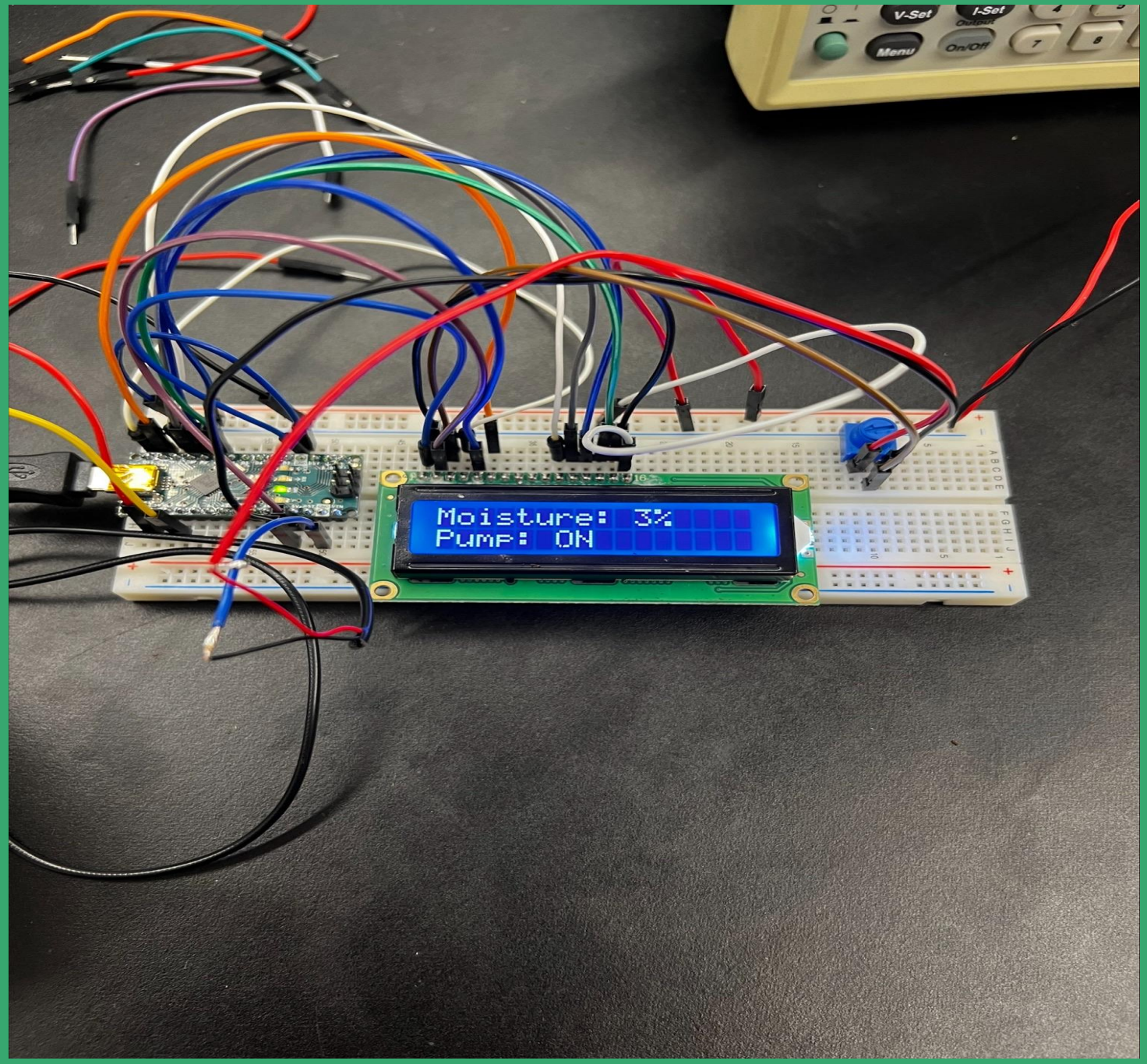
The case was sized and resized to contain the breadboard and display, as well as house the wires.



Project Results






V-Set
I-Set
Menu
On/Off
Moisture: 3%
Pump: ON

Factual Truth

- If the Moisture Percentage is Above 70 percent, It’s High though but the Relay turns off (PUMP OFF).
- If the Moisture Value is Below 30, It indicates that the Moisture is Low but the Relay turns on for the pump to activate and it also starts process of Irrigation.(PUMP ON)

Keep this in mind: Arduino has the decision to turn on and Off if the Moisture Percentage is low.

Project Results Sequel

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
const int AirValue = 600;
const int WaterValue = 310;
const int ThresholdValue = 484;
int soilMoistureValue = 0;
const int RelayPin = 2;
void setup()
{
Serial.begin(9600);
lcd.begin(16, 2);
pinMode(RelayPin, OUTPUT);
digitalWrite(RelayPin, HIGH);
}
void loop()
{
soilMoistureValue = analogRead(A0)
```

```
Serial.println(soilMoistureValue);
lcd.setCursor(0, 0);
lcd.print("Moisture: ");
float moisturePercentage = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
lcd.print(moisturePercentage, 0);
lcd.print("%");
int upperLimit = ThresholdValue + 0.1 * (AirValue - WaterValue);
int lowerLimit = ThresholdValue - 0.1 * (AirValue - WaterValue);
if (moisturePercentage < 30.0)
{
digitalWrite(RelayPin, HIGH);
lcd.setCursor(0, 1);
lcd.print("Pump: ON ");
}
else if (moisturePercentage > 70.0)
{
digitalWrite(RelayPin, LOW);
```

```
lcd.setCursor(0, 1);
lcd.print("Pump: OFF");
}
else
lcd.setCursor(0, 1);
lcd.print("Pump: ");
if (digitalRead(RelayPin) == HIGH)
{
lcd.print("ON");
}
Else
{
lcd.print("OFF");
}
}
delay(250);
lcd.clear();
```

Main Codes(Movements)

Set Cursor: Sets a Location

Print: Printing Process

Delay: Waiting

Semicon {}: Block or Braces.

Setup: Starting or Initializing Ex. (.Begin)

Loop: Change or Respond

Main Codes (Movements) Sequel

Constant Integer: a value determined to compile time but cannot change time.

Integer: stores a number in 2 bytes.

The Code Defines the Threshold value, and the Air and Water Value.

Q&A?