

The Schneider Prize for Technology Innovation



(L-R) Frank Schneider, US Army; Dr. Tony Schneider, US Army Air Corps; Dr. Henry Schneider, US Marines; Edward A. Schneider, Jr, US Navy

The Schneider Prize for Technology Innovation was founded in 2017 to honor the accomplishments of the Schneider family of York County, PA. Four brothers from York enlisted in the four branches of the military during World War II, while their sisters and parents served at home. The brothers returned to become scientists, engineers and civil servants, making their mark on our community. This prize honors their family's achievements in our community and will be awarded to a Hack that results in ideas and innovations for communicating and educating about specific technology.

In 2018, the prize was offered to the developer or team of developers that could improve upon the use of our current Grease Thief® technology. The winning team shared the \$1000 prize for their coding design and solution for creation of a criteria data set that generated actions based on lubricant color changes. This led to our current patented Grease Thief Colorimeter that customers can use on-site to analyze and quantify color changes in grease.

In 2021, the Schneider Prize addressed using a Raspberry Pi to develop a program to connect and collect data from multiple devices (such as sensors, scales, barcode scanners, any student designed device, etc) to be uploaded to a website via a HTTP request. The winning entry was able to demonstrate a design linked to their previous work as an intern in the York College Manufacturing Fellowship grant project.

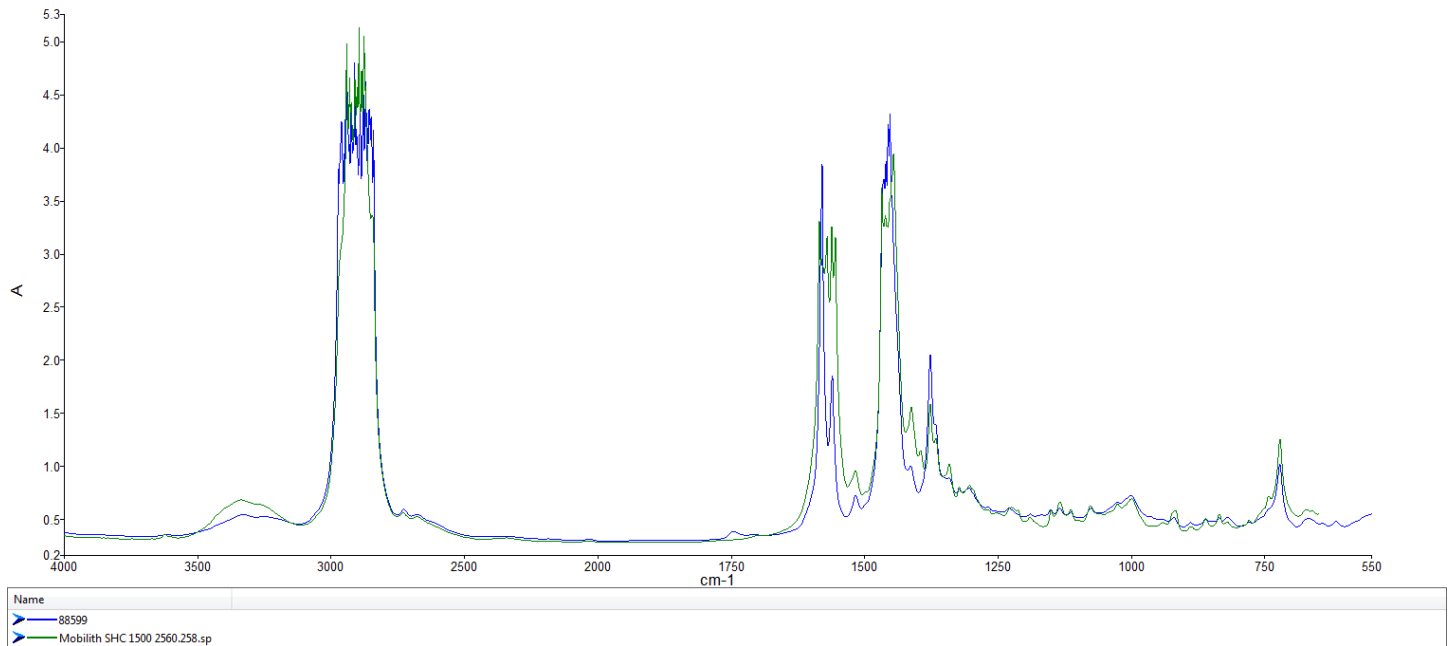
The challenge of the 2025 Schneider Prize for Technology Innovation will be:

