DUONG BAO KHANG

1. CONTACT INFO

LinkedIn: www.linkedin.com/in/khang-duong-6684a7223

Facebook: Facebook

Email: khang.2003c@gmail.com

Phone number: (+84) 778 718 209

2. INTERNSHIP TECHNICAL EXPERIENCE

A. EXCEL AUTOMATION AND DATA ANALYSIS WITH PYTHON (BAKER HUGHES)

- Summary: write Python scripts to speed up the team's manual workflow when dealing with Excel files.
- Tools: Python (Pandas, OpenPyxl, re)
- Details:
    - Designed modular backend logic using Python functions to extract, validate, and transform Excel data into structured outputs. Implemented regex-based anomaly detection and data correction logic as part of a reusable validation pipeline.
    - Compared two 80,000+ row datasets to detect version differences (e.g., MODBUS IO Schedule ver000 vs ver001).
    - Engineered a structured backend process to parse, merge, and validate datasets from multiple sources into a unified Excel file. Included logic for duplicate removal, field matching, and metadata tagging for downstream analysis.

B. TASK AUTOMATION WITH PYTHON (BAKER HUGHES)

- Summary: write Python scripts to automate the machine resources logging task for further analysis
- Tools: Python (Pandas, OpenPyxl, Seaborn, Matplotlib)
- Details:
    - Built a data pipeline that logs system performance metrics (CPU, RAM, network) and feeds them into a backend logic layer that performs aggregation, anomaly detection, and time-series transformation before visualization.
    - Visualized system performance metrics to detect spikes and resource saturation.
    - Automated startup behavior using Bash scripts and packaged Python executables.

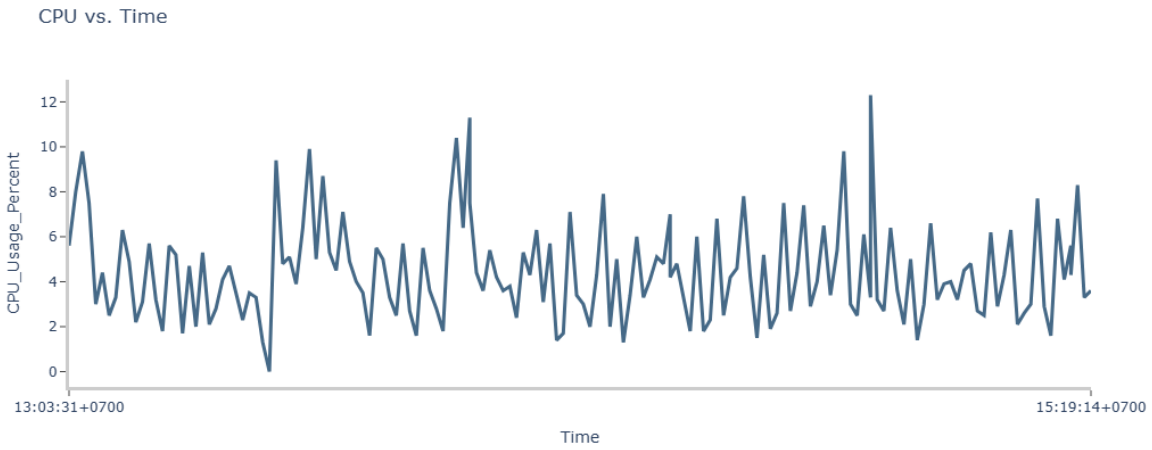

Figure 1. Example output from the data visualization task.

C. WEB SCRAPING, USER INTERFACE DESIGN, AND DATA ANALYSIS WITH PYTHON (INTEL)

- Summary: design a user interface to execute the Python scripts that automate the data collection and processing work.
- Tools: Python (Pandas, OpenPyxl, Selenium, Matplotlib, bs4, json, numpy, PrettyTable, xlsxwriter, playwright)
- Details:
    - Separated frontend (PyQt5 GUI) from backend logic modules handling web scraping, parsing (HTML/JSON), and Excel generation. Ensured loose coupling and clear input-output definitions for maintainability and scalability.
    - Implemented Selenium-based bots to extract and screenshot data from complex web pages.
    - Parsed HTML (BeautifulSoup) processed JSON/CSV data and exported outputs to Excel using xlsxwriter and PrettyTable.

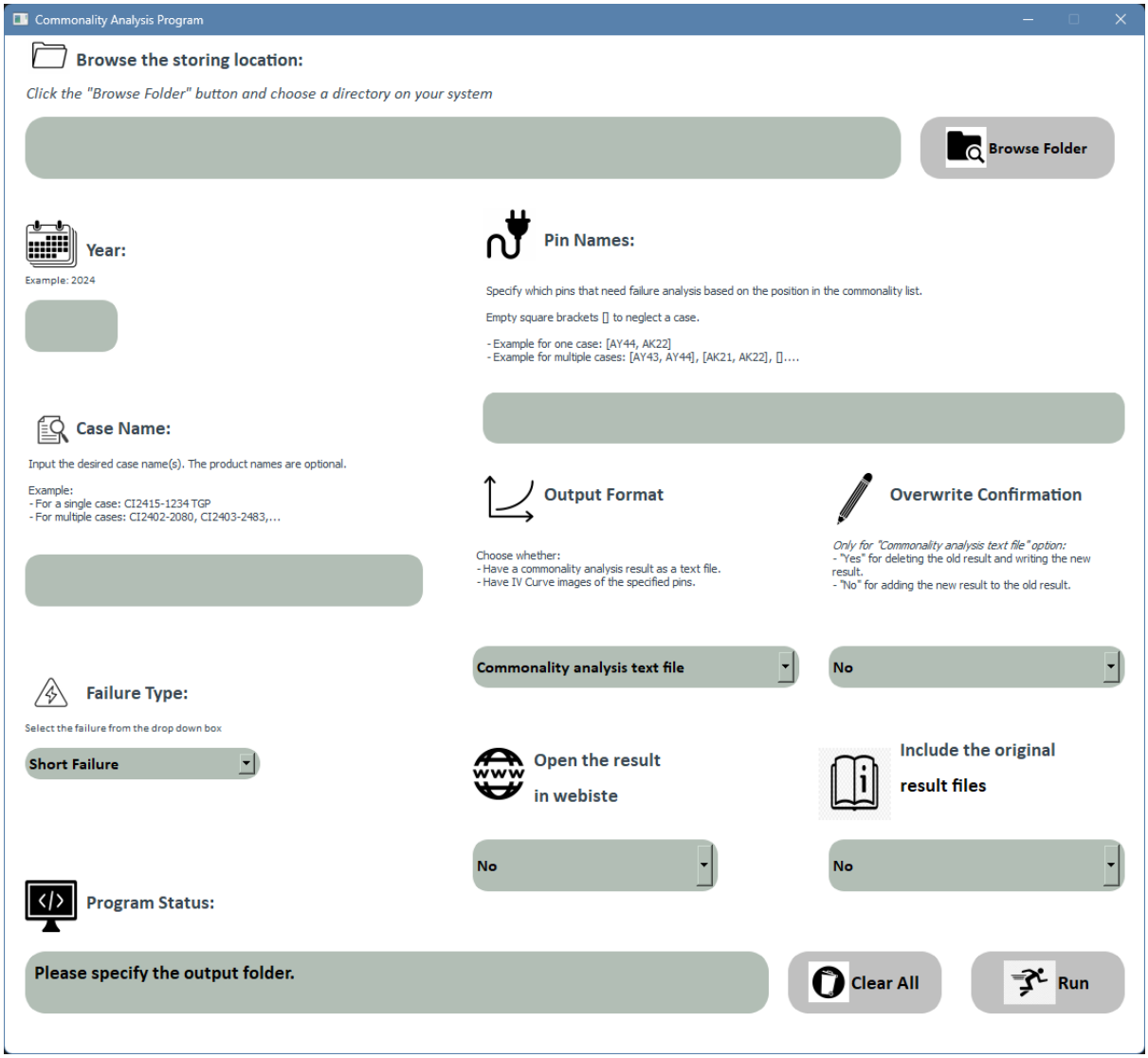

Figure 2. User Interface developed at Intel using PyQT5.

