**Prototype Phase**

1. Students will use Wireshark to capture real-time network traffic data, exporting the captured data to a CSV file for further analysis. The capture will include features such as protocol types, IP addresses, packet sizes, time intervals, and more.
2. Students will preprocess the captured data by cleaning it, encoding categorical variables (e.g., protocol types), and normalizing numerical features (e.g., packet sizes, time intervals).
3. The processed data will then be structured into a CSV dataset, including labels for normal and malicious traffic activities, to be used for training the machine learning models.
4. Students will explore various machine learning classification techniques and select advanced ensemble models, including TabNet, CatBoost, LightGBM, and stacking ensembles. These models will be trained on the preprocessed network traffic dataset.
5. Model evaluation will be performed using state-of-the-art classification metrics such as accuracy, precision, recall, F1-score, and ROC-AUC.

**Tools:**

* Programming Language: Python
* Traffic Capture Tool: Wireshark for capturing network traffic data in real-time and exporting it to a CSV file for analysis.

<https://www.wireshark.org/download.html>

* Machine Learning Libraries: Scikit-learn, XGBoost, LightGBM, pytorch-tabnet, CatBoost

https://github.com/

* Web Development Frameworks: Python Flask or Django for building the web application
* Operating System: Any (e.g., Windows, Linux).
* Additional Tools: Jupyter Notebook for data analysis and model training, Matplotlib/Seaborn for data visualization.