2022 Design and Prototype Finalists

GUI NanoLab

Students: Arthur Trapp, Luke Ehrisman
Teacher: Michael Couvillon
School: Jesuit Dallas, Texas

Students: Josueth Salverredy, Aurelie Elias
Teacher: Thomas Viola
School: Platt Technical

Students: Noah Kay, Noah Kaye
Teacher: Chris Regini
School: Half Hollows Hills, New York
Students: Ben
Teacher: Stacy Armstrong
School: Cypress Wood, TX

Students: Joseph Abbot, Reilly Moore, xxxx
Teacher: Robert Burke
School: iSchool of Lewisville, Texas

Students: Bridgette Nealy and Kavin Lajara
Teacher: Thomas Viola
School: Platt Technical, Connecticut

Students: Rianah Allen
Teacher: Thomas Viola
School: Platt Technical, Connecticut

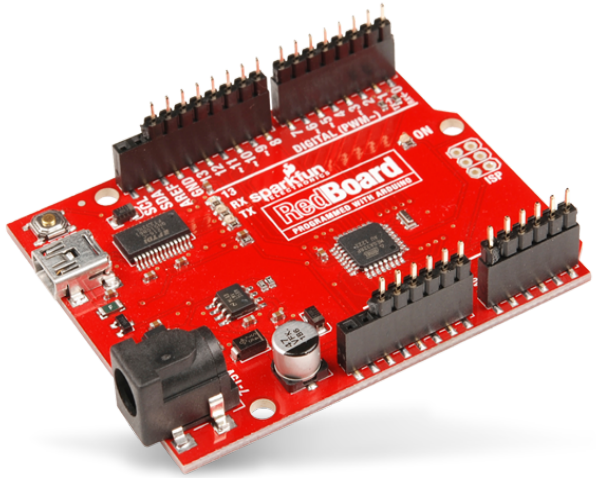
NanoLab Software GUI

Arthur Trapp, Luke Ehrisman
Jesuit Dallas
12345 Inwood Rd,
Dallas, TX 75244
(972) 387-8700

https://youtu.be/FpbfQ7FKk6Y

Task

Create an easy to use GUI to control and view data of the Agriculture, Fungus, Fermentation, and Crystallization NanoLabs. Must have all sensors and can be interacted with. Must be able to transfer photos and data from orbit. Simple concept.



Why Arduino?

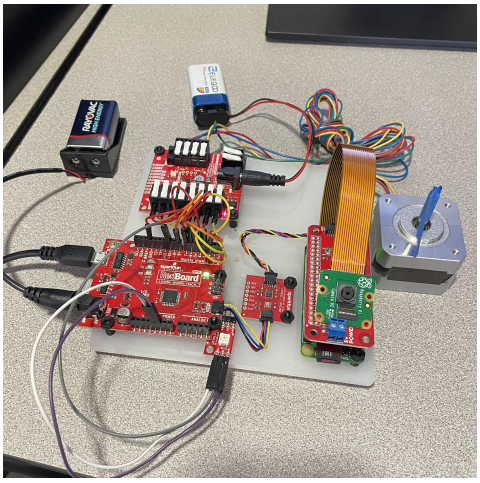
Microcontroller tasks such as reading data from sensors and controlling motors are better with Arduino, operating system is not required for Arduino to run, the simplicity of Arduino, and we already have slight experience with Arduino. We used a Sparkfun Kit