C. PLC PROGRAMMING AND TESTING (BAKER HUGHES)

- Summary: update and test PLC code
- Tools: Studio 5000 (Allen-Bradley PLC), and TwinCat (Beckhoff PLC)
- Details:
    - Updated logic in Allen-Bradley PLC using Studio 5000.
    - Executed and validated test cases on Studio 5000 and TwinCAT platforms.

3. UNIVERSITY PROJECTS

A. FAILURE PREDICTION DATA PROCESSING AND MACHINE LEARNING

- Summary: write Python scripts to speed up the team's manual workflow when dealing with Excel files.
- Tools: Python (Pandas, OpenPyxl, re)
- Details:

- Applied full preprocessing pipeline: import libraries, read dataset, sanity check, exploratory data analysis, missing value treatment, outliers' treatment, duplicates and garbage value treatment, normalization, encoding of data.
- Construct a Random Forest model to predict the robotics arm failure with more than 90% accuracy.
- Evaluated model performance with confusion matrix, learning curves, and validation sets.

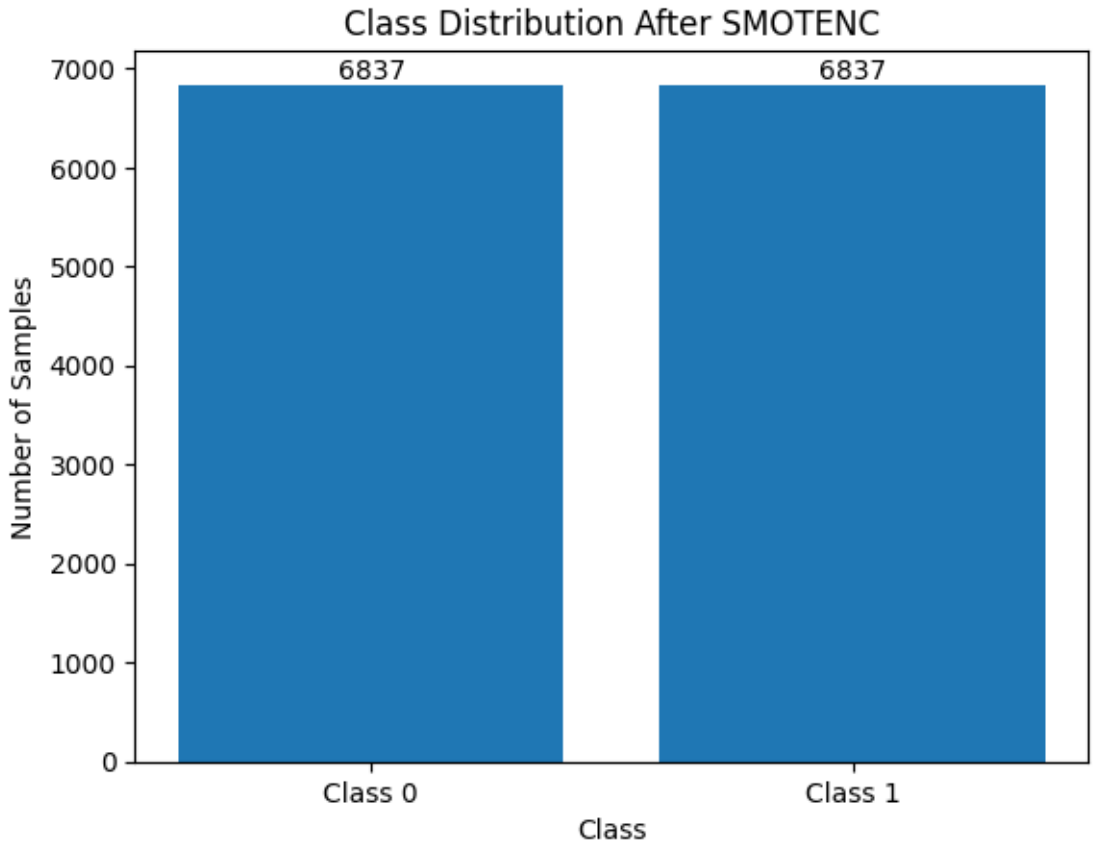

Figure 3. Balancing the dataset.

```
[ ]  import numpy as np
     # Calculate class weights (inversely proportional to class frequencies)
     class_weights = {0: 1, 1: np.mean(y) / (1 - np.mean(y))}  # Example: Assuming class 0 is 0 and class 1 is 1
     rf_model = RandomForestClassifier(n_estimators=200, #200
                                       random_state=42,
                                       max_features='sqrt',
                                       max_depth=20, #10
                                       min_samples_leaf=50,
                                       class_weight='balanced'
                                       )
```

Figure 4. Machine learning model parameters.