Create a software solution that can take a baseline and multiple sample CSV files as input, superimpose the baseline CSV with x number of sample CSVs onto x total graphs, and batch save the generated graphs to a desired location on the file path. More information and instructions about this are offered on the back of this page.

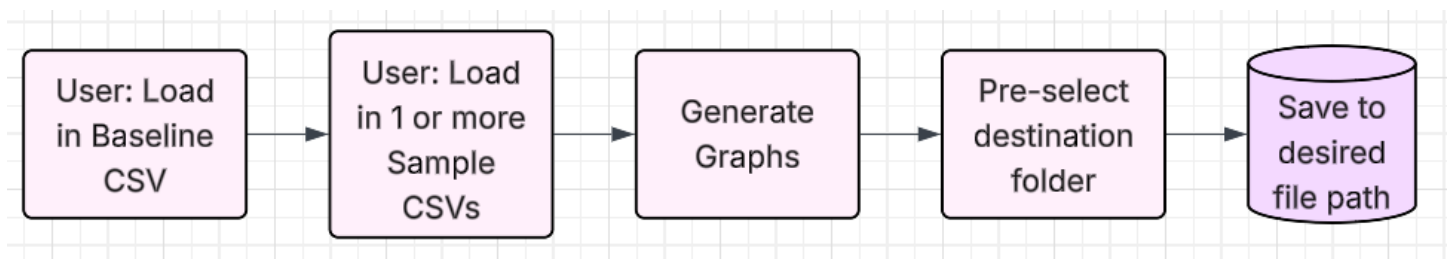
MRG Labs designs instrument solutions for deployment in analytical laboratories and for end-users as mini-lab solutions, allowing the evaluation of greases sampled from machinery to determine grease condition, contamination levels, and equipment health. As a part of this process devices are often integrated into forms of automation that must receive and handle data output from these devices. Currently York College is collaborating with MRG Labs for a Manufacturing Fellowship funded by a Pennsylvania state grant. This Fellowship builds on existing efforts to automate laboratory operations, namely the Grease Monkey, a robotic setup that scans, masses, assesses ferrous debris, measures color, and uploads the data to a website, MRG LIMS. Each team's solution will be judged on design, execution, and creativity. The team that makes the best application will receive a \$1000 Grand Prize. If a second worthy submittal is received, a Reserve Prize of \$500 will also be awarded.

More information can be provided and questions can be answered in a brainstorming session given by JD Brown Center for Entrepreneurship Center member MRG Labs during the Hackathon. Insight will be provided by Rich Wurzbach (President), Andrew Mott (Reliability Engineer) and Dylan Kletzing (Lab Manager). Good Luck to all the Hackathon participants!

Example final graph:



Program Flow chart:



Other Requirements:

- Input:
 - Baseline
 - Line graph: Green
 - Only one baseline is loaded at a time
 - When a baseline is loaded in, it should overwrite the last baseline, not append it to a list
 - Sample
 - Line graph: Blue
 - Samples should be designed such that you can load in multiple at a time
 - Each time you load a set in, it should overwrite the last set
- Output:
 - User must have the ability to decide where the images are saved
 - Images must be saved in .png or .jpg formats
 - Graphs must have a legend that shows the file name of the loaded in Baseline and Sample greases (similar to what's shown in the above graph)
- Extra Challenge:
 - It would be nice to be able to have a graph generated on the GUI when a baseline is loaded in, then when a batch of samples are loaded in, the user can select between samples and have the graph change with the selected sample in real time.
 - With this addition, we still need to be able to batch save all graphs in one go, regardless of what is being shown on the GUI.