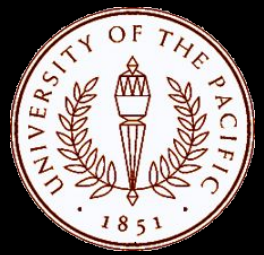
Projects and Achievements

Joanna Hughes
(530) - 490 - 1455
joannakimikohughes@gmail.com

Github:
JoannaKHughes



B.S. Computer Engineering &
M.S. Computer Science
Expected May 2025

LinkedIn:
Joanna Hughes

Fall 2023

&

Spring 2024

CO2 Tracking Website

ELECTRONIC TREE

- A way to reduce CO2 -

What is it?

A sustainable system created to reduce global carbon dioxide emissions through solar power and biodegradable materials.

Project Statement

The system is a sustainable mechanical/electrical tree that takes in air, filters out Carbon Dioxide (CO2) molecules, stores the CO2, and releases the filtered air back into the environment.
This device also records the amount of CO2 pre-filtering, post-filtering, and the aggregate amount of CO2 captured over the "tree"s lifetime.

Motivation

- Approximately 400 parts per million of Carbon Dioxide are emitted into the atmosphere daily worldwide.
- The amount of CO2 and other greenhouse gases in the air causes the climate change impacts such as rising sea levels, stifling heat waves, and forest fires.
- This device will be a useful addition to the local ecosystem by increasing the air quality.
- The "tree" is designed to continuously capture CO2 throughout the day.
- This system will retain the CO2 until it reaches its saturation point. Notably, the device will capture more CO2 than it emits, thereby serving as a valuable enhancement to the local ecosystem.

Project Objectives

- Mimic the operation of a full grown tree
- Filter CO2 out of the atmosphere
- Achieve Carbon Negative status
  - a.k.a., Remove more CO2 than it produces

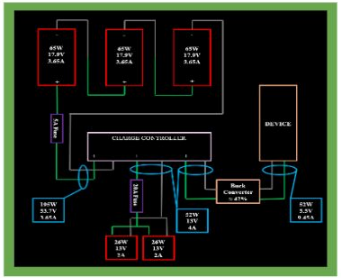
Filtration System

SODA LIME - CaHNaO2

- Soda lime absorbs about 19% of its weight in carbon dioxide, hence 100 g of soda lime can absorb approximately 26 L of carbon dioxide.
- Purple Soda Lime indicates CO2 saturation.
- Saturated Soda Lime becomes limestone.

Acknowledgements

- Mr. Mark Foreman, Mr. Jeremy Hanlan, and all the student staff at Makerspace: For all the help while working in the Makerspace
- Dr. Pallipuram: For advising during the second semester
- Dr. Khoie: For advising during the first semester & sponsoring the project



Components/Testing Completed

- 3 solar panels connected in series - provide 65W of power on a cloudy day and 150W+ on a sunny day
- 2 batteries connected in parallel - Totaling to 52W of power and 4Ahr of current
- Charge controller - connects the solar panel array and battery array to the rest of the components
- Tested the soda lime filters under 3 different conditions such as damping the substitute with water, leaving a 2 tablespoons of soda lime into a plastic bottle with the ends cut off and open and the bottle was kept inside of a room with stagnant air and confirmed that they capture CO2, and finally using a candle as a CO2 source.
- Tested a Raspberry Pi 4 with Linux for quality assurance as well as performance optimization of applications and configurations specifically tuned for embedded Linux on Raspberry Pi hardware.
- Sensor testing and Pi compatibility was done by connecting the K30 CO2 sensors parallel to each other and allowing the sensors to gather CO2 readings of a room.
- Data is collected for the CO2 sensor such that the data is saved onto a text file using the following format "hh:mm:ss MM/DD/YYYY : CO2 PPM". This is done such that there is a "hard copy" of data should internet connection be lost in the event of network failures or power cycling. Data is then sent to a google sheet using Google's API and is stored into the sheet using the same format as it is saved in the text file.

Power Budget

19.89 Watts

Dollar Budget

$566.43

Carbon Footprint Calculations

CO2 emissions during manufacturing the "Tree":
- Solar Panels = 0.15 Kg
- Batteries = 1.26 Kg
- 4ft x 8ft plywood = 190.3 Kg
- RPI & Sensors = NA

} Total ≈ 192 Kg ≈ 37282.92 ppm

CO2 emissions during running for 8 hours:
- Solar Panels = 0.064 Kg
- RPI = 0.0076 Kg

} Total ≈ 0.072 Kg ≈ 13.98 ppm

Current Filtration Rate (5 Hr/Day):
- 50 ppm/Min = 17,000 ppm/day
- System will be CO2 Negative in **3 Days** of operation.