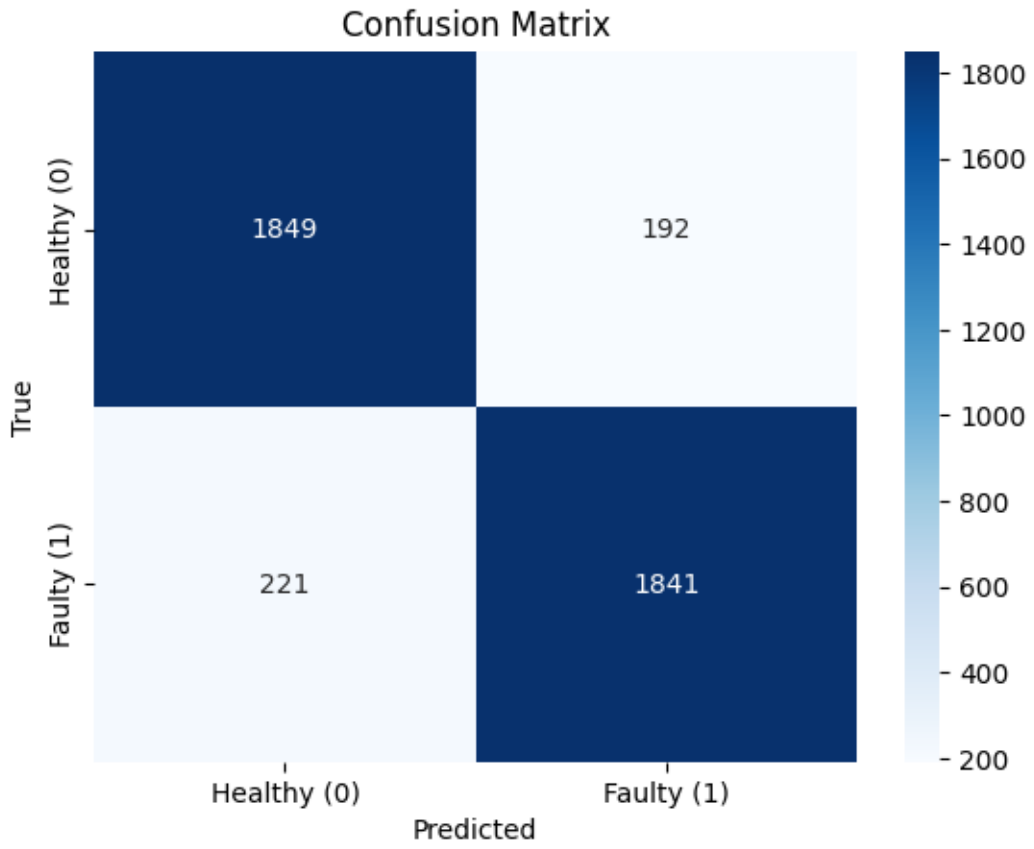

Figure 5. Confusion Matrix.

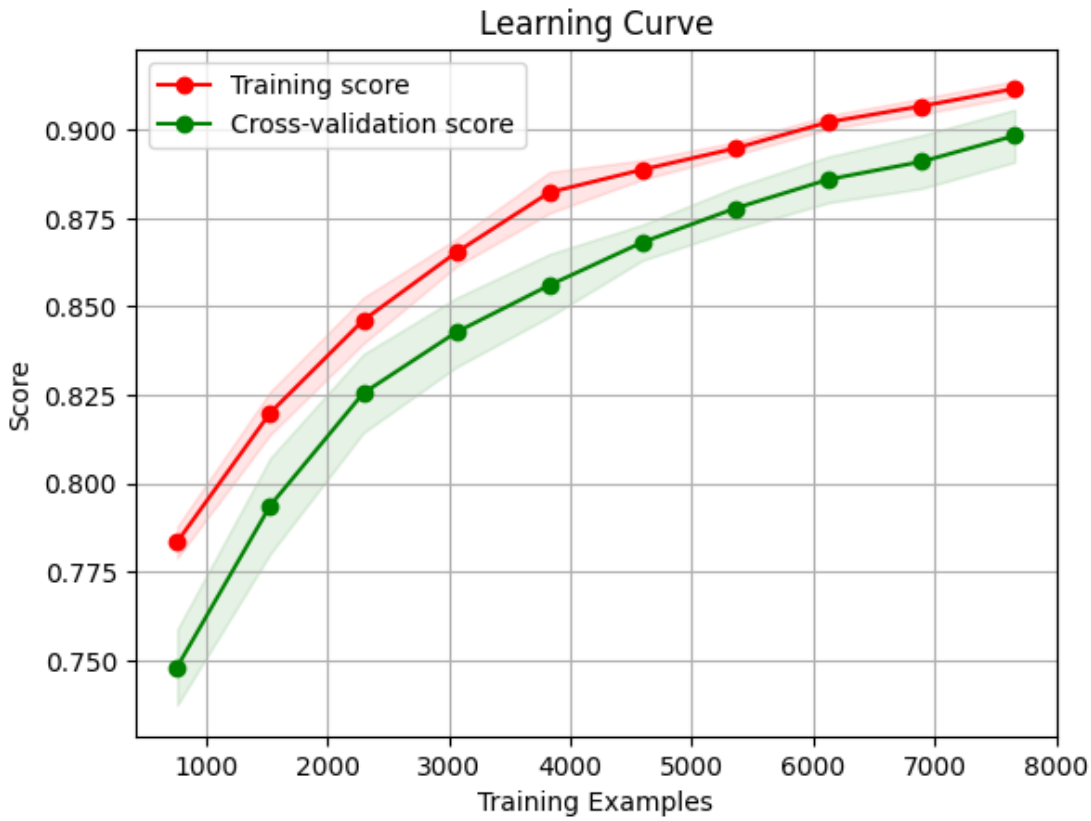

Figure 6. Learning Curve.

B. FULL-STACK WEBSITE TO REMOTELY CONTROL THE HARDWARE

- Summary: program the NUC140 board with C/C++ at the register level
- Tools: Python FastAPI, websockets
- Details:
    - Built frontend using HTML/CSS/JS and served it via public tunnel (Cloudflared + Uvicorn).
    - Enabled bidirectional real-time communication via websockets for device control.

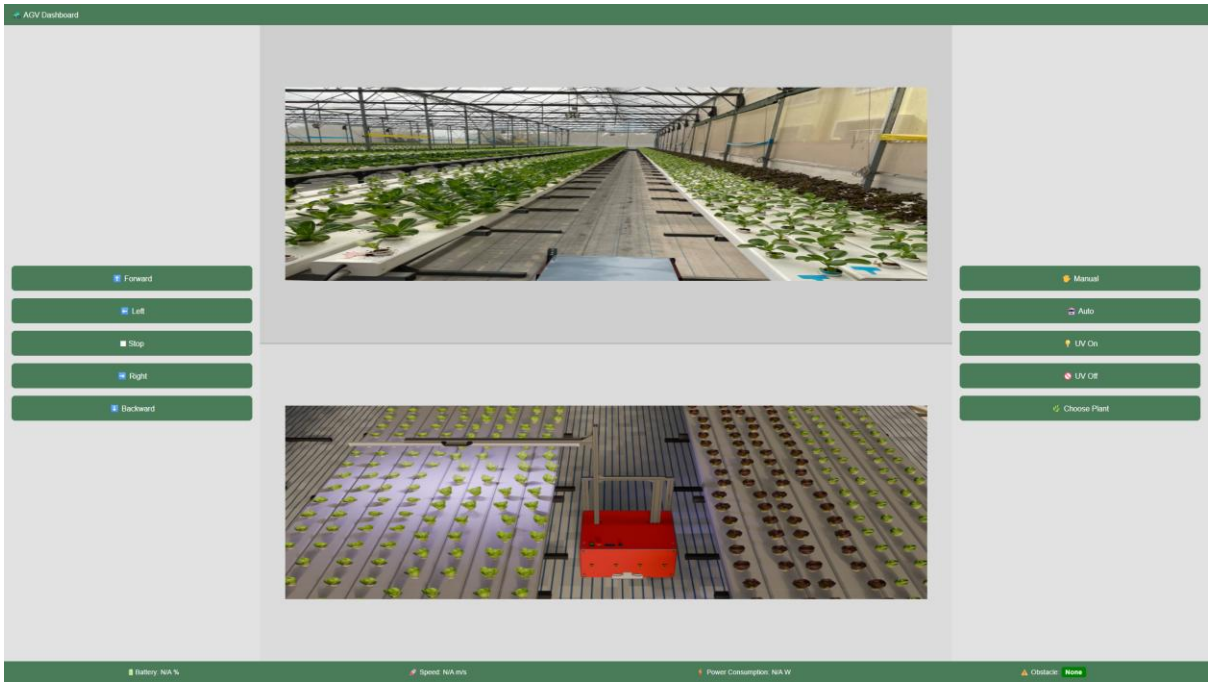

Figure 7. Website frontend.