Program

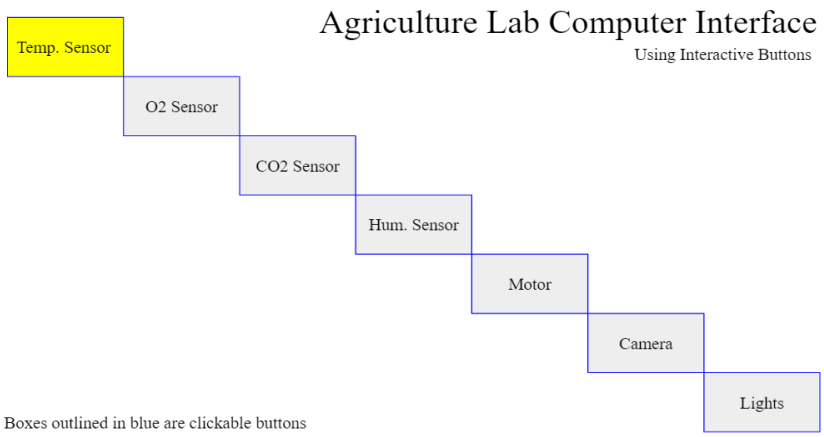
Our NanoLab coding will be done with a computer program called Processing. The Processing program is an easy to learn and simple software sketchbook, which is compatible with the Sparkfun Kit. Our program will only display the Agriculture Lab as an example, but this program can be used with the other labs using the same methods.

Agriculture Lab Materials

SEN-14348: Environmental Sensor (Temp., Humidity, O2, CO2)
SUB-15182: LED Light
ROB-09238: Motor
ROB-16836: Motor Board
DEV-15945: SparkFun Qwiic pHAT v2.0 for Raspberry Pi
DEV-14028: Raspberry Pi Camera Module V2
DEV-15470: Raspberry Pi Zero W (with Headers)



Temperature Sensor Walkthrough

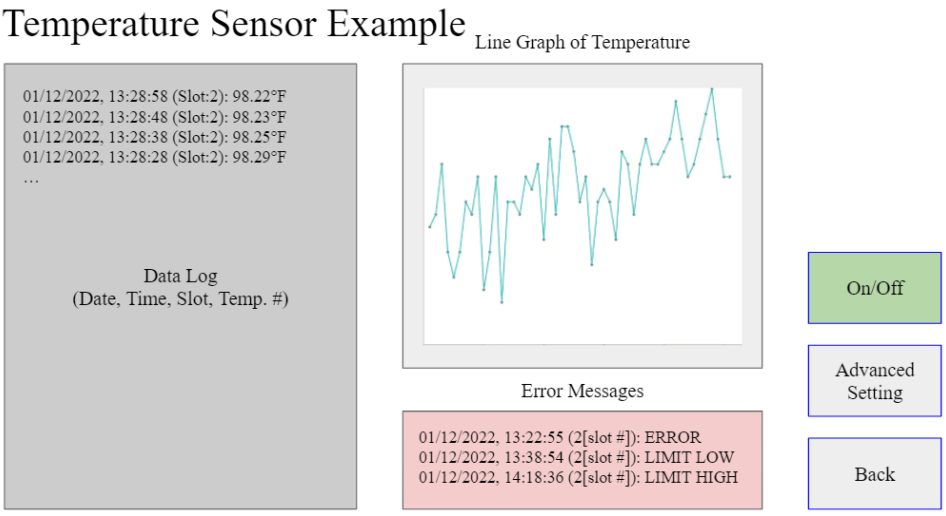


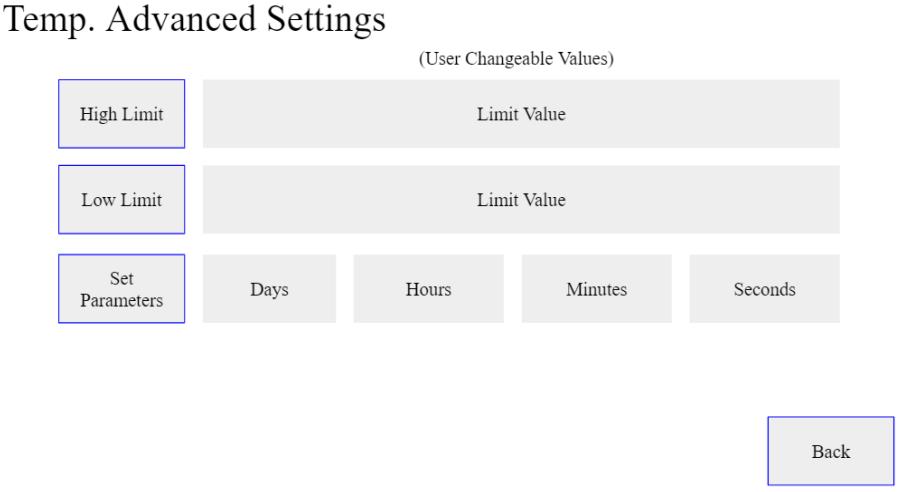
Stage I

This is the homescreen of the Agriculture Lab. Here, the user can select what specific technology they would like to select by clicking on each button.

