**Documentation Digital Twin (with updates)**

Contents

[1. Introduction 1](#_Toc177990331)

[2. High-level explanation of DT1 1](#_Toc177990332)

[3. Documentation of latest deliverables (D4.1) 1](#_Toc177990333)

[4. Technical explanation of DT code 4](#_Toc177990334)

[5. Current status & next steps 5](#_Toc177990335)

[6. Greek Demo-site 5](#_Toc177990336)

[7. WP4 5](#_Toc177990337)

# Introduction

As part of work package 4 (WP4) DTU is developing the first digital twin (DT1): Digital Twin for Real-time Network Congestion and Voltage Violations Identification and Management (DT-1). This digital twin will be developed for the Danish and Greek demo-site. The digital twin has now been developed for the digital twin of the Danish demo-site, since we have access to Danish demo-data. The Greek demo data has recently been uploaded, and a similar digital twin needs to be developed for the Greek demo site. DT1 is only part of the first demo-case (note that demo-cases are something different from demo sites). This digital twin is meant to be real-time and perform power flow calculation, screen contingency scenarios and validate the flexibility cleared by the market. In the next sections, the digital twin will be explained in more detail.

# High-level explanation of DT1

The digital twin consists of three engines:

* Real-time security assessment engine (RSAE)
* Contingency assessment engine (CAE)
* Security Management and Flexibility Activation Engine

Demo-case 1: The interaction and process of demo-case 1 can be found [here](https://txtgroup.sharepoint.com/:u:/r/sites/HESYNERGIES/_layouts/15/Doc.aspx?sourcedoc=%7B9EB11ACB-B1D1-429A-A5BE-5CAE4701C428%7D&file=WP4%20tasks%20elements%20relation.vsdx&action=default&mobileredirect=true), where we are involved in step 4 and 5. After ICCS sends the cleared flexibility from the market, we validate this flexibility (whether it does not cause any problems in the network) and send a signal back, saying for example that this percentage of flexibility can be activated (or ideally all of it). Additionally, we should check if it solves the occurring problems in the network (i.e. voltage violations, line/transformer overloadings).

A diagram of a process

Description automatically generated

*Figure 97: Real-time Security Assessment engine – Core interactions with other Components/Services*

A diagram of a company

Description automatically generated

*Figure 98: Contingency Assessment Engine – Core interactions with other Components/Services*

A diagram of a project

Description automatically generated with medium confidence

*Figure 99: Security Management and Flexibility Activation – Core interactions with other Components/Services*

# Documentation of latest deliverables (D4.1)

D4.1 (July 2024): [20240630\_ETRA\_SYNERGIES\_D4.1\_Release of the Energy Services, Applications and Marketplace\_pu\_v1.pdf](https://txtgroup.sharepoint.com/:b:/r/sites/HESYNERGIES/Shared%20Documents/05-Work%20Packages/WP4/Deliverables/D4.1/20240630_ETRA_SYNERGIES_D4.1_Release%20of%20the%20Energy%20Services,%20Applications%20and%20Marketplace_pu_v1.pdf?csf=1&web=1&e=qjMDA5)

The Digital Twin for Real-time Network Congestion and Voltage Violations Identification and Management provides a high-fidelity representation of the real-time status of the power distribution network. This digital twin is built based on highly granular real-time SCADA and smart meter measurements provided by the network operator to the energy data space. With the aid of different statistical tools and pre-trained machine learning algorithms defined, the digital twin facilitates steady state estimation and monitoring of the network security and stability. The high-quality data simulated from the digital twin facilitates the prediction of potential security violations, such as voltage security limit violations, line congestions and potential contingencies that can affect the network. Moreover, with the aid of a decision-making framework, the digital twin will support the output of the spatial and temporal flexibility margins and additional countermeasures where necessary to mitigate any security violations and contingencies during real-time operation.