C. SCADA SYSTEM DESIGN WITH SIEMENS ECOSYSTEM

- Summary: develop HMI/PLC for a material handling problem.
- Tools: Siemens TIA Portal v19.0, WinCC, S7-1200 PLC, Siemens KTP700 HMI
- Details:
    - Programmed ladder logic using IEC 61131-3 standards to control conveyor belts, sensors, and actuators under varying simulated loads.
    - Integrated fault-tolerant logic to handle sensor failures and emergency stop conditions, improving uptime simulation reliability.
    - Designed intuitive HMI screens with real-time feedback on system status (motor speed, load weight, error messages).
    - Configured Ethernet-based communication between HMI and PLC using Profinet for reliable data exchange.
    - Enabled alarm logging and diagnostic messages to reduce manual monitoring needs and improve response times in error scenarios.

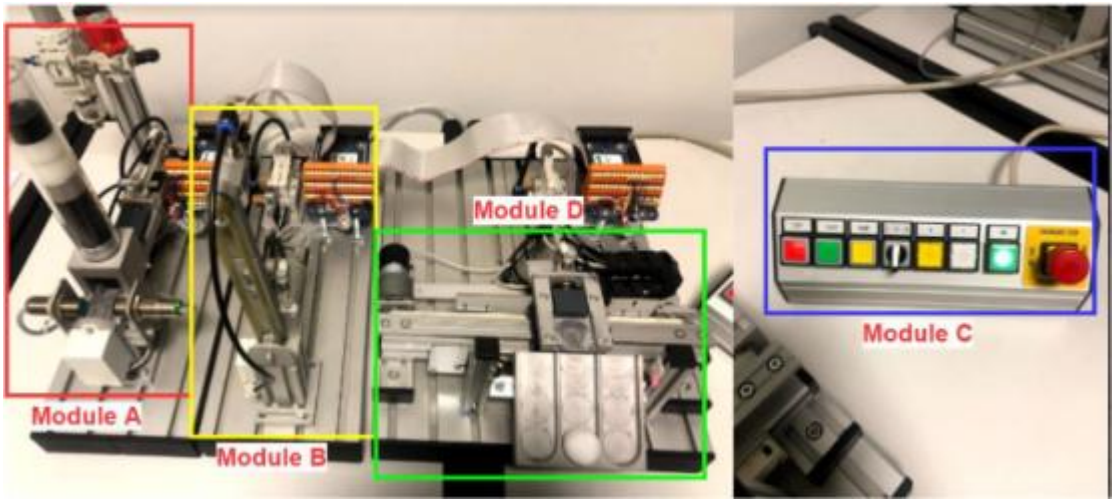

Figure 8. Siemens System.

D. C/C++ PROGRAMMING IN THE NUC140 BOARD

- Summary: program the NUC140 board with C/C++ at the register level
- Tools: C/C++, Keil v5
- Details:
  - Implemented drivers for GPIO, UART, SPI, LCD, timers, and clock systems.
  - Processed serial data and handled bit-level operations.

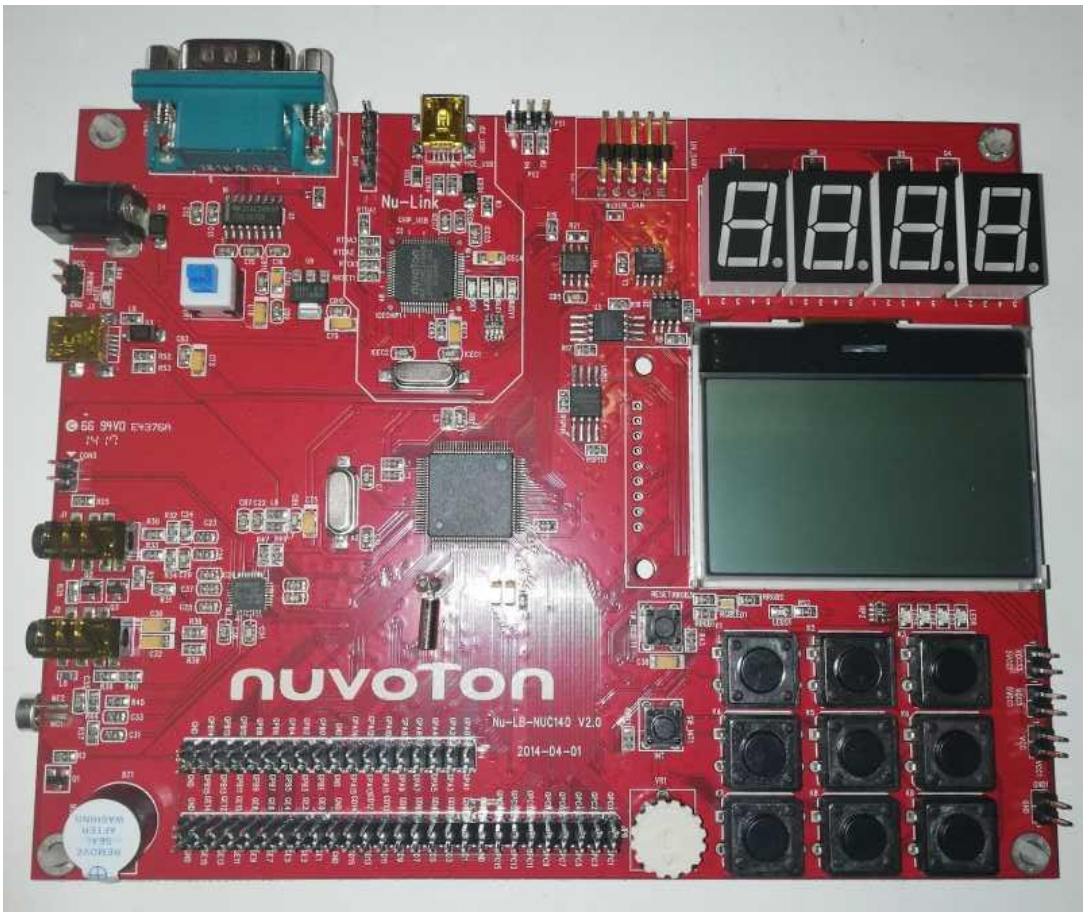

Figure 9. NUC140 board.

E. WIND TURBINE DESIGN

- Summary: design a wind turbine with detailed mechanical calculations for components selection
- Tools: SOLIDWORKS
- Details:
    - Performed mechanical stress analysis to size components (shaft diameter, torque loads).
    - Selected gears, bearings, and brakes based on catalogue specifications.
    - Modeled and drafted a complete 3D/2D prototype aligned with calculated requirements.