Stage II

After clicking on the "Temp. Sensor" button on the homescreen, the user may observe a data log, error messages, and line graph, which will all be stored on a harddrive for further usage. The user may turn off the sensor if designed and access the "Advanced Settings."





Stage III

Here, the user has access to make specific changes to the sensor. The user can set High/Low limits of the temperature, alerting the user and set how often the sensor records data.

Nano Lab GUI

PLATT TECH

INFORMATION TECHNOLOGY

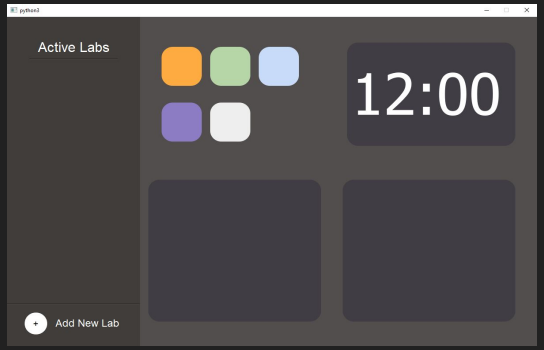
Platt Technical High School

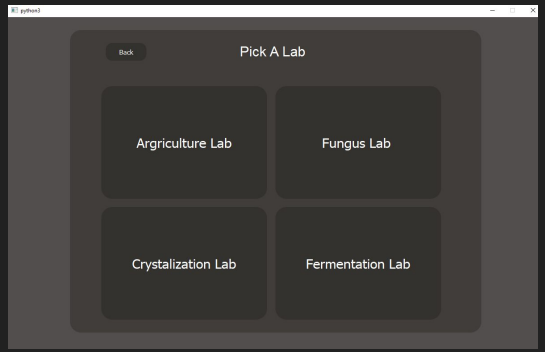
Josueth Salverredy & Aurelie Elias

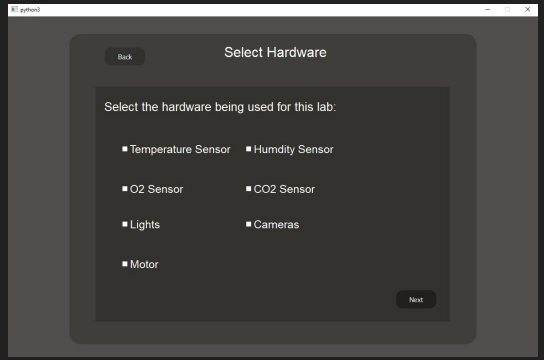
Thomas Viola

Project Description

The Main Goal is to create Software that will allow the user to Manage Labs, control sensors, Extract Data, and other actions related to lab functionality.







Live Photos of Our Prototype

The Team

demonstration video
https://youtu.be/40Dh25a84t8

Graphical User Interface

Engineers: Noah Kay and Noah Kaye
HHH High School East
Mr. Regini

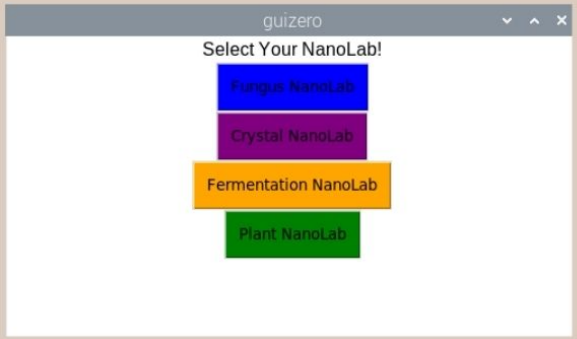
Data Collection

- After the 'Confirm Changes' button is pressed, the data will start being collected, and then when it is finished, it will be sent into a CSV file
- The CVS file will display the data collected in graphs and charts