To accomplish the functionalities indicated above, the digital twin consists of three engines:

1. A Real-time Security Assessment Engine (RSAE), to monitor the status and security of the network during real-time operation.

2. A Contingency Assessment Engine (CAE), to assess potential critical events, such as N-1 and N-2 scenarios i.e., scenarios involving the loss of one or two components of the power system such as line disconnection or generation failures based on the current operating conditions.

3. A Security Management and Flexibility Activation Engine (SMFAE), to determine the flexibility volumes and margins required to mitigate existing security violations in the network. In addition, the engine sends out flexibility activation requests to the aggregator based on the market clearance volumes and where needed provides additional measures to mitigate existing critical events.

4.1.2 Implemented Features

This section contains the features that have been implemented and partially implemented in the digital twin. The digital twin contains the three engines previously mentioned which monitor the real-time security status of the grid and identifies the upward and downward regulation necessary to mitigate security violations during real-time operation.

The table below shows to which extent different components of this digital twin have been deployed:

A white and black text on a white background

Description automatically generated

4.1.3 Technology Stack and Implementation Tools

The table below includes the versions and license of the technology used for this release of the digital twin. The digital twin is developed using Python 3.11.5.

A screenshot of a computer

Description automatically generated

4.1.4 Integration with the SYNERGIES Data Space

In the current release, the digital twin for Real-time Network Congestion and Voltage Violations Identification and Management has not yet been integrated with the SYNERGIES Energy Data Space. As uploading real data from the demo sites to the Energy Data Space is still in progress, the Digital Twin has been developed based on sample historical data obtained externally from the Danish demo site. Further calibration of the functions provided by the analytic models will ensue when more historical and real time data is available through the energy data space. In future releases, the energy services from the digital twin will be incorporated within the Energy Data Space.

4.1.5 Assumptions and Restrictions

The functionalities have been validated using sample and static data from the Danish demonstration site. Upon establishing the connection to the Data Space, enabling access to real-time data, these functionalities will be tested under real-time conditions. Following the establishment of this connection, real-time testing will also be conducted. The following data assets have been used for this release:

* Static network topology data from the Danish Demo site.
* Sample historical smart meter measurements with active power production and consumption at each node from the Danish demo site at a 15min interval for the year 2022.
* Sample historical SCADA measurements for the same period as the smart meter measurements from the Danish demo site (at a 15 min interval).

Currently, the integration of forecasts for energy generation and demand at the grid level has not been implemented due to the unavailability of models validated with real data coming from the demo sites. These forecasts are expected to be included in the subsequent release.

4.1.6 Planned Features for the Next Release

The table below includes a description of the features, to which extent they currently have been released and which functionalities will be added in a later stage.

A white sheet of paper with black text

Description automatically generated

A screenshot of a computer

Description automatically generated

ID4.1: [SYNERGIES\_ID4.1\_v1.2.pdf](https://txtgroup.sharepoint.com/:b:/r/sites/HESYNERGIES/Shared%20Documents/05-Work%20Packages/WP4/Deliverables/ID4.1/SYNERGIES_ID4.1_v1.2.pdf?csf=1&web=1&e=lsTwbU)

# Technical explanation of DT code

The digital twin is based on the model that DTU has in PowerFactory of Bornholm.

## Scripts

*DT\_v1.ipynb*

The main digital twin where all the functions are being called in a Jupyter Notebook.

*Create\_BH\_net.py*

The digital twin includes a script that transforms this PowerFactory model to a pandapower network. This decision was made as it is easier to perform calculations and generate scenarios and work with ML models in Python.

*Data\_preprocessing\_BH.py*

This script includes processing of smart meter and Scada measurements (based on transformer measurements). Among other things, it aggregates the smart meter measurements to the 16 substations.

*Power\_flow.py*

This script includes the power flow engine. The power flow is computed using the pandapower library and then outputting an Excel file to visualize these results.