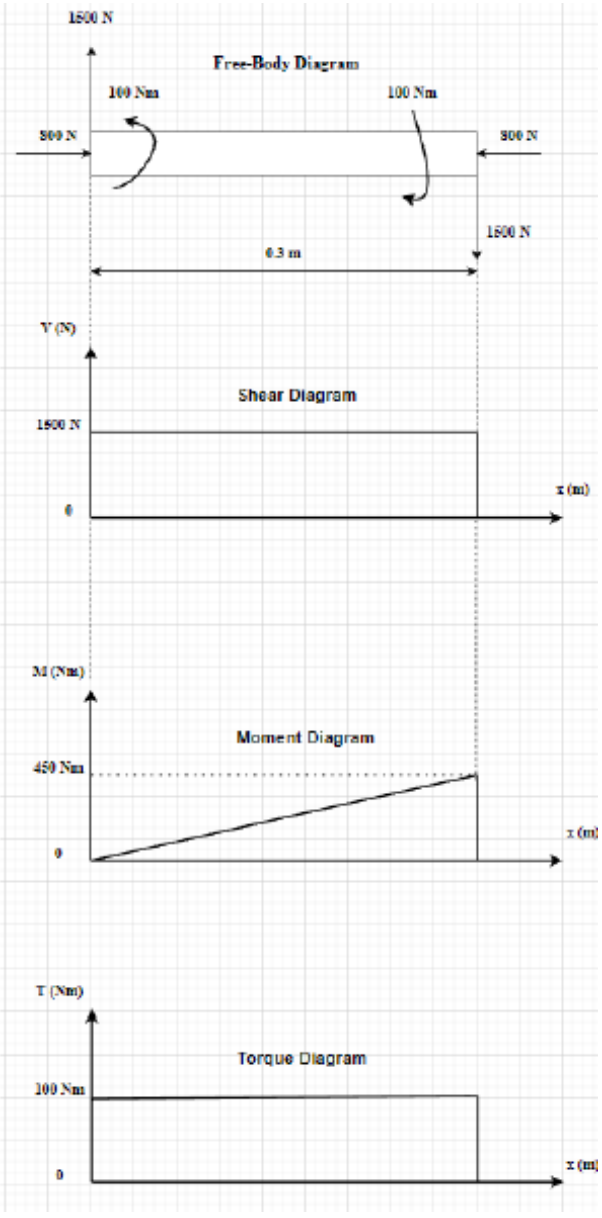

Figure 10. Free-body, Shear, Moment, and Torque Diagrams of the main shaft.

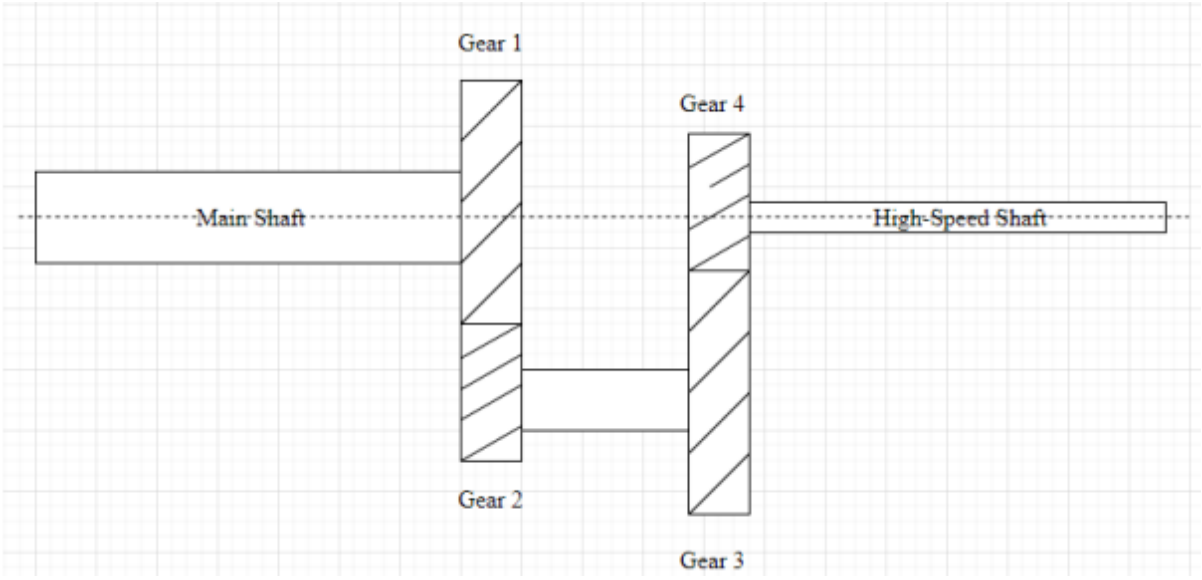

Figure 11. Gearbox internal layout.

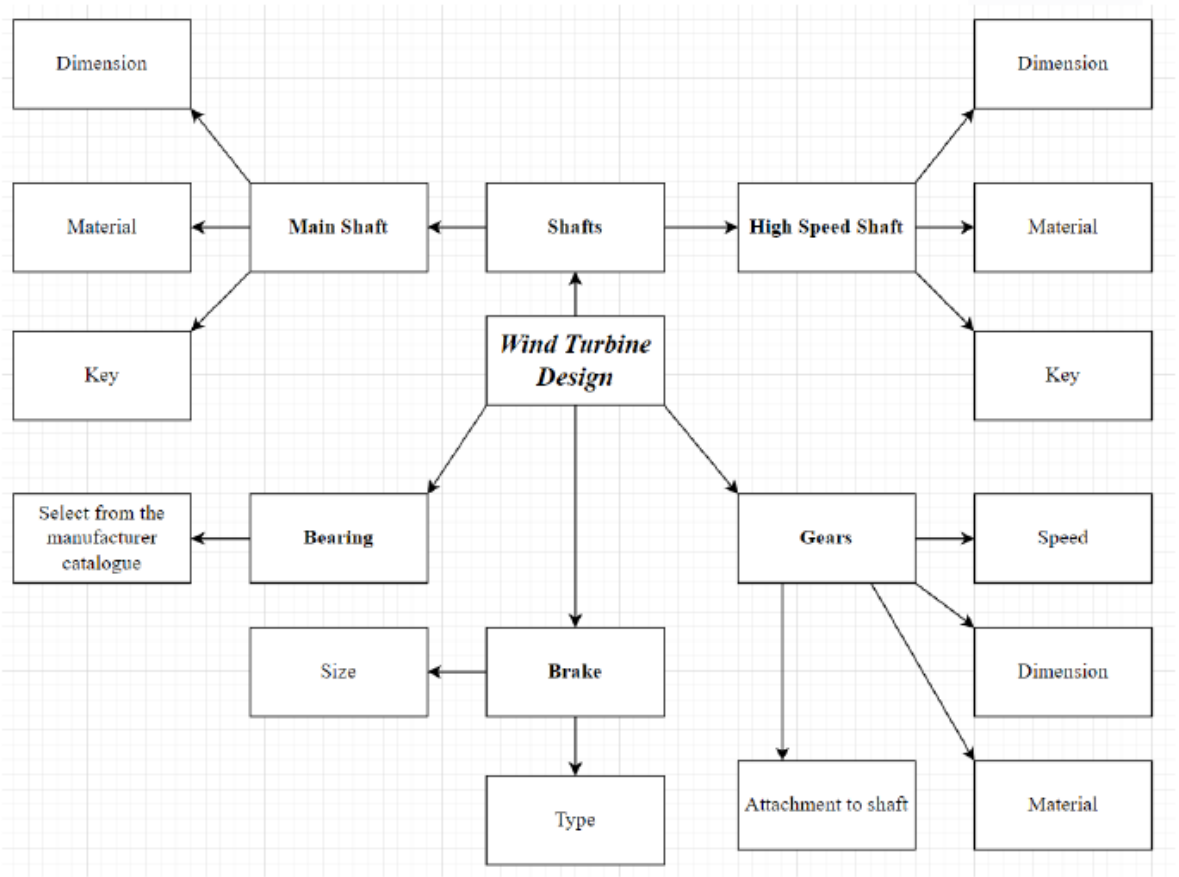

Figure 12. Main components and their design parameters of the turbine system diagram.