Help Buttons

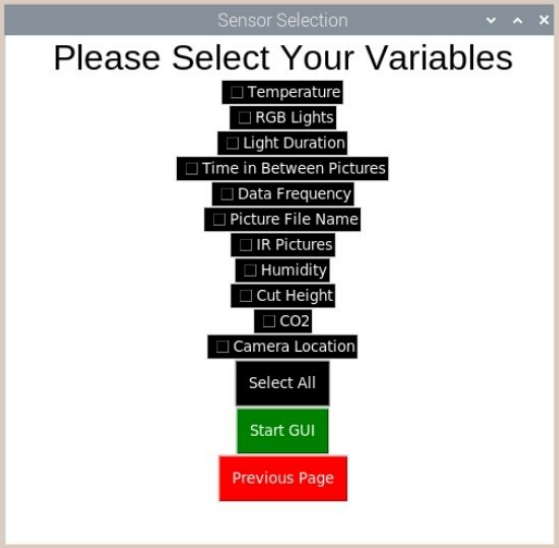
- The photo is an example of the RGB light help
- The help screen will explain what the button is necessary for and what it will do when changed
- This will make the experience of the GUI more user-friendly because they will further understand what they are selecting

NanoLab Selection



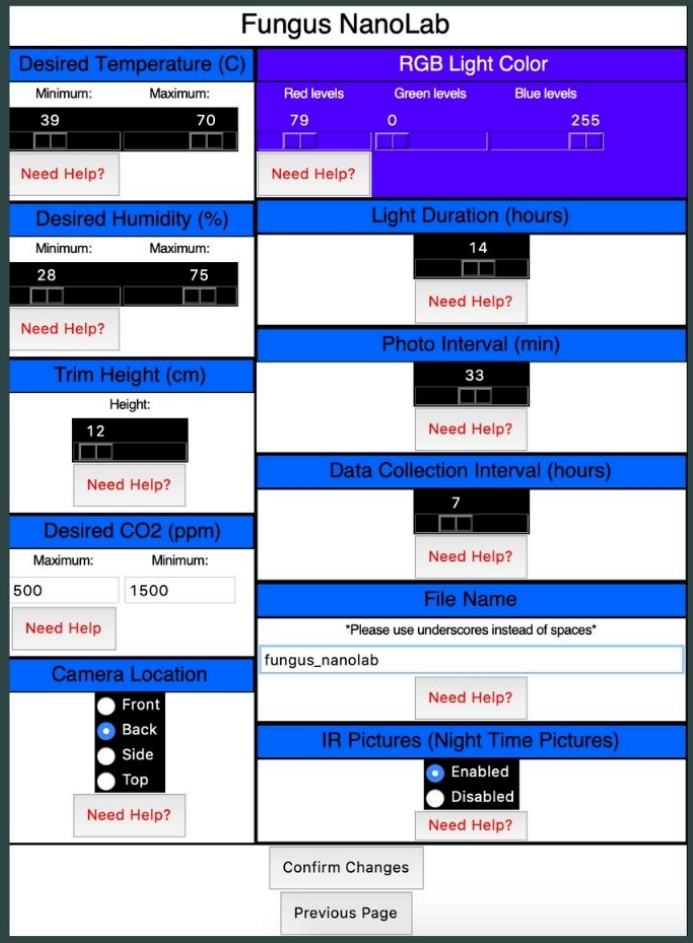
- This is the opening window of the GUI
- 4 push buttons that represent the 4 NanoLabs
- This button will take the user to the next window

Variable Selection



- This window follows after selecting the NanoLabs
- 12 check boxes that represent the variables
- A 'Select All' button to select all variables
- The green button to start the GUI and the red button to return to the previous page. (NanoLab Selection)

Variable Screen



- This window will display the sensors that the user has selected
- If the user doesn't know what the sensor means. there is a 'Need Help' button to explain the button
- If the user didn't put all of the variables they want, they can select 'Previous Page' to go back
- After the user puts in there desired values, they will press the confirm changes button.