*Contingency.py*

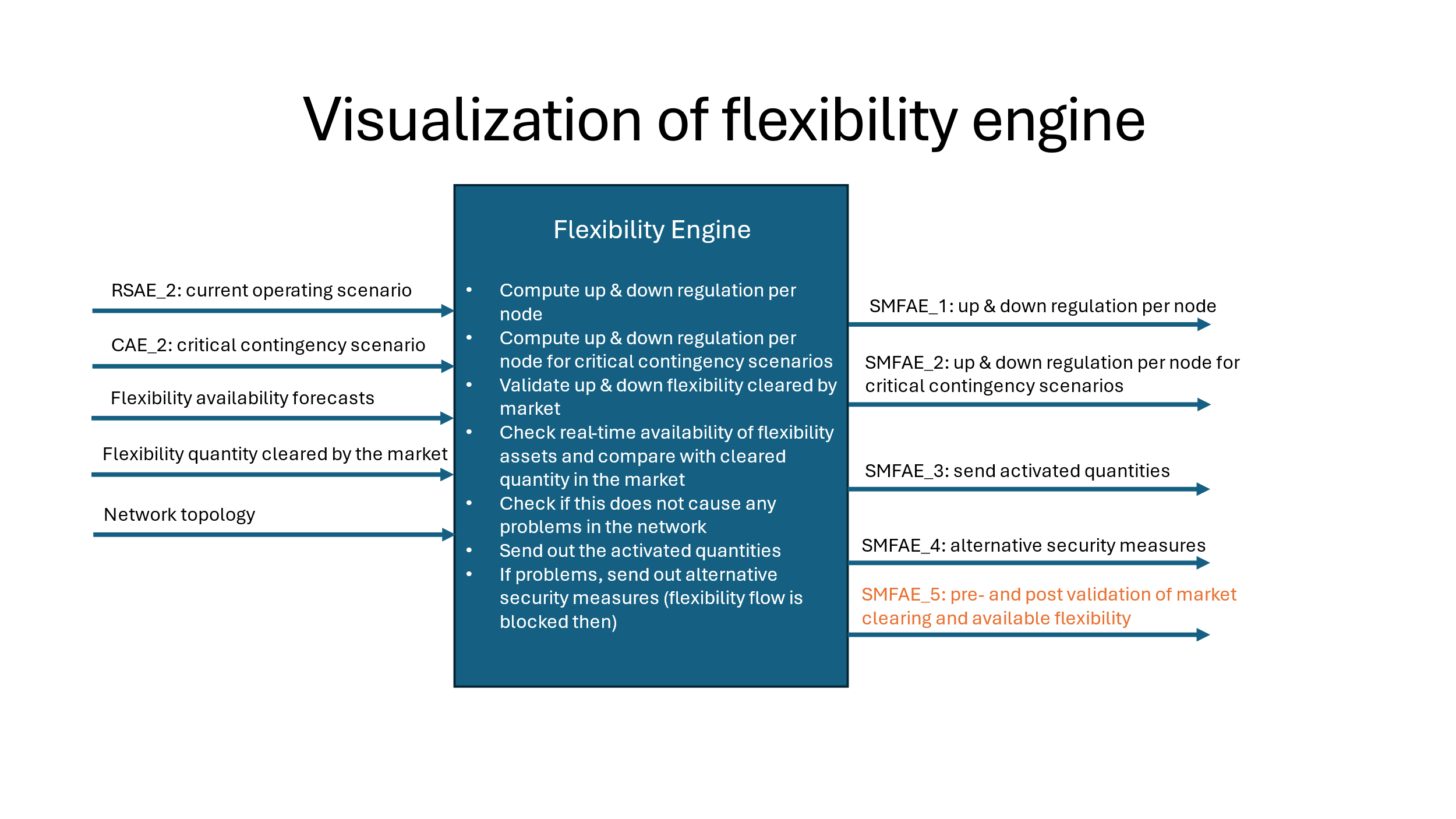
This script includes the contingency engine. It removes lines and computes a power flow. Furthermore, it generates scenarios in the function ‘sample\_normal\_distributions’. The results are then saved to an Excel.