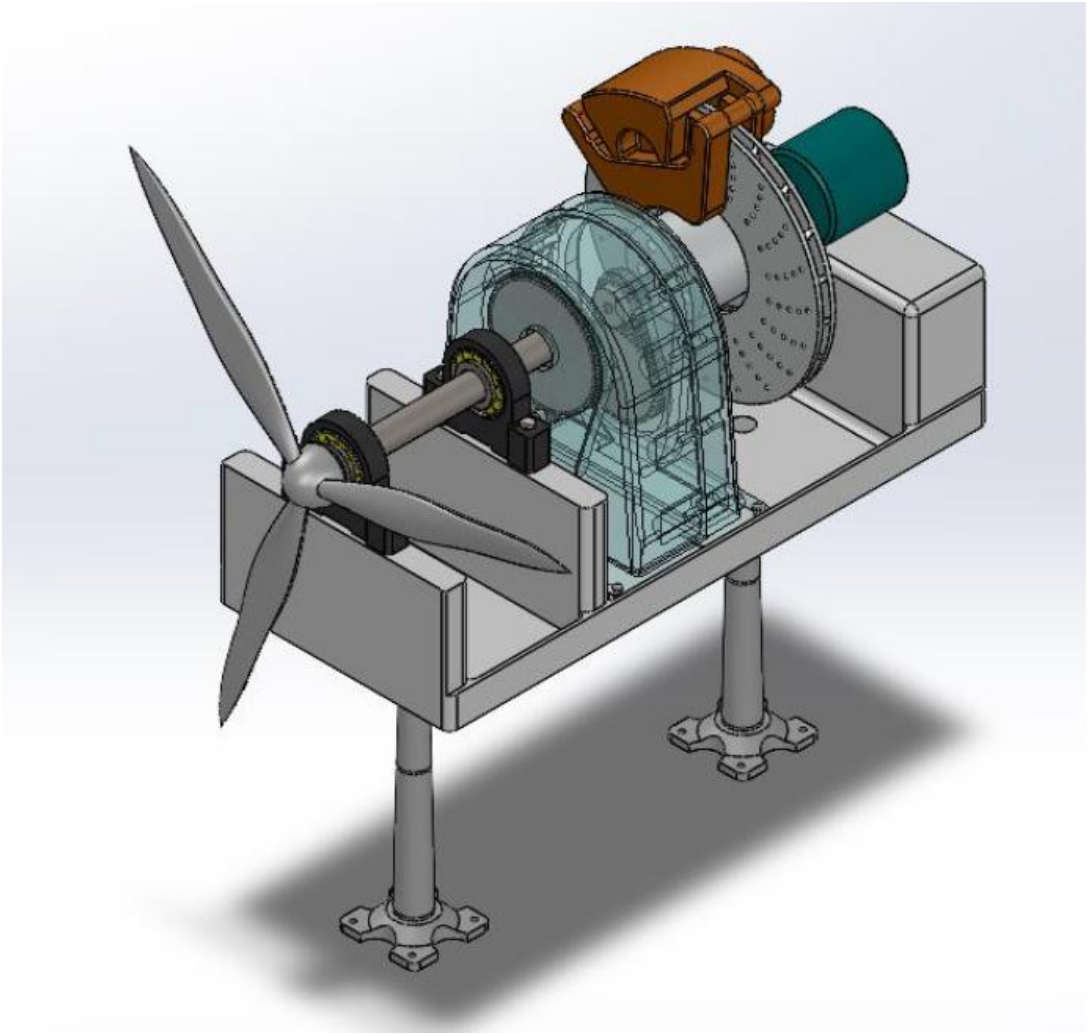

Figure 13. Wind Turbine 3D model.

F. BATTERY-POWER VEHICLE

- Summary: developed a lightweight 3D-printed electric vehicle under manufacturing constraints.
- Tools: SOLIDWORKS
- Details:
    - Designed for 3D-printing using resin with adherence to DFM principles.
    - Managed bill of materials, 3D printing logistics, and project documentation.
    - Performed transmission calculations and developed mechanical drawings.

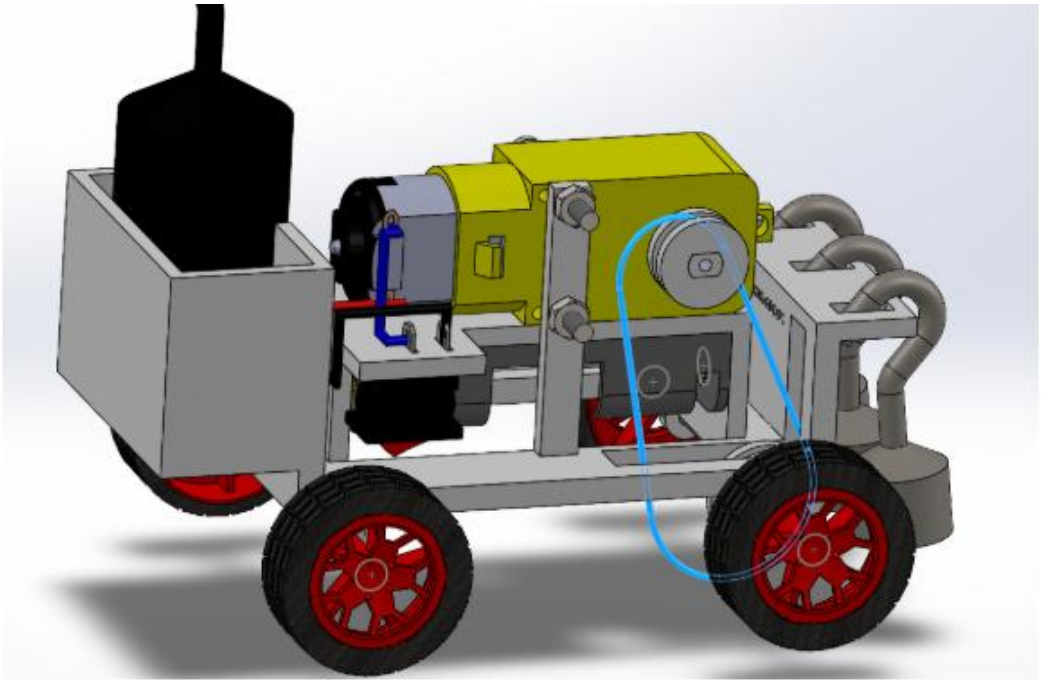

Figure 14. 3D model of the vehicle.