ELMER'S
Tri-Fold Display Board
ELMER'S
Tri-Fold
Purell

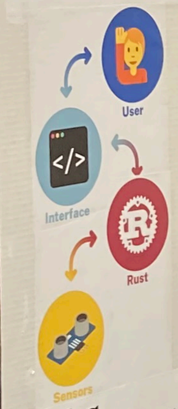
Nanode

next-level control, next-level science.

Nanode allows anyone to **design**, **develop**, and **deploy** Nanolab experiments with ease.

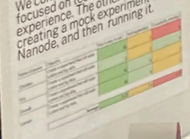
NASA Problem

The **complexity** of programming prevents scientists from conducting research **efficiently**. Design a **graphical user interface** that scientists can use to program **Nanolabs** without the **hassle** of programming languages.

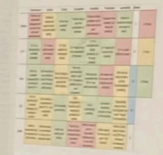


Testing

We conducted two tests, one which focused on testing the user experience. The other test was creating a mock experiment with Nanode, and then running it.



Decision Matrix



Development

Using **Nim** to develop our **frontend** has proven to be very successful, until it wasn't. However, our foray into **Go** for our **backend** has shown **problems**. For this reason, we switched to using **Rust**, which has proven to be a very **beneficial** decision.

Questions

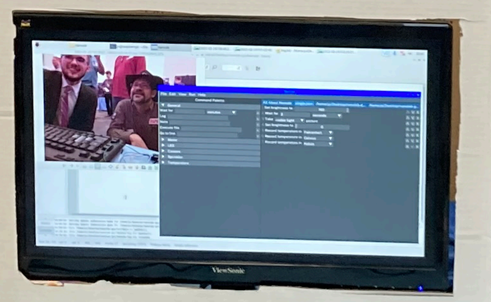
Will the GUI have a connection to the internet?
No, it will only have internet during the initial setup on Earth.

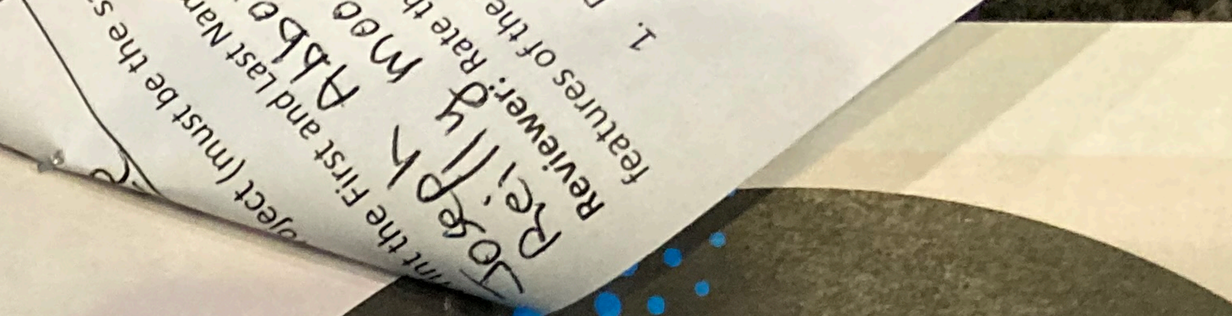
How will it connect to other computers on the ISS?
The Raspberry Pi will connect through a USB connection.

What is the "NanoRacks Downlink"?
It's a product created by NanoRacks for data transfer.

Where will the GUI be hosted in relation to the NanoRacks?
Each Nanolab project will have its own Raspberry Pi.

How will collected data be stored?
On the Raspberry Pi's memory, to be transferred later.





Nanode allows anyone to design, develop, and deploy Nanolab experiments with ease.