Electronic "Tree" by Joanna Hughes, Salma Sakhizada, Michael Solis, and Eric Lopus

Fall 2022 & Spring 2024

MP3 Player

- Coded hardware with C and assembly
- Worked as a group to create a functioning MP3 player on a TM4C microcontroller
- Used knowledge of circuits to connect an auxiliary port and an LCD screen

Applied Skills

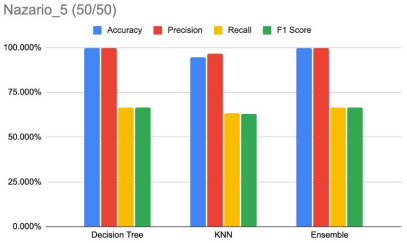
- C
- Datasheet Interpretation
- Circuits & Wiring
- Python
- Google Collab
- Machine Learning
- Data Sets

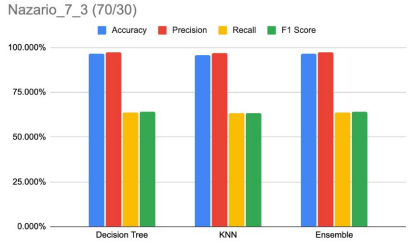
Machine Learning Phishing Email Catcher

To develop this model, the Scikit-learn library in Python was used to load classifier models for our Stacking Ensemble model. The data is labeled with 0s and 1s to indicate whether it is a phishing attempt. The distinction between the Nazario_5 and Nazario_7_3 datasets lies in the different proportions of phishing emails they contain. We employed the Pandas library to load the dataset for our model. During the testing phase, we also evaluated the performance of Decision Tree and K-Neighbors models.

Nazario_5	Decision Tree	KNN	Ensemble
Accuracy	99.837%	94.625%	99.837%
Precision	99.894%	96.707%	99.894%
Recall	66.667%	63.248%	66.667%
F1 Score	66.613%	63.133%	66.613%

Nazario_7_3	Decision Tree	KNN	Ensemble
Accuracy	96.580%	95.928%	96.580%
Precision	97.514%	96.896%	97.514%
Recall	63.861%	63.446%	63.861%
F1 Score	64.016%	63.503%	64.016%





Spring 2021 & Fall 2021

Banking Interface

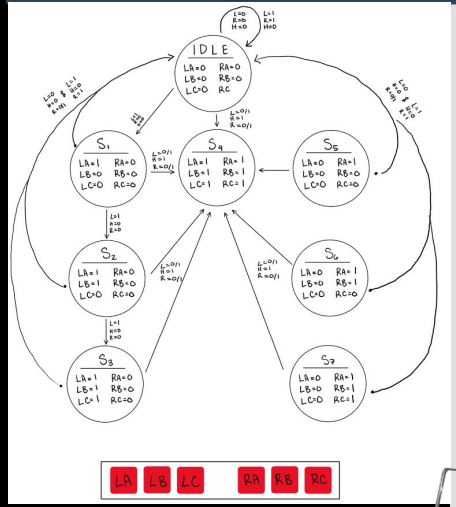
Created a user-friendly banking interface allowing employees to manage customer accounts efficiently.

- Proficiency in C++ programming.
- Implementation of efficient data structures.
- Organization of projects using .h and .c files.
- Application of inheritance for code structure.
- Effective management of variable access and messaging clarity.

Applied Skills

- C++
- Data Structures
- Object Oriented Programming
- Algorithm Analysis
- System Verilog
- State Machines
- Digital Logic
- Cyclone V

FPGA Tail Light Sequences



Design and implement a state machine that mimics the drive circuitry for the taillights of a Ford Mustang.

Summer 2021

Smart Wheelchair Prototype

- Developed an automated wheelchair prototype robot
- Utilized Python and C++ for programming
- Obtain directions using QR codes deciphered using Python and Computer Vision
- Presented findings at a professional conference

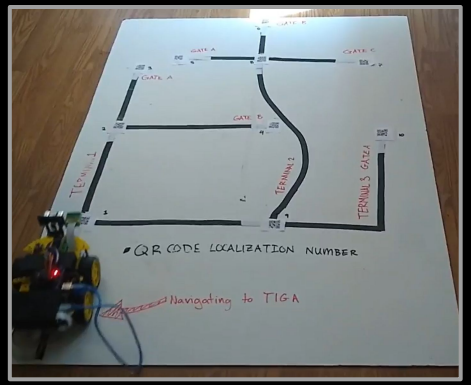
Applied Skills

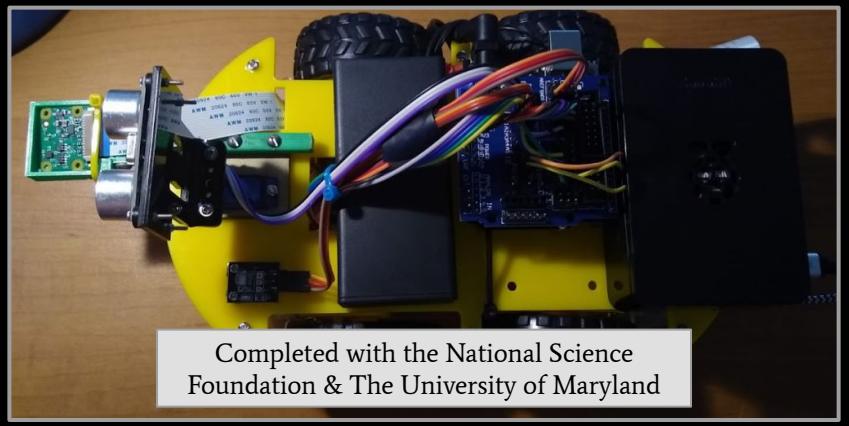
- C/C++
- Python
- Computer Vision
- Linux Terminal
- Virtual Machine

Hardware

- Raspberry Pi 3
- Arduino

Link to Presentation





Completed with the National Science Foundation & The University of Maryland