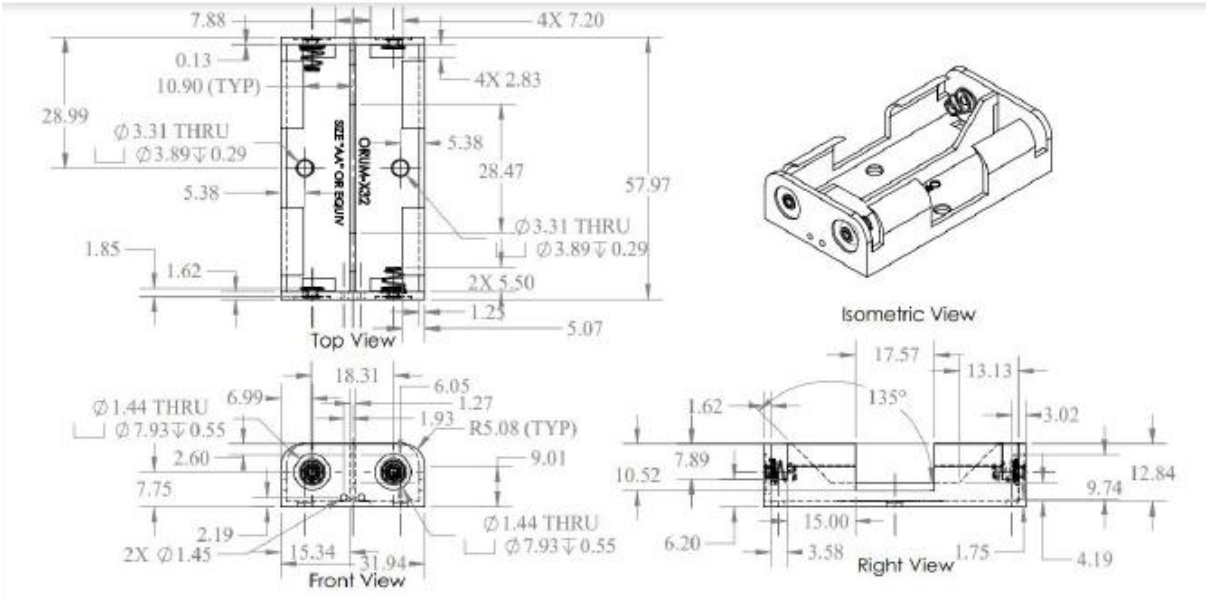

Figure 15. 2D drawing of a battery holder.

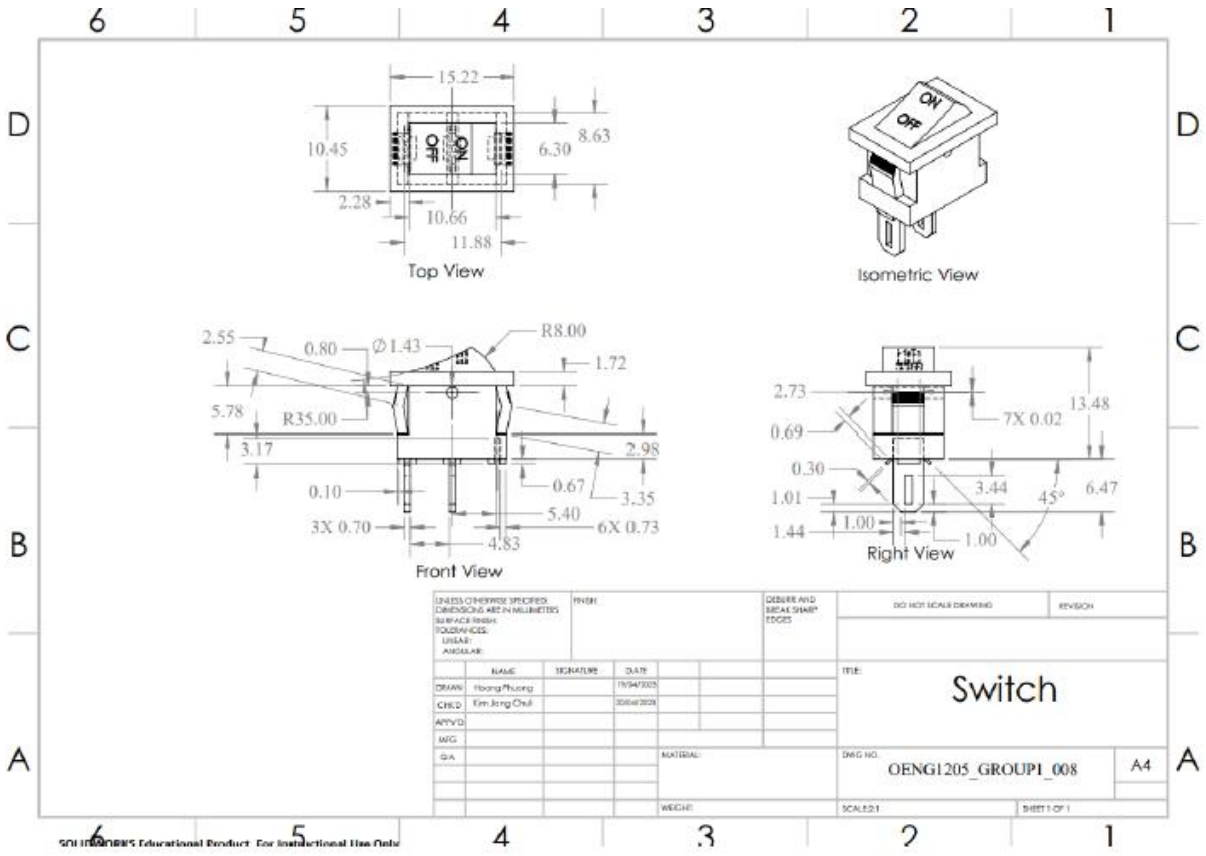

Figure 16. 2D drawing of a switch.

F. STRESS ANALYSIS AND BRIDGE DESIGN

- Summary: Designed a bridge structure under dynamic loading conditions.
- Tools: SOLIDWORKS
- Details:
    - Conducted stress simulations to derive optimal cross-section geometries.
    - Designed for manufacturability with plastic laser cutting.
    - Assembled fabricated parts and verified integrity under simulated loads.