Nanode
Next-level control,
Next-level science.

Contact us

650 Bennett Ln., Lewisville, TX
75057
contact@nanode.click
www.nanode.click

Key features.

Every device.
Every configuration.

Nanode provides support for a wide array of device types. Mix and match devices to your exact specifications, and Nanode will verify compatibility before you even click *Run*.

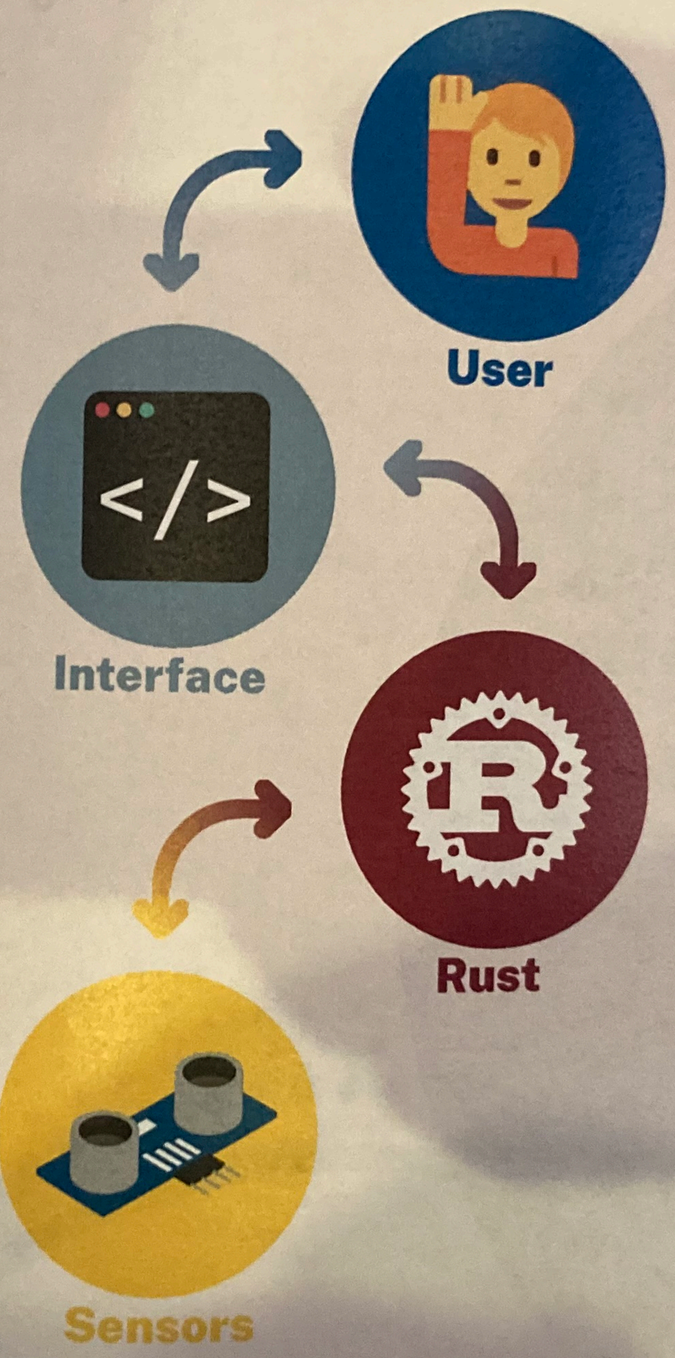
Runs everywhere.
(Batteries not included.)

Nanode is designed to run on Raspberry Pi hardware, but the frontend will run on all major operating systems. Adding GPIO support for your favorite devices is easy, as Nanode is heavily documented.

Nice to look at.
Nicer to use.

We're sick of professional software looking like a dog's dinner. We designed Nanode from the ground up to balance form and function. With an elegantly designed, and aesthetically pleasing user interface that doesn't skip on functionality.

Our flow.



Let's talk technicals

Down to the metal.
Crazy performant.

