**Machine Learning Project Documentation**

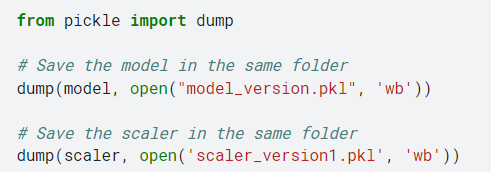
**Deployment**

**1. Overview**

In the deployment phase of machine learning, the trained model undergoes a crucial transformation for real-world utilization. Initially, the model is serialized into a transportable format and seamlessly integrated into the production environment. Infrastructure is set up to support the model's operations, considering scalability, performance, and reliability. An API may be developed to facilitate interaction with other applications, and robust monitoring and logging mechanisms are implemented for real-time performance tracking. Security measures, scalability testing, documentation, user acceptance testing, and the establishment of a continuous integration/deployment pipeline contribute to a comprehensive deployment process, ensuring the model's effective and secure functionality in a production environment.

**2. Model Serialization**

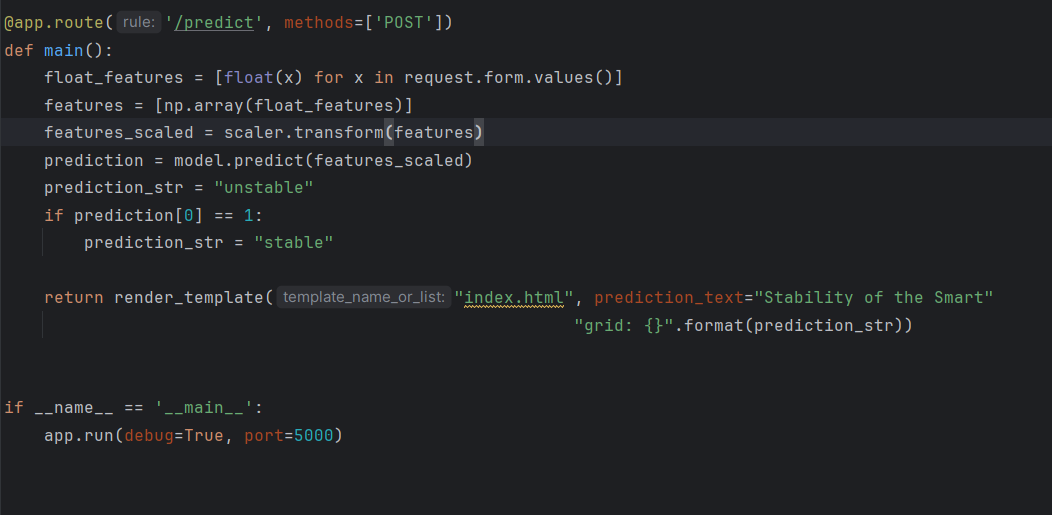
For our model deployment we faced a serious problem to import Tensorflow and keras libraries. So, we were obliged to use our second model of classification based on xgboost. That model has almost the same average with our Deep Learning model.



By using pickle, we create a model and scaler folder to be used in our flask web server.

**3. Model Serving**

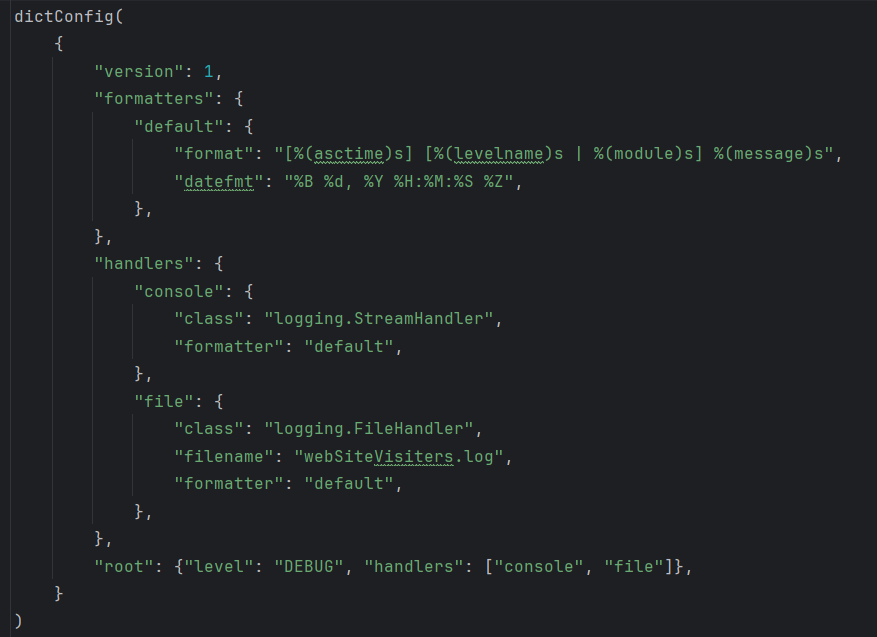
We used Flask which a comprehensive, and user-friendly tool for ML model deployment. Our solution is mainly on-premises solution.



**4. API Integration**

We used Rest API through Flask for our deployment. Inputs are from a form then convert in an array with numpy, scaled with our scaler model download from the prediction code.

1. **Security Considerations**







In flask, we can configure a secret key. I set up one secret key to ensure the security of the connexion on the web app.

**6. Monitoring and Logging**

Our model will be moniroted with WHYLABS. There we will track features behavior and determine one determine most the stability of the smart grid