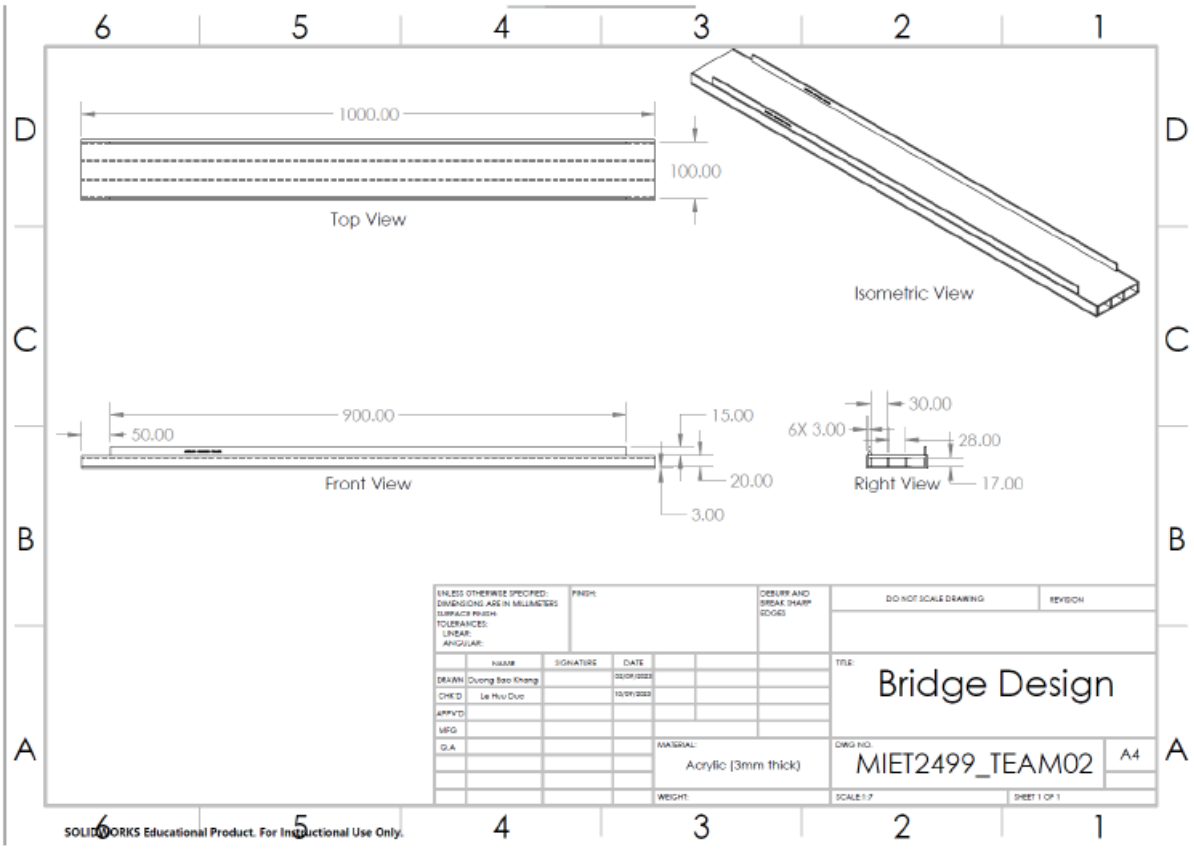

Figure 17. Bridge design.

G. VOLTAGE REGULATOR PCB DESIGN

- Summary: Designed a bridge structure under dynamic loading conditions.
- Tools: Multisim, Cadence
- Details:
    - Simulated circuit behavior under varying input conditions using **NI Multisim** to verify functional integrity and transient response.
    - Designed PCB layout in **Cadence PCB Editor**, including component footprint placement, trace routing, and netlist verification.
    - Assembled, soldered, and tested the physical circuit using the **NI ELVIS board** and digital multimeter to validate voltage readings, continuity, and response accuracy under test loads.

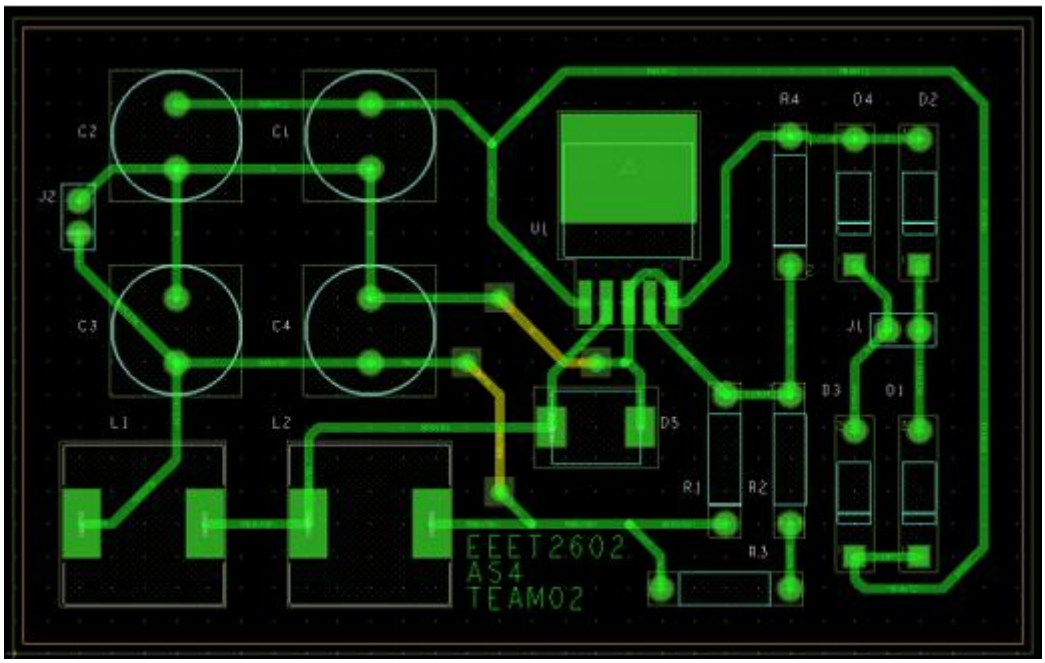

Figure 18. PCB Layout in Cadence PCB Editor design.

Figure 19. Completed PCB looking from downside to upside of TOP view.

4. CERTIFICATIONS & TOOLS

- **Microsoft Excel** – Advanced functions, Pivot Tables
- **Python for Data Science** – Strong command of data cleaning, modeling, and visualization
- **SQL** – Basic querying and data manipulation
- **C/C++** – Embedded systems, microcontroller development
- **Power BI** – Familiarity through project integration
- **SOLIDWORKS, AutoCAD, Cadence -** Familiarity through project integration
- PyQt5, Selenium, Pandas, Matplotlib, WebSockets, REST APIs

5. SOFT SKILLS / ACHIEVEMENTS

- **Soft Skills**: Cross-functional collaboration, technical documentation, time management, adaptability

- **Achievement**: RMIT Vietnam Academic Achievement Scholarship (2025) – GPA: 3.9 / 4.0
- **Additional certifications**: Industrial Electrical Control Panel Designing and Installation Course (Issued Oct 2023) at PLCTECH Automation Training Center.

Figure 20. Industrial Electrical Control Panel Designing and Installation Course.

- **Extracurricular activities**:
  - Bronze Volunteer Recognition
  - SSCC Committee Certificate
  - Project Management Acknowledgment

Figure 21. Bronze Certificate for being an enthusiastic volunteer.

Figure 22. Student-Staff Consultative Committee (SSCC) Certificate.

Figure 23. Recognition of contribution as a Project Manager.