No runtime. No bloated GUI library, or dependencies. Nanode uses safe OpenGL for hardware-accelerated graphics, and Dear ImGui to draw optimized vertex buffers, which means our rendering pipeline is tiny and blazingly fast!

Written for everyone.
Delightfully hackable.

Nanode is built using modern programming techniques, which promotes reliability, readability and maintainability. Nanode is designed to be hackable by everyone, no computer science degree required.

Every line open source.
All the way down.

Not only is Nanode fully open source, and under the permissive MIT license, but every resource used to build Nanode is also available under an open-source license. Even our programming language is open-source.

NASA HUNCH GUI PROJECT(*Design Prototype And Software*)

Info of group:

Platt Technical High School, Milford CT.

Teacher: Thomas Viola

Group Members: Bridgette Nealy and Kavin Lajara

Description:

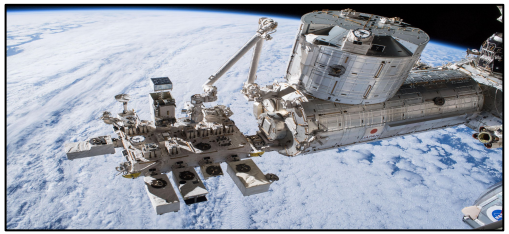
With the interface we are creating, we are going to have a section where you can access recent labs that are in progress, there will be a sensor and camera activator to determine whether the cameras and sensors are working, off, or having issues, this way it makes it easier on researchers. With our start up menu, we will have a login screen for each researcher to login so that way their files are organized, then if a researcher does not have an account within our system, there will be an option to create one! Once that is all completed the researcher will see the main page and from there they will be able to access the settings, lab page and other applications. Once a lab screen is picked there will be a select hardware page where they can select any hardware applications they may need for their experiment that they need to connect to. SQLite will be the database behind the GUI to organize information and make sure the GUI keeps everything together.

Video to our displays:

https://www.loom.com/share/6b4894b2eb5b4472a3d7b477a941d288

NanoLab Software Python GUI

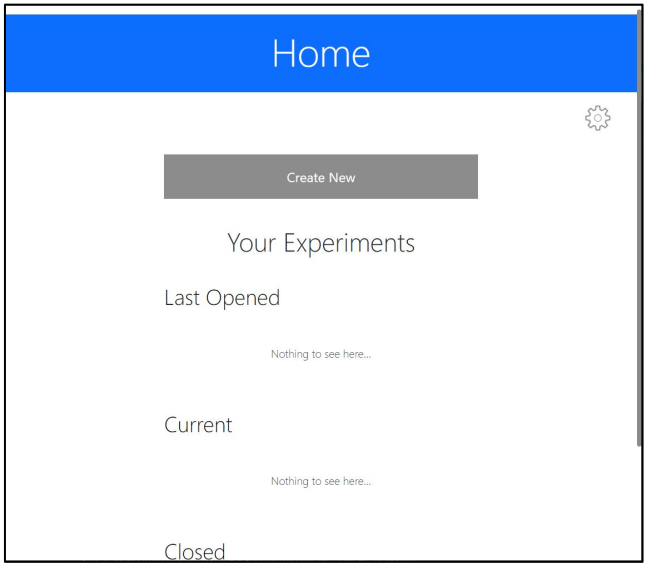
Platt Technical High School
Milford, CT
Mr. Viola
Rainah Allen



Some of the main features of this GUI are simplistic design that make the GUI easy to use and understand for a variety of users. The GUI is an easy-to-use software that will allow experimenters to effortlessly create and outfit their NanoLab experiments.

Generalization and Simplicity are important factors when it comes to the development of a GUI designed for many users of different age levels and backgrounds. That is why this GUI is designed with these factors in mind. The way that this GUI is designed allows for easily creating different types of experiments along with the basic 4, and allows for many different people to easily design their own experiments. These factors keep the GUI simple enough for use, but complex enough for expansion.

My GUI design makes the process of creating and outfitting your own experiments quick and easy for everyone of all ages, from engineers to elementary school students. This design includes large and clear fonts and colors, labels, icons, etc. It includes dropdown menus with simple and clear buttons, and instructions.



