

# PVMCF Data Explorer

CTSD Brent Thompson

Summer 2025

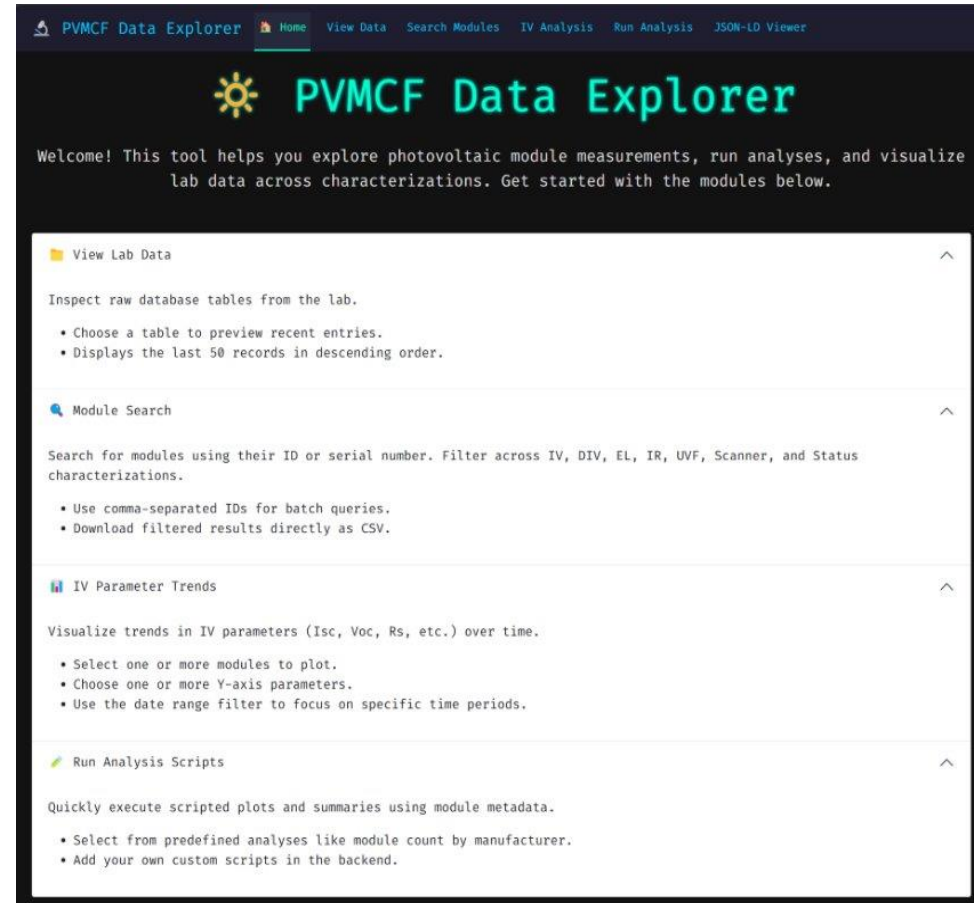
CEN 4930 Sem in Advanced Software Development

Professor Corey Leong



# About the Project

- Full stack software for EDA in photovoltaic research.
- SQLite database backend for speed and reliability.
- Reactive Shiny Python front end for viewing module data.



# Languages and tools

## Local Databases

- SQLite Databases indexing local file system.
- Primary keys of Module ID and measurement ID used to track modules.
- Backed up to Postgres each night.

## Python Classes

- Libraries used include; shiny, pandas, SQLAlchemy, matplotlib, logging, io, os.
- Custom python classes for database connectivity and automation.
- Workflows as python scripts for curating data and visualizations.

# Implementation

- Project is implemented using the UV python package manager.
- The software is designed to run off the database located at FSEC.
- The program is used through the local web browser.
- Integrating the Postgres database from the cloud was considered.
- Local databases were used to ensure speed and reactivity.

# Findings and Results

- Generating experimental datasets can be done 99% faster than prior methods.
- Human error is reduced to only the module-id selection step.
- Decreased technical burden and speed for plotting IV results of modules by over 95%.
- Identifying outliers and bad measurements occurs much faster.
- Graphs for studies can be produced instantly, alongside the measurements.

# Technical Challenges

- Using Postgres to retrieve data slowed down the application, and prevented reactivity for large datasets(VCAD)
  - Firewall/Network problems exist at the organization level.
- Data pulled from Postgres has no PII, which is needed to perform make/model level data science.
  - Technical complexity to rejoin cleaned data was not justified for a locally running program collecting the raw data.
- Python versioning and dependency issues.
  - Lab computer is running python 3.7
  - Solved using virtual environments.

# Successes and Disappointments

- Successes
  - Having the ability to plot as measurements are preformed, without any knowledge of plotting or data management.
  - Identifying outliers that exist in a graph and visualizing when the occur.
- Disappointments
  - Not being able to integrate Postgres in an efficient manner.
  - Missing plotting features for identifying individual points on the graph.

# Project Do Over

- More time would have been spent in the automated visualization.
  - Features for single point selection.
  - Option to select all from a model.
- Exploration of Python front end frameworks Streamlit / Marimo.
- Utilization of a complied version of python to create an exe.
- Starting sooner and having milestones completed earlier.



# Thank you

- Thank you viewers for time
- Thank you professor for direction
- Thank you MDS3 for funding the work



[Florida-Solar-Energy-Center/PVMCF-Data-Explorer: Bespoke data curation tool for the Photovoltaic Module Characterization Facility](#)