***Flexibility.py***

This script includes the flexibility engine. The required up and down regulation is computed using Gurobi solver by minimizing the flexibility required to keep the system within limits in the function ‘minimize\_flexibility’, this is a simple DC-OPF flexibility minimization.

The implementation for AC OPF is not in this script but implemented in the main Jupyter Notebook and based on the following paper: Strong SOCP Relaxations for the Optimal Power Flow Problem arXiv:1504.06770v4 [math.OC] 31 Oct 2015 Burak Kocuk, Santanu S. Dey, X. Andy Su: <https://arxiv.org/abs/1504.06770>

I have been trying to convert this implementation in Gurobi to Pyomo, in order to be able to use a different solver such as IPOPT or any other solver. Because Gurobi requires a license which can not be deployed on the specific DSO/TSOs of the demo sites itself because they will not have this license.

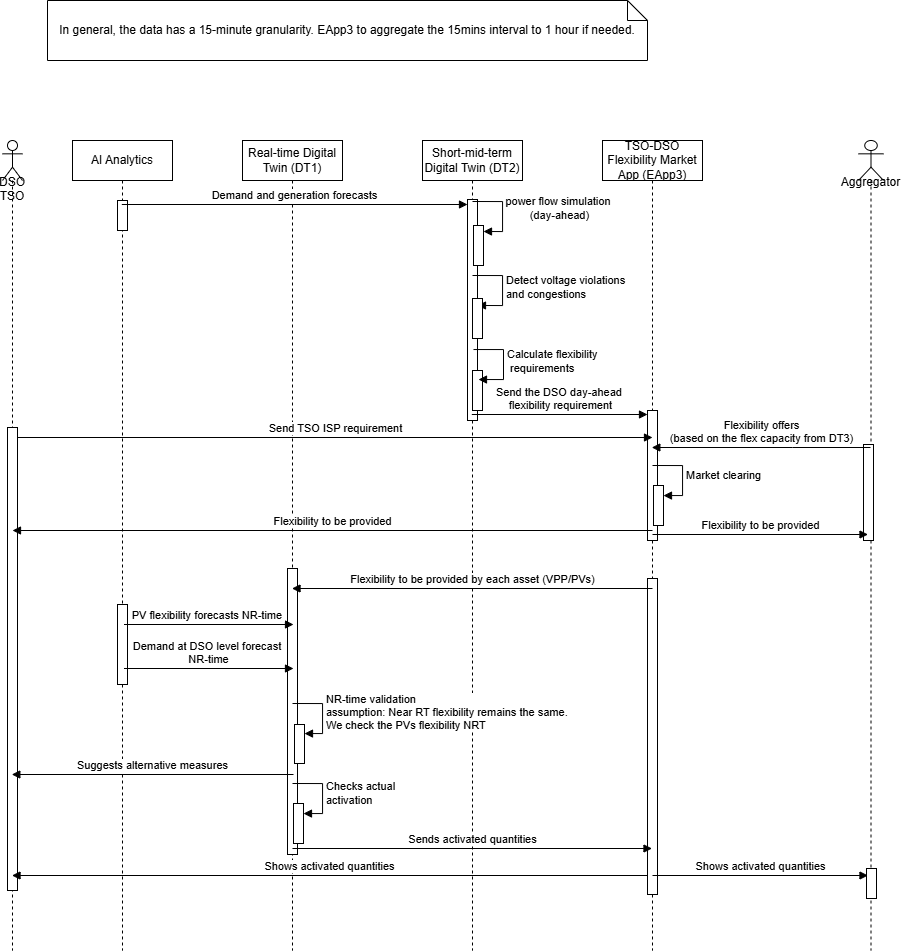


Initial (previous) Version:

A diagram of a company

Description automatically generated

Updated version:



The flexibility process diagram can be found in WP4: <https://txtgroup.sharepoint.com/sites/HESYNERGIES/Shared%20Documents/Forms/AllItems.aspx?id=%2Fsites%2FHESYNERGIES%2FShared%20Documents%2F05%2DWork%20Packages%2FWP4&viewid=58a4475f%2D3698%2D4ee7%2Db198%2D2836aa5ca28a>

Flexibility market clearing

The market is cleared at 12.00 PM (11.00 PM algorithm is started). The marketplace clearing is discussed in more detail here:

D4.1 <https://txtgroup.sharepoint.com/sites/HESYNERGIES/Shared%20Documents/05-Work%20Packages/WP4/Deliverables/D4.1/20240630_ETRA_SYNERGIES_D4.1_Release%20of%20the%20Energy%20Services,%20Applications%20and%20Marketplace_pu_v1.pdf?CT=1728307851722&OR=ItemsView>

## 4Software & datasets

The digital twin is built in Python 3.8.19 and built around the Bornholm grid. It uses the SCADA measurements from EnergydataDK and a smart meter measurement sample.

The packages used for the digital twin are the following:

Pandapower (pip install pandapower["all"]) = 2.13.1

Typing-extensions

Openpyxly

Python-calamine

gurobipy-11.0.3

**Pyomo**

Ipopt

They can be easily installed in a new conda environment using the following prompt:

conda install --name newenv --file packagesdigitaltwinv1.txt

[packagesdigitaltwinv1.txt](https://dtudk.sharepoint.com/:t:/r/sites/SYNERGIESProject--DTUTeam/Delte%20dokumenter/General/Digital%20Twin%201/packagesdigitaltwinv1.txt?csf=1&web=1&e=OqqyJM)

# Current status & next steps

|  |  |
| --- | --- |
| **Functionality** | **Deployed** |
| RSAE\_1: Data cleaning and preprocessing | Yes, but integrated within RSAE engine, should be checked with Greek demo site measurements |
| RSAE\_2: Real-time power flow computation | Yes |
| CAE\_1: Contingency scenario generation | Yes, could include more scenarios |
| CAE\_2: Contingency power flow computation and classification | Yes |
| SMFAE\_1: Locational flexibility margins for real-time operation | Yes |
| SMFAE\_2: Flexibility requirements for critical contingencies | No, but should be similar as previous one, just the critical contingency scenario has to be inserted, so is essentially the same as previous functionality |
| SMFAE\_3: Flexibility validation for near real-time operation | No, as we are in discussions on how data should flow and what this flexibility process should look like |
| SMFAE\_4: Alternative Security control measures | No |

Initial test cases have been completed: <https://txtgroup.sharepoint.com/sites/HESYNERGIES/Shared%20Documents/Forms/AllItems.aspx?id=%2Fsites%2FHESYNERGIES%2FShared%20Documents%2F05%2DWork%20Packages%2FWP5%2FT5%2E2%2FManual%20Functionality%20Test%20Case%20Documentation%20Results&viewid=58a4475f%2D3698%2D4ee7%2Db198%2D2836aa5ca28a>

## Next steps

* Convert the flexibility engine to Pyomo or other
* Integrate forecasts in the engines
* Integrate all the different engines with each other
* Look at Greek demo site data and create the DT for the Greek demo site

The followings can be investigated more as potential replacement for the gurobi:

* **HiGHS** – Efficient and well-maintained, suitable for LP and MILP problems.
* **CBC (Coin-or branch and cut)** – Open-source MILP solver with good performance.
* **GLPK (GNU Linear Programming Kit)** – Works for LP/MILP but may be slower for large-scale problems.
* **Ipopt** – Good for nonlinear optimization (if you need NLP formulations).

We do not keep the exact versioning or jupyter notebook. You can continue with (pycharm, jupyter notebook, vscode, etc) the personal preference, in the end everything will be combined and run in a docker container.