Snapshot of the Home Page

Using HTML, CSS/Bootstrap, Javascript, and Python, I am developing the GUI for offline use. This is through a Python library called “Eel.” I am also using SQLite3 for storage of experiment data. The diagram to the right shows how Eel works in the simplest way. Javascript is used as the middle-man between front and back end. This allows for use of languages HTML, CSS and libraries like bootstrap to aid in designing the GUI, while having every capabilities that Python provides.

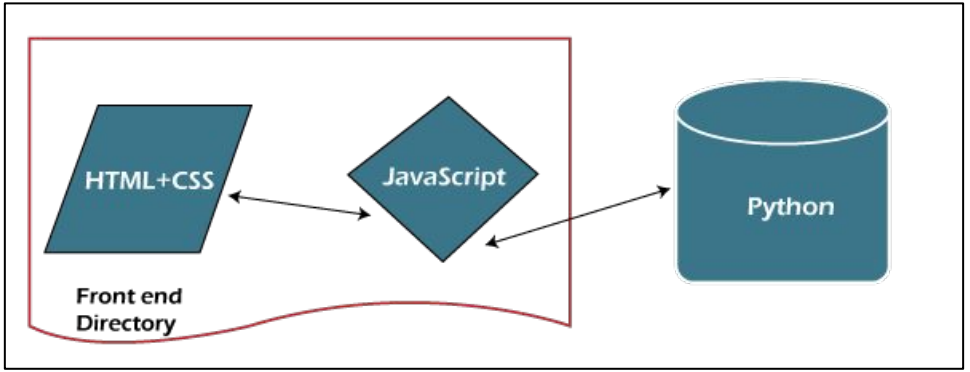
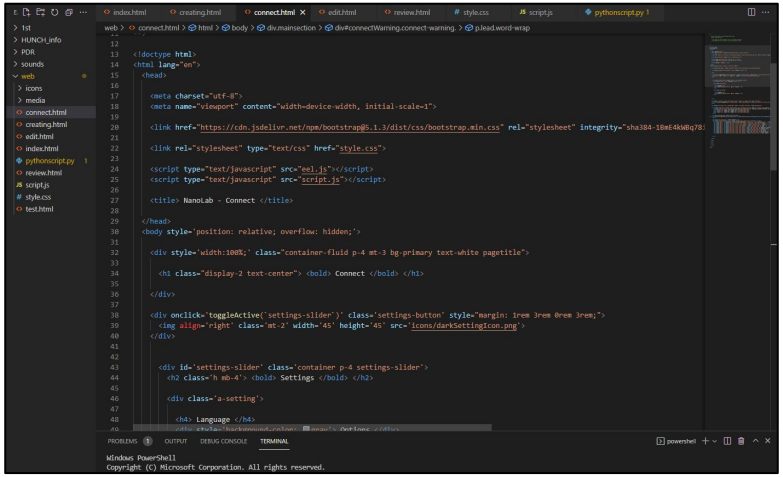


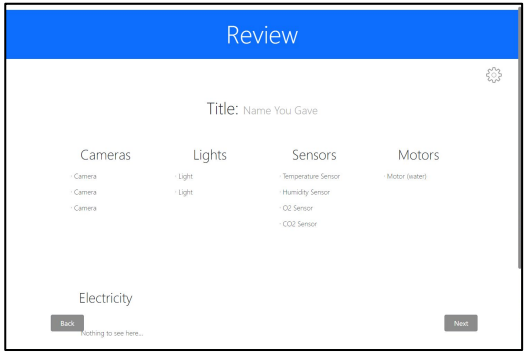
Diagram of how the Eel library functions



Snapshot of HTML behind the GUI

Main Features:

- Simple, minimalistic UI
- Easily navigate through designing your experiment
- Clear and concise
- Settings Bar to edit preferences
- Generic and expandable design
- Easily edit your experiment through easy to understand interface



Snapshot of